

Metadata of the chapter that will be visualized online

Chapter Title	Clinical Decision-Support Systems	
Copyright Year	2021	
Copyright Holder	Springer Nature Switzerland AG	
Corresponding Author	Family Name	Musen
	Particle	
	Given Name	Mark A.
	Suffix	
	Division	
	Organization/University	Stanford Center for Biomedical Informatics Research, Stanford University
	Address	Stanford, CA, USA
	Email	musen@stanford.edu
Author	Family Name	Middleton
	Particle	
	Given Name	Blackford
	Suffix	
	Division	
	Organization/University	Apervita, Inc.
	Address	Chicago, IL, USA
Author	Family Name	Greenes
	Particle	
	Given Name	Robert A.
	Suffix	
	Division	
	Organization/University	Biomedical Informatics, College of Health Solutions, Arizona State University
	Address	Phoenix, AZ, USA

Abstract	This chapter discusses information technology that provides health-care workers and patients with situation-specific advice that can inform their decision making. The intricacies of the clinical environment, new legislative mandates, and the increasing complexity of medical practice all escalate the demand for clinical decision-support systems (CDSS) that can deliver tailored information at the right time and in the right context. The chapter describes methods for building CDSSs, which include context-specific information retrieval, grouping information within order sets, learning from data, and the use of declarative knowledge representations. The landscape for deploying CDSS technology is evolving rapidly, and the chapter discusses current standards and challenges for CDSS implementation. It concludes with a presentation of opportunities for future research.
Keywords (separated by “ - ”)	Clinical decision support - Infobutton - Order set - Machine learning - Rule-based system - Belief network - Clinical Quality Language - Situation-specific factors

Clinical Decision-Support Systems

Mark A. Musen, Blackford Middleton, and Robert A. Greenes

Contents

24.1	The Nature of Clinical Decision Making – 804	5
24.2	Motivation for Computer-Based CDS – 806	6
24.2.1	Physician Information Needs and Clinical Data Management – 807	7
24.2.2	EHR Adoption and Integration of CDS – 807	8
24.2.3	Precision Medicine – 808	9
24.2.4	Savings Potential with Health IT and CDS – 809	10
24.3	Methods of CDS – 810	11
24.3.1	Acquisition and Validation of Patient Data – 811	12
24.3.2	Decision-Support Methodologies – 812	13
24.3.3	Coda – 829	14
24.4	Translating CDS to the Clinical Enterprise – 830	15
24.4.1	Standard Patient Information Model – 831	16
24.4.2	Adoption of Standard Knowledge-Representation Models – 832	17
24.4.3	Modes of Deployment of CDS – 834	18
24.4.4	Workflow and Setting-Specific Factors – 835	19
24.4.5	Sharing of Best-Practice Knowledge for CDS – 836	20
24.5	Future Research and Development for CDS – 836	21
24.5.1	Standards Harmonization for Knowledge Sharing and Implementation – 836	22
24.5.2	Context-based Knowledge Selection – 837	23
24.5.3	Representation Models – 837	24
24.5.4	Externalizing CDS – 838	25
24.5.5	Usability Research and CDS – 838	26
24.5.6	Data-Driven CDS – 839	27

24.6 Conclusions – 839

References – 842

Learning Objectives

After reading this chapter, you should know the answers to these questions:

- What are the key motivations for clinical decision support?
- What are typical design considerations when building a decision-support system?
- What are some ways in which developers of decision-support systems encode and represent clinical knowledge?
- What are some current standards in the HIT industry that facilitate the construction of decision-support applications?
- What are the current main areas of research and development in clinical decision-support systems?

In this chapter, we discuss information technology aimed at furnishing **clinical decision support** (CDS) – the process that “provides clinicians, staff, patients, or other individuals with knowledge and person-specific information, intelligently filtered or presented at appropriate times, to enhance health and health care” (Osheroﬀ et al. 2007). CDS systems (often referred to as **CDSSs**) communicate information that takes into consideration the particular clinical context, offering situation-specific information and recommendations. Generally, we think of such systems as reasoning about the clinical situation and presenting their conclusions as recommendations to the user. **Information retrieval** systems that find relevant information from repositories of documents with high relevance to a specific patient and clinical context can also serve in this role, but do not themselves make patient-specific recommendations for care. CDS systems do not directly perform clinical decision making; they provide relevant knowledge and analyses that enable the ultimate decision makers—clinicians, patients, and health-care organizations—to develop more informed judgments; hence the importance of the word “support” in the term “CDS”. (Closed-loop systems such as in implantable cardioverter defibrillators and other devices

such as insulin pumps are exceptions to this model; decision support for these systems is not covered in this Chapter.) Ideally, CDSSs can be described in terms of the five *rights* that they aim to accomplish: “provide the right information, to the right person, in the right format, through the right channel, at the right point in workflow, to improve health and health-care decisions and outcomes” (Osheroﬀ et al. 2012).

Systems that provide CDS may be considered in terms of three basic categories: (1) those that use information about the current clinical context to retrieve highly relevant online documents, as with so-called “**Infobuttons**” (introduced in ► Chap. 14); (2) those that are **intelligent systems** that provide patient-specific, situation-specific alerts, advice, reminders, physician order sets, or other recommendations for direct action; or (3) those that organize and present information in a way that facilitates problem solving and decision making, as in **dashboards**, graphical displays, documentation templates, structured reports, and order sets.

Systems that create order sets offer an ideal example of CDS, not only because they can facilitate decision making by providing an actionable set of recommendations such as a combination of orders, but also because they provide a mnemonic function by gathering together items that should be associated in a particular setting. Order sets also can enhance workflow by providing a means to select a group of relevant activities quickly. Not all CDS systems have the ability to optimize workflow, as we shall discuss. In fact, some CDS systems are subject to concern if they are poorly implemented, as they actually can impede workflow or usability.

This chapter provides a review of computer-based decision aids, emphasizing their role and adoption within the current health-care milieu of the United States while keeping an eye on global trends. It offers some thoughts on the nature of the decision-making process, then it provides a description of current implementation strategies and challenges, and it closes with a discussion of critical research questions

that must be addressed to ensure optimal effectiveness of CDS in clinical practice.

24.1 The Nature of Clinical Decision Making

If you ask lay people what the phrase “computers in medicine” means, they often describe a computer program that helps physicians to make diagnoses. Although computers play numerous important clinical roles, people have recognized, from the earliest days of computing, that computers might support health-care workers by helping these people to sift through the vast collections of possible diseases, findings, and treatments.

We can view nearly all the contents of this book as addressing clinical data and clinical decision making. In ► Chap. 2, we discussed the central role of accurate, complete, and relevant data in supporting the decisions that confront clinicians and other health-care workers. In ► Chap. 3, we described the nature of good decisions and the need for clinicians to understand the proper use of information if they are to be effective and efficient decision-makers. In ► Chap. 4 we introduced the cognitive issues that underlie clinical decision-making and that influence the design of systems for decision support. Subsequent chapters have mentioned many real or potential uses of computers to assist with such decision-making. Medical practice *is* medical decision-making, so most applications of computers in health care are intended to have a direct or indirect effect on the quality of health-care decisions. In this chapter, we bring together these themes by concentrating on methods and systems that have been developed specifically to assist health workers in making decisions.

By now, you are familiar with the range of clinical decisions. The classic problem of **diagnosis** (analyzing available data to determine the pathophysiologic explanation for a patient’s symptoms) is only one of these. Equally challenging, as emphasized in ► Chaps. 3 and 4, is the **diagnostic process**—deciding which questions to ask, tests to order, or procedures to

perform, and assessing the value of the results that can be obtained in relation to associated risks or financial costs. Thus, diagnosis involves deciding not only what is true about a patient, but also what data are needed to determine what is true. Even when the diagnosis is known, there often are challenging **management** decisions that test the physician’s knowledge and experience: Should I treat the patient or allow the process to resolve on its own? If treatment is indicated, what should it be? How should I use the patient’s response to therapy to guide me in determining whether an alternate approach should be tried or, in some cases, to question whether my initial diagnosis was incorrect after all? (In that sense, the response to treatment is also a type of diagnostic test.) Also, when a clinician and a patient are faced with alternative treatments, and they seek help to choose among them, the estimation of the chance for cure or the risk of death or of complications is an important decision-making activity. Lastly, many disease processes evolve over time, and evaluation and change in treatment must evolve with them, resulting in the need for guidelines for management that take into account the temporal aspects of prior states and activities in order to decide what to do next. Decision making also may involve integrating data from multiple providers, treatments, and responses to them, as well as increased use of personal sensors and apps to provide additional data, so that aspects of care coordination over time play an important role. This is beyond our scope, but the topic is addressed in part in ► Chaps. 21, 22 and 23.

Biomedicine is also replete with decision tasks that do not involve specific patients or their diseases. Consider, for example, the biomedical scientist who is using laboratory data to help with the design of her next experiment or the hospital administrator who uses management data to guide decisions about resource allocation in his hospital. In addition, new financial models for health-care payment or reimbursement based on value (roughly, patient outcomes and cost of care), rather than on fee-for-service calculations based on individual patient care activities, require that decisions be made on the basis of aggregate

data on outcomes for groups of patients to determine expected norms and to identify outliers. Although we focus on systems to assist with clinical decisions in this chapter, we emphasize that the concepts discussed generalize to many other problem areas as well. In ► Chap. 29, for example, we examine the need for formal decision techniques and tools in creating health policies. As we develop databases that can identify patients with specific diseases, with risks of complications, or in need of specific interventions such as screening tests or immunizations (see ► Chap. 18), **population management** can be used to provide a form of decision support for groups of patients. Some clinical decision support is also aimed directly at patients, in terms of alerts, reminders, or aids to interpretation of information, especially given increasing use of smartphone apps and connected sensors and devices; techniques for assessing prognosis and risk of alternative strategies should involve shared decision making between providers and patients, which is also an important area of activity.

In this chapter, we focus on decision aids for the provider in particular—the clinician seeing the patient at the point of care. The requirements for excellent decision-making fall into three principal categories: (1) accurate data, (2) pertinent knowledge, and (3) appropriate problem-solving, or clinical reasoning, skills.

The data about a patient must be adequate—both accurate and sufficiently comprehensive to include everything relevant for making an informed decision—but they must not be excessive (see ► Chap. 4). Indeed, a major challenge occurs when decision-makers are bombarded with so much information that they cannot process and synthesize the information intelligently and rapidly (see, for example, ► Chap. 21). Thus, it is important to know when additional data will confuse rather than clarify and when it is imperative to use tools (computational, visual, or otherwise) that permit data to be summarized for easier cognitive management (see ► Chap. 4). Operating rooms and intensive-care units are classic settings in which this problem arises; patients are monitored extensively, numerous

data are collected, and decisions often have to be made on an urgent basis.

Access to data from all relevant sources is required but difficult to achieve in practice. Patients may be seen in different venues, including a primary care office (perhaps via a telecare visit), a specialist office, an emergency room, a laboratory or imaging facility, a hospital, an extended care facility, or they may be monitoring themselves at home. Each venue may be using different and incompatible EHRs or data repositories, terminologies, and data representation models. Access to these data and interoperability across data repositories may be limited. Typically, the data available are only those within a health system and its EHR—which may include affiliated office practices, clinics, emergency rooms, and hospitals, or by interaction with an available **health information exchange** (HIE) (see ► Chap. 15).

Equally important is the quality of the available data to input to a CDSS. In ► Chap. 2, we discussed imprecision in terminology, illegibility and inaccessibility of records, and other opportunities for misinterpretation of data. Similarly, measurement instruments or recorded data may simply be erroneous; use of faulty data can have serious adverse effects on patient-care decisions. Thus, clinical data often need to be validated.

Even good data are useless if we do not have the knowledge necessary to apply them properly. Decision-makers must have broad knowledge of medicine, in-depth familiarity with their area of expertise, and access to pertinent additional information resources. Their knowledge must be accurate, with areas of controversy well understood and questions of personal choice well distinguished from those where a more prescriptive approach is appropriate. Their knowledge must also be current; in the rapidly changing world of medicine, facts decay just as certainly as dead tissue does.

Good data and an extensive factual knowledge base still do not guarantee a good decision; good problem-solving skills are equally important. Decision-makers must know how to set appropriate goals for a task, how to reason about each goal, and how to make explicit

the trade-offs between costs and benefits of diagnostic procedures or therapeutic maneuvers. Skilled clinicians draw extensively on personal experience, and new physicians soon realize that good clinical judgment is based as much on an the ability to reason effectively and appropriately about what to do as it is on formal knowledge of the field or access to high-quality patient data. Thus, clinicians must develop a strategic approach to the selection and interpretation of diagnostic tests, understand ideas of sensitivity and specificity, and be able to assess the urgency of a situation. Similar issues relating to test or treatment selection, in terms of costs, risks, and benefits, must be understood. Awareness of biases and of the ways that they can creep into problem-solving also are crucial (see ► Chap. 3). Also, as noted above, patient preferences and concerns must be adequately addressed as part of the decision-making process. Thus, good communication and interaction with the patient is essential. This brief review of issues central to clinical decision-making serves as a fitting introduction to the topic of computer-assisted decision-making: Precisely the same topics are pertinent when we develop a computational tool for CDS. The program must have access to good data, it must have extensive background knowledge encoded for the clinical domain in question, and it must embody an intelligent approach to problem-solving that is sensitive to requirements for proper analysis, appropriate cost-benefit trade-offs, and efficiency.

24.2 Motivation for Computer-Based CDS

Since the 1960s, workers in biomedical informatics have been interested in CDS systems because of a desire both to improve health care and to understand better the process of medical decision-making. Building a computer system that attempts to process clinical data to offer situation-specific advice can provide insight into the nature of medical problem solving and can enable the creation

of formal models of clinical reasoning. At the same time, construction of such systems offers obvious societal benefits if the computer programs can aid practitioners in their care of patients and can lead to better clinical outcomes. Although the more academic considerations have provided strong motivation for work in the area of computer-based decision aids over several decades, the recognition of the importance of CDSSs as practical tools has increased markedly in recent years as a result of the inexorable growth in health-care complexity and cost, as well as the introduction of new health-care legislation, regulatory initiatives, and payment incentives aimed at addressing these trends—which have all made the development and broad adoption of CDS technology a priority.

The twenty first century has seen changes in health-care practices that make the development of CDS technology particularly necessary. Computer-based CDS has taken on increasing urgency for four reasons: (1) increasing challenges related to knowledge and information management in clinical practice, thus increasing physician information needs, (2) the ubiquity of electronic medical records and the desire to enhance health care through the communication and integration of the relevant data, (3) the goal of delivering increasingly personalized health-care services—tailored to the patient's preferences for care and to his or her individual genome, and (4) the growing evidence that CDS can not only improve the quality of care delivered but also reduce costs of care. We consider these four reasons in the sections that follow.

We note another factor that is shaping new directions for CDS which we will explore further at the end of this chapter—the trend toward use of personal devices and apps, as well as more portable, distributed laboratory or analytic procedures, to provide a growing range of sources of data to incorporate into decision making. These devices and apps also provide an ability to track, monitor, and intervene early whenever needed, both interacting with the user/patient directly, and also for those conditions warranting it, with the provider.

24.2.1 Physician Information Needs and Clinical Data Management

Modern health care is characterized by an ever-expanding knowledge base of clinical medicine, and by a growing clinical data base describing every patient characteristic from phenotype to genotype (Kohn et al. 2002). Despite the growing amounts of data and knowledge with which physicians need to work, health-care practitioners have seen the average time for a clinical encounter steadily curtailed, particularly in the United States, where the pressures of the prevalent fee-for-service reimbursement system and a concomitant rise in the amount of paperwork required for administrative management and billing continue to squeeze practitioners (Baron 2010). Studies of information needs among physicians in clinical practice have long revealed that unanswered clinical questions are common in ambulatory clinical encounters, with as many as one or two unanswered clinical questions about diagnosis, therapy, or administrative issues arising in every visit (Covell et al. 1985). Prior to the broad adoption of EHRs, in as many as 81% of clinical encounters in ambulatory care, clinicians were found to be missing critical information, with an average of four missing items per case (Tang et al. 1994, 1996). Currently, even with an EHR, providers may face major challenges in accessing relevant information, acquiring a complete picture of the patient's clinical state and history, and knowing what further testing or therapeutic actions are best to take. Prior to broad EHR adoption, studies suggested that as many as 18% of medical errors might be due to inadequate availability of patient information (Leape 1994). Today, conversely, the overabundance of information in the EHR can lead to 'information chaos' and may make it difficult for the clinician to find relevant information for clinical decision making (Melnick et al. 2019; Beasley et al. 2011). The demands for increased information management – in the setting of an ever-

expanding clinical knowledge base, and now as more data sources are available and use of EHRs is almost ubiquitous – are primary drivers for the adoption of CDS systems. (See ► Chap. 23 for a deeper discussion of physician information needs.)

24.2.2 EHR Adoption and Integration of CDS

The motivations for adoption of EHRs and CDS are affected by the way in which health care is financed and paid for, by the structure and organization of the health-care system of a nation or a region, and by political forces that can create constraints, regulations, and incentives. In this section, we use the United States as an example to demonstrate these influences.

Health-care safety and quality concerns, coupled with a seemingly inexorable rise in health-care costs, have led in the United States to a variety of cost-containment and quality-improvement strategies in recent years. Health-care delivery in the United States is in the midst of a profound transformation, in part due to Federal public policy efforts to encourage the adoption and use of health information technology (HIT). The American Recovery and Reinvestment Act (ARRA) of 2009, and the **HITECH regulations** within it, created incentives for the widespread adoption of health information technologies (Blumenthal 2009; see ► Chap. 29). These public policy efforts, while ultimately suffering some reversals for political reasons, are often viewed as a long-term adjunct to current health-care-payment reform efforts in the U.S., and a prelude to additional health-care-delivery redesign, payment reform, and cost containment. As recently as 2012, only 34.8% of physicians in ambulatory practice in the US used a basic or comprehensive electronic medical record (Decker et al. 2012), and only 26.6% of U.S. hospitals used health information technologies in inpatient care-delivery settings (DesRoches et al. 2012), although these numbers have had a rapid upward trajectory.

The ARRA and HITECH policies, and the resulting technology adoption, changed the practice of medicine and clinical care delivery in both beneficial and untoward ways (Sittig and Singh 2011).

To achieve meaningful and effective use of HIT, the software must be viewed as one component of a complex sociotechnical system, in which all elements must work effectively (Institute of Medicine 2011a). The movement toward value-based reimbursement, a more recent U.S. trend, encourages emphasis on wellness, prevention, and early intervention in disease processes, and requires a new level of connectivity, continuity, and coordination of care across multiple venues is gaining in acceptance. A prominent example is the advent of Accountable Care Organizations (ACOs; McClellan 2015), which require emphasis on not only integration of data from all sources (clinical and financial) but also aggregate, population-based data on outcomes, costs, and identification of outliers. These alternative payment models are distributing upside and downside financial risk based upon quality outcomes, patient experience, and costs, and thus call for CDSSs to support both payers and providers.

One of the principal motivations for EHR adoption is to provide an infrastructure with which to improve the quality, safety, and efficacy of health-care delivery. In the past decade, the U.S. government placed considerable emphasis on the adoption of quality measures and quality-reporting requirements as part of meaningful use of HIT (Clancy et al. 2009; Institute of Medicine 2011a). Quality measures, despite their ability to provide feedback that stimulates improved performance by the clinician, are only part of the process needed to make the desired improvements. Prospective, proactive clinical decision support must also be in place. The U.S. government's rules for **meaningful use** of HIT, which became progressively more demanding over a 4–6 year period, required only minimal CDS compliance in phase I and II, but Phase III of the meaningful use regulations in 2016 was intended to increase the mandate for CDS in EHR systems substantially by requiring APIs (application program interfaces) for access to

EHR data (Blumenthal and Tavenner 2010; Adler-Milstein et al. 2017). More recently, the Medicare Access and CHIP Reauthorization Act of 2015 (MACRA),¹ the Twenty-first Century Cures Act of 2016 in the United States,² and other initiatives have promoted a wide array of additional enhancements and improvements to the use of health information technology in clinical practice. Notably, the Twenty-first Century Cures Act removed from U.S. Food and Drug Administration (FDA) consideration as a medical device software devoted to performing the functions of an EHR, administrative tools, or providing clinical decision support. Significant push-back to the burden on both EHR vendors and on health-care organizations to comply with Meaningful Use Phase III resulted in relaxation of these constraints. Meaningful Use regulations have been superseded in the United States by the Cures Act, promoting more sweeping goals for interoperability and access to quality measures (Sinsky and Privitera 2018; see ► Chap. 29). More recently, political considerations regarding health-care financing in the United States have made the speed of adoption of such measures somewhat uncertain.

From a more worldwide perspective, interoperability of data and of CDS knowledge models are essential for wide adoption, as well as for improving the evidence base from which knowledge is derived.

24.2.3 Precision Medicine

The fundamental model for the practice of medicine has undergone dramatic change in the past century or so. The objectives of clini-

1 MACRA (2015). The Medicare Access and CHIP Reauthorization Act of 2015 (MACRA). Retrieval February 19, 2020: ► <https://www.cms.gov/medicare/quality-initiatives-patient-assessment-instruments/value-based-programs/macra-mips-and-apms/macra-mips-and-apms.html>

2 twenty-first Century Cures Act. H.R. 34, 114th Congress (2016). Retrieval February 19, 2020: ► <https://www.gpo.gov/fdsys/pkg/BILLS-114hr34enr/pdf/BILLS-114hr34enr.pdf>

cal care have shifted radically from the archaic goal of correcting putative imbalances of bodily humors to the scientific understanding of pathophysiology, of mechanisms for eliminating pathogens, and of remedying biological aberrancies. The resulting view of medicine as the application of biological principles was at the core of the report produced by Abraham Flexner (1910) that upended medical education in the early twentieth century and that had led to the **reductionist biomedical model** that prevailed for the rest of that century. More recently, however, George Engel's **biopsychosocial model** (Engel 1977) brought to the fore of clinical care the need to address psychological and social factors in clinical treatment plans in addition to underlying biomedical problems. By the end of the twentieth century, it became increasingly accepted that CDS requires not only the rapid communication of appropriate scientific medical knowledge, but also the adaptation of that knowledge to reflect the psychological and social situation that would temper the application of the knowledge. Added to this complexity is the aging of the population, owing in part to advances in health and health care. The result has been a much higher burden of chronic diseases, multiple diseases, and multiple testing and treatment options, with both their positive and negative consequences that must be balanced—all contributing to the increasing intricacy of care. Indeed, the view of care is so complex for some patients that a major role of CDS is to integrate models of patient state and context (provider, setting, specialization, and activity) to provide selective visualization, analysis, and decision-making support for optimal care management (Greenes et al. 2018).

As a further extension of these trends, the genomic era in which we now live has further increased the need for clinical practice to reflect **precision medicine** and the need to tailor care to individual factors in ways that never before were imaginable (Ginsburg and Willard 2009). Precision medicine is characterized by decision making that may take into account patient personal history, family history, social and environmental factors, along with genomic data and patient preferences

regarding their own care (see ► Chap. 26). In this approach, clinical decision making is explicitly patient-centered in new ways, bringing the best evidence at the genetic level to bear on many clinical scenarios, while incorporating patient preferences for acquiring and applying genetic information (Fargher et al. 2007; Marcial et al. 2018). The increasing use of genomics in medicine (Chan and Ginsburg 2011) is generating data that outstrip the information and knowledge-processing capabilities of practitioners, and many clinicians feel threatened by the impending tsunami of additional knowledge that they will need to master (Baars et al. 2005). As precision medicine becomes the norm, primary-care and specialist practitioners alike will need to manage their patients by interpreting genomic tests along with myriad other data at the point of care. It is hard to imagine how clinicians will manage to perform such activities without substantial computer-based assistance. Informatics is well suited to support a personalized approach to clinical genomics (Ullman-Cullere and Mathew 2011).

As mentioned earlier, another related change is growing recognition of the importance of promoting optimal health and wellness, not just by treating disease but by encouraging healthy lifestyles, fostering compliance with health and health-care regimens, and carrying out periodic health-risk assessments. Key to prescriptive medicine are tools to support *prospective medicine* (Langheier and Snyderman 2004)—assisting the acquisition of a detailed family history, social history, and environmental history, using personal apps and sensors, providing health-risk assessments, and managing genomic information (Hoffman and Williams 2011; Overby et al. 2010).

24.2.4 Savings Potential with Health IT and CDS

CDS has been shown to influence physician behavior (Colombet et al. 2004; Lindgren 2008; Schedlbauer et al. 2009), diagnostic test ordering and other care processes (Bates and Gawande 2003; Blumenthal and Glaser

2007), and the costs of care (Haynes et al. 2010), and it may have a modest impact on clinical outcomes (Bright et al. 2012). While there is enormous promise for HIT and CDS, their implementation is not without potential peril: HIT poorly designed or implemented, or misused, can generate unintended consequences (Ash et al. 2007; Harrison et al. 2007; Bloomrosen et al. 2011), and introduce new types of medical errors (Institute of Medicine 2011a).

Only a handful of studies have examined the **return on investment (ROI)** for HIT, and even fewer have investigated ROI for decision-support specifically. The value of CDS in terms of ROI is difficult to measure. Isolated studies of various hand-crafted systems in academic centers have shown value (Wang et al. 2003; Kaushal et al. 2006), but adoption elsewhere has often been problematic. Broad adoption has not occurred, for many reasons discussed later in this chapter, including the proprietary nature of systems for CDS and for representation of knowledge, the lack of interoperability of data and knowledge, the mismatch of CDS to workflow, and usability concerns.

Systematic reviews of the scientific literature, such as the one performed by Bright et al. (2012), have not been able to demonstrate an effect of CDS on patient outcomes except in the short term. This finding is not surprising, however, because the time point at which CDS occurs is often long before a final outcome, and many intervening factors may have a greater effect. In the case of CDS for many chronic diseases whose complications ensue over years or decades, it simply may be impractical to continue longitudinal studies of sufficiently long duration to be able to measure meaningful differences in outcome. Nevertheless, economic simulation studies of the potential effect of CDS on chronic diseases have demonstrated benefit in the long term (McClellan 2015; Bu et al. 2007; Adler-Milstein et al. 2007).

Historically, the adoption of CDS technology has been motivated by a virtuous desire to enhance the performance of clinicians when dealing with complex situations. The recent

advent of legal, regulatory, and financial drivers, as well as the increasing importance of personalizing medical decision making on the basis of genomic data, now make CDS an essential element of modern clinical practice.

24.3 Methods of CDS

As we have already noted, CDS systems (1) may use information about the current clinical context to retrieve pertinent online documents; or (2) they may provide patient-specific, situation-specific alerts, reminders, physician order sets, or other recommendations for direct action; or (3) they may organize information in ways that facilitate decision making and action. Category (2) largely consists of the various computer-based approaches (“classic” CDS based on the application of intelligent systems) that have been the substrate for work in informatics since the advent of applied work in probabilistic reasoning and artificial intelligence in the 1960s and 1970s. Such systems provide custom-tailored assessments or advice based on sets of patient-specific data. They may follow simple logics (such as algorithms), they may be based on decision theory and cost-benefit analysis, or they may use probabilistic approaches or derive their conclusions on the basis of machine learning from large amounts of data. Some diagnostic assistants (such as DXplain; Barnett et al. 1987) suggest differential diagnoses or indicate additional information that would help to narrow the range of etiologic possibilities. Other systems suggest a single best explanation for a patient’s symptomatology. Other systems interpret and summarize the patient’s record over time in a manner sensitive to the clinical context (Shahar and Musen 1996). Still other systems provide therapy advice rather than diagnostic assistance (Musen et al. 1996).

CDS systems can achieve their results using a wide variety of computational methods. These approaches include Bayesian probabilistic reasoning (see ► Chap. 3), the use of machine learning to make predictions based on large amounts of data (often

from the EHR), performing inference using IF–THEN rules, or by the identification of relevant templates for the clinician to fill in, such as knowledge-based groupings of physician orders (order sets), or some combination of these approaches (James et al. 2013). CDS systems may acquire the data on which they base their recommendations interactively from users or directly from a health information system (or some combination of these approaches). We now discuss the issues that drive CDS system design, and we highlight how these issues are manifest in current clinical decision aids.

24.3.1 Acquisition and Validation of Patient Data

As mentioned in the introduction to this chapter, a prerequisite to any decision-making process is having available all the data that are needed to perform the required actions. As emphasized in ► Chap. 2, few problems are more challenging than the development of effective techniques for capturing patient data accurately, completely, and efficiently. You have read in this book about a wide variety of techniques for data acquisition, ranging from keyboard entry, to speech input, to methods that separate the clinician from the computer (such as scannable forms, real-time data monitoring, and intermediaries who transcribe written or dictated data for use by computers).

The problems of data acquisition go beyond entry or extraction from the EHR, or from other repositories, of the data themselves, however. A primary obstacle is that we lack standardized ways of expressing most clinical situations in a form that computers can interpret. As discussed in detail in ► Chap. 7, there are several controlled medical terminologies that health-care workers use to specify precise diagnostic evaluations (e.g., the International Classification of Diseases and SNOMED CT), clinical procedures (e.g., Current Procedural Terminology and LOINC codes), drugs (e.g., RxNorm),

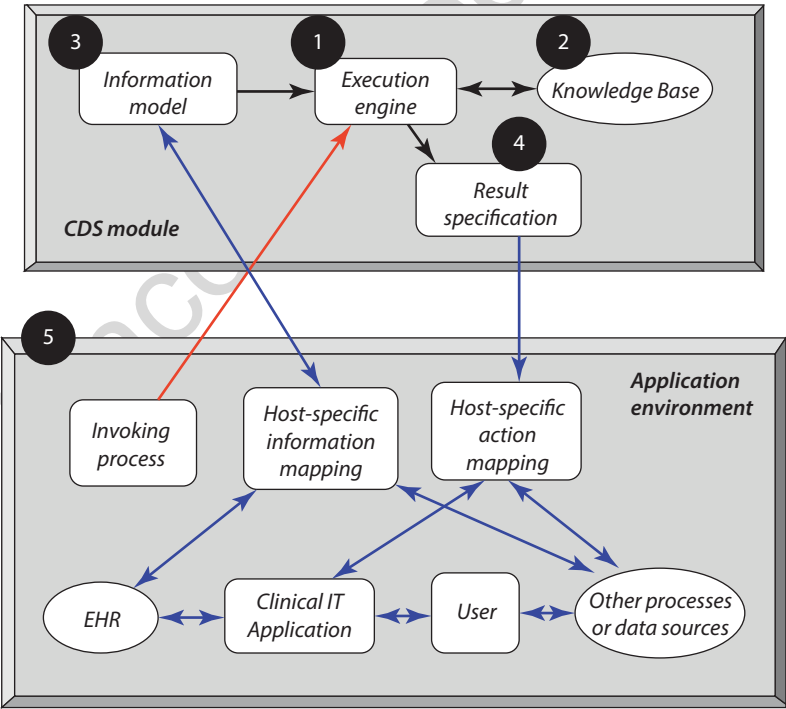
and so on. Still, there is no controlled terminology that can capture all the nuances of a patient's history of present illness or findings on physical examination. There is no coding system that can reflect all the details of physicians' or nurses' progress notes. Given that much of the information in the medical record that we would like to use to drive decision support is not available in a structured, machine-understandable form, there are clear limitations on the data that can be used to assist clinician decision-making. The prose of progress notes, consultation notes, procedure or operation reports, discharge summaries, and other documents contains an enormous amount of information that never makes it to the coded part of the EHR. Nevertheless, even when computer-based patient records store substantial information only as free-text entries, the data that are also available in coded form (typically, diagnosis codes and prescription data) can be used to significant advantage (van der Lei et al. 1991).

The desire to access information from the EHR that may be available only in text has been a topic of great interest to the CDS community. Some information systems provide options for **structured data entry**, asking clinicians to use fill-in-the-blanks forms or templates on the computer screen to enter patient-related information that otherwise would be entered as part of a prose note. In general, providers have resisted such human–computer interfaces, often finding it restrictive and cumbersome to make selections from predefined menus when they would much rather express themselves more freely in prose. In fact, structured templates or methods for collecting information about a particular problem or finding are themselves often regarded as a form of CDS, in that they provide an organized framework and a mnemonic function. Fortunately, work in **natural language processing** has made major advances in recent years, making it increasingly possible to mine the textual notes of EHRs to identify information that might bear on the CDS process (see ► Chap. 8).

24.3.2 Decision-Support Methodologies

When designing a CDS system, it is helpful to consider several aspects (■ Fig. 24.1). We consider these aspects as components interacting with one another, because each requires thought and effort to accomplish, and can be facilitated by developing standard, interoperable approaches for implementing them. Further, they can be independently enhanced over time, and are able to be shared and reused if engineered in a component-based manner. Much CDS can be accomplished by direct implementation and embedding into a clinical system, but the result is a set of one-off implementations that cannot be readily maintained, updated, or shared, especially of concern given that an organization may have hundreds, if not thousands, of CDSS artifacts in operation.

There are five aspects of CDS that, at least in principle, can be viewed as independent of each other, and that work together to produce CDS capability (Greenes 2014). These aspects include: (1) the method of computation or inferencing or, more generally, execution of the CDS function (e.g., targeted information retrieval, hard-coded algorithms, Bayesian estimation, neural-network classification, rule-logic evaluation); (2) the knowledge needed to carry out the function (e.g., prior and conditional probabilities, rule assertions, mathematical formulae, clinical guidelines); (3) the information model that governs how data are provided to the CDSS (i.e., the data needed and the method of encoding, such as specific FHIR data from the clinical setting, laboratory-test results encoded in LOINC, environmental data, or medical facts in specific coding schemes); (4) the type of recommendation to be provided (e.g., prediction,



■ Fig. 24.1 A conceptual model of CDS components. There are five aspects of CDS that work together to produce CDS capability. The aspects are: (1) the method of computation or inferencing; (2) the knowledge needed to carry out the computation; (3) the information model for the data that drive decision making; (4) the type of

recommendation to be provided; and (5) how the process interacts with the application environment, including how it is invoked and how the data and recommendations are communicated. (Adapted from Greenes 2014)

action, classification, relevant citations); and (5) how the process interacts with the application environment, including how it is invoked (e.g., a process launched by an event monitor, by the user explicitly, or by being embedded in workflow) and how the data and recommendations are communicated (e.g., provided via a FHIR API or delivered as a popup alert). In this section, we focus primarily on the methods of computation required for CDS (Aspect 1 of ■ Fig. 24.1), and to a lesser extent on the knowledge needed during computation (Aspect 2). We consider the other elements in subsequent sections.

Aspect 1 was central to much of the early work on CDS development, beginning in the 1960s and largely led by academic centers, which focused on developing and exploring different computational models of CDS in the absence of any health IT infrastructure in which to embed the systems. In recent years, much of CDS has been built by vendors of proprietary systems, giving rise to a range of different approaches, often embedded in those systems, and with methods that are less able to be inspected, formalized, and shared. Some of the trends in health-care delivery that we cited in ► Sect. 26.2 are hoped to give rise to increased interoperability and sharing in vendor-developed systems. Even if current commercial products may not manifest the different components of ■ Fig. 26.1 distinctly, we adopt this conceptual focus in our discussion of various strategies for CDS to clarify the underlying principles.

24.3.2.1 Context-Specific Information Retrieval

Early developers of CDS systems argued that decision support entails more than identifying relevant information that can help a clinician to solve a problem; it was believed that a CDS system has to suggest specifically how the problem should be solved. Thus, simply providing information for the user to read and digest was not considered “real” decision support. Nevertheless, as information-retrieval systems have improved over the years—with better performance characteristics—it is often hard to maintain that such systems do not offer decision support in a genuine sense.

Indeed, such systems are now ubiquitous in health care. (► Chap. 25 provides a comprehensive discussion of information-retrieval methods.)

The simplest, and perhaps most common, form of CDS uses contextual information from an EHR to perform information retrieval from a database of information about online documents. A person viewing data in an EHR may see selectable icons (**infobuttons**) next to the names of drugs, laboratory tests, patient problems, or other elements of the patient record, or the items themselves may be hyperlinked to an information retrieval engine. Clicking on an infobutton causes the clinical information system to perform a query on the database, providing the user with one or more immediately accessible resources that can offer more information about the item in question. Alternatively, the system may automatically query one or more of those external resources and return the results of the queries for display (Cimino et al. 2002). Clicking on an infobutton next to a drug, for example, might allow the user to access information about customary dosing, side effects, or alternative medications (see ■ Fig. 14.15 in ► Chap. 14). The query that retrieves the links to the documents is tailored based on whatever is next to the infobutton icon on the screen. The query may also take into account contextual information, such as patient-related data, the activity in which the user is engaged, and the role of the user in the health-care enterprise (physician, nurse, patient, and so on).

An **infobutton manager** mediates the queries between the clinical information system and the available information resources. The standards development organization Health Level Seven has created a standard for “context-aware knowledge retrieval,” leading to infobutton managers that have been adopted by many commercial EHR vendors.³

3 HL7 International (2014). HL7 Version 3 Standard: Context Aware Knowledge Retrieval Application (“Infobutton”), Knowledge Request, Release 2. Retrieval February 19, 2020: ► https://www.hl7.org/implement/standards/product_brief.cfm?product_id=208

Infobutton managers need to anticipate how the clinical context might tailor the specific query performed by any given infobutton, so that the result of the query is highly precise and relevant to the situation at hand. Detailing specifically how contextual information might alter the queries performed by each infobutton type can be tedious, and the process requires developers to be adept at second-guessing all the reasons that might cause a user to click on a particular infobutton. Current research concentrates on the development of a Librarian Infobutton Tailoring Environment (LITE; Jing et al. 2015) that promises to aid the authoring of infobutton queries via “wizards” and other user-interface conveniences.

Although infobuttons are unquestionably important knowledge resources, many people would argue that they are not true CDS systems. Infobuttons retrieve relevant information for a user, but they do not explicitly address particular *decisions* that the user needs to make. The possible reasons that a user might click on an infobutton are folded into the query specification at the time that the infobutton is created; at runtime, of course, there is no way for the system to know exactly why the user selected the infobutton. Infobutton managers therefore require sophisticated query capabilities, but they do not need to reason from a clinical situation to a particular recommendation.

When the goal is to generate a situation-specific recommendation regarding diagnosis or therapy, developers need to turn to methods that can perform some kind of inference. The sophistication of the required technique is a function of the kind of inference that is necessary to render a result for the user.

24.3.2.2 Organizing or Grouping Information as a CDS Method

As we have noted earlier, a valuable kind of CDS is to organize information to provide a ready collection of items that need to be considered together (e.g., order sets for particular clinical indications or settings, documentation templates for particular purposes, or structure

and formatting of reports). This capability serves not only as a convenience to the user, but also as a mnemonic function (i.e., offering a check list; Gawande 2009).

When patients are admitted to the hospital with a particular condition such as a suspected myocardial infarction or pneumonia, when hospital staff must prepare them for diagnostic procedures or surgery, or when they need to be transferred to another care team or to be discharged home, there often are stereotypical groups of medical orders that physicians tend to request. For example, patients with possible pneumonia often require a chest x-ray examination, the recording of vital signs at certain intervals, the administration of supplementary oxygen, cultures of their sputum or blood, and administration of antibiotics. When patients are admitted to a hospital with the diagnosis of pneumonia, the EHR can automatically suggest to the treating physicians that such a set of orders be considered. Systems that produce such order sets can use the clinical situation to tailor the recommended orders (e.g., the computer may not recommend a chest x-ray examination if the patient has just had one; knowledge that the patient has been placed on an artificial ventilator may trigger a separate set of associated orders for consideration).

Clinicians routinely face many other stereotypical tasks, such as describing the results of a diagnostic procedure such as an imaging study or reporting the sequence of events that took place during a surgical operation. Automated systems that recognize the clinical context can provide a tailored template for the clinician to fill in, helping to ensure that information is provided accurately and completely.

Throughout their professional activities, clinicians constantly must keep in mind large amounts of information when treating patients with even routine medical problems and reporting the results of their work. The use of simple mnemonics such as check lists can be remarkably effective in ensuring that care givers remember everything that they need to do or to report in a given context (Algaze et al. 2016; Alamri et al. 2016; Pageler

et al. 2014). Transforming such check lists into groups of orders to consider, groups of features to note in a diagnostic evaluation, or groups of steps that may have been followed when performing a surgical procedure can help clinicians to remember important details and to improve both the quality of medical care and what clinicians may report about it.

Computationally, systems that offer order sets or templates to clinicians typically assemble predefined information from collections of text strings or database entries. The HL7 organization has issued a standard for creating libraries of order sets. Systems that suggest specific order sets to users by making selections from such a library are generally embedded within the EHRs that call on their services, however. Thus, the specific methods that these systems use tend to be proprietary.

Researchers have experimented with the use of machine-learning techniques to create order sets on the fly based on empirical data. For example, Wang et al. (2018) have built a system that infers groups of orders for particular situations by examining a data warehouse for the individual orders that physicians have administered historically. The system then suggests that the groups of orders that it discerns from the database may be reasonable for clinicians to administer as an ensemble.

24.3.2.3 Hardcoding Clinical Algorithms

From a computational perspective, there is nothing simpler than encoding an algorithm directly in a computer program. In health care, there is generally nothing simpler than defining a decision process in terms of a flowchart. Numerous CDS systems thus have taken problem-specific flowcharts designed by clinicians and encoded them for use by a computer. Although such flowcharts have been useful for the purpose of triaging patients in urgent-care situations and as a didactic technique used in journals and books where an overview for a problem's management has been appropriate, computable interactive flowcharts have been largely rejected by physicians as too simplistic or generic for routine use (Grimm et al. 1975),

other than for particular kinds of computations such as drug dose calculations (e.g., for pediatric patients). In addition, the advantage of their implementation on computers has not been clear; the use of simple printed copies of the algorithms generally has proved adequate for clinical care (Komaroff et al. 1974). A noteworthy exception that gained enormous attention in the early 1970s was a computer program deployed in Boston at what was then the Beth Israel Hospital (Bleich 1972); it used detailed algorithmic logic to provide advice regarding the diagnosis and management of acid-base and electrolyte disorders. More recently, such branching-logic and other inference methods approaches have been widely adopted in the administrative information systems that third-party payers use to process requests to pre-certify payment for expensive services such as MRI studies and elective surgery (the HL7 Da Vinci Project is a notable body of work in this area).

Although representing clinical algorithms simply as computer code offers a very direct approach to implementing a CDSS, there are obvious challenges that occur when it becomes necessary to refine or update the program's behavior. Every modification requires reprogramming the system. It may not always be obvious how to reprogram the system to render the desired behavior and making the necessary changes in one part of the program may have unintended consequences when other parts of the program execute. Thus, although it may seem appealing simply to hardcode clinical algorithms, developers of decision support systems generally seek more flexible mechanisms to represent clinical knowledge and to reason about it. We discuss this matter further in ► Sect. 24.4.5, and we address future research directions in this area in the Conclusion.

24.3.2.4 Learning from Data

Considerable flexibility is achieved when the CDSS is largely data-driven. The advent of extraordinarily fast computers and the ability to process enormous amounts of data has led to an explosion of interest in the use of

large datasets to learn patterns in the data to support clinical decision making (James et al. 2013). The success of such data-driven methods has led to considerable excitement about the use of “big data” in health care. It is important to appreciate, however, that workers in biomedical informatics benefited from such approaches long before computers could manage the enormous datasets that they process today.

Probabilistic Systems

Attempts to drive computer-based decision support from relationships inferred directly from data began during the earliest days of research in biomedical informatics. A seminal article in 1959 first introduced Bayes theorem and also value theory (later known as utility theory) into health-care decision making (Ledley and Lusted 1959). In the 1960s, workers in the field recognized that they could use computers to apply **Bayes’ rule** to determine the posterior probability of diseases based on observations of patient-specific parameters (see ► Chap. 3). Such calculations were based on the determination of appropriate probabilistic relationships between findings and diseases by analyzing available datasets. Large numbers of **Bayesian diagnosis programs** have been developed in the intervening years, many of which have been shown to be accurate in selecting among competing explanations of a patient’s disease state.

Among the most significant of the early experiments were those of F. T. de Dombal and his associates (1972) in England, who focused on the diagnosis of acute abdominal pain. De Dombal’s group used a **naïve Bayesian model** that assumed that there are no conditional dependencies among findings (i.e., a model that makes the inappropriate but convenient assumption that the presence of a finding such as upper abdominal pain never affects the likelihood of the presence of a finding such as lower abdominal pain). Using surgical or pathologic diagnoses as the gold standard, de Dombal’s group used sensitivity, specificity, and disease-prevalence data for various signs, symptoms, and test results to calculate, using Bayes’ theorem, the proba-

bility of seven possible explanations for acute abdominal pain (appendicitis, diverticulitis, perforated ulcer, cholecystitis, small-bowel obstruction, pancreatitis, and nonspecific abdominal pain). To keep the Bayesian computations manageable, the program made the “naïve” Bayesian assumptions of (1) conditional independence of the findings for the various diagnoses, (2) mutual exclusivity, and (3) exhaustiveness of the seven diagnoses (see ► Chap. 3).

In one system evaluation (de Dombal et al. 1972), physicians filled out data sheets summarizing clinical and laboratory findings for 304 patients who came to the emergency department with abdominal pain of sudden onset. The data from these sheets provided the attributes that were analyzed using Bayes’ rule. Thus, the Bayesian formulation assumed that each patient had one of the seven conditions and it selected the most likely one on the basis of the recorded observations.

In contrast to the clinicians’ diagnoses, which were correct in only 65–80% of the 304 cases (with accuracy depending on the individual clinician’s training and experience), the program’s diagnoses were correct in 92% of cases. Furthermore, in six of the seven disease categories, the computer was more likely to assign the patients to the correct disease category than was the senior clinician in charge of the case.

De Dombal’s system began to achieve widespread use—from emergency departments in other countries to the British submarine fleet. Surprisingly, however, the system never obtained the same degree of diagnostic accuracy in other settings that it did where it had initially been deployed—even when adjustments were made for differences in prior probabilities of disease. There are several reasons possible for this discrepancy, which are relevant for all Bayesian CDS systems. The most likely explanation is that there may be considerable variation in the way that clinicians interpret the data that must be entered into the computer. For example, physicians with different training or from different cultures may not agree on the criteria for identification of certain patient findings on physical

examination, such as “rebound tenderness.”⁴ Another possible explanation is that there are different probabilistic relationships between findings and diagnoses in different patient populations.

Although a naïve Bayesian model may have limitations in accurately modeling a diagnostic problem, a major strength of this approach is computational efficiency. When the findings that bear on a hypothesis are assumed to be conditionally independent, then the order in which the findings are considered in the Bayesian analysis does not matter. The computer starts by considering a given finding, the prior probability of each possible diagnosis under consideration (generally the prevalence of each diagnosis in the population), and the conditional probabilities of the finding (or the absence of the finding) given each diagnosis (or the absence of the diagnosis)—the *sensitivity* and *specificity* of the finding (see the discussion of these concepts in ► Chap. 2). The computer then applies Bayes’ rule to calculate the posterior probability of each diagnosis given the value of the finding. The computer now is poised to update the probability of each diagnosis given the value of a second finding. The prior probability for each diagnosis in this case is not the prevalence of the diagnosis in the population, however. Having applied Bayes’ rule once, we have more information than we had at the start. We can treat the *posterior probability* of each diagnosis given the first finding as the *prior probability* of the diagnosis when we apply Bayes’ rule a second time. When it is time to consider a third finding, the posterior probability for each diagnosis after processing the second finding serves as the prior probability for the next application of Bayes rule. The process continues until the value of each finding has been considered. This **sequential Bayes** approach was explored as early as the 1960s for the diagnosis of congenital heart

disease (Gorry and Barnett 1968) and has been used in many CDS systems since.

In recent years, the use of naïve Bayesian models has been seriously challenged by the adoption of systems based on Bayesian **belief networks** (see below), which can take advantage of efficient algorithms that overcome the limiting assumptions of naïve Bayesian approaches—albeit at the cost involved in creating a more complex (and more nuanced) model of the underlying probabilistic relationships. In addition, systems that adopt the sequential Bayes approach lack the ability to choose the next test to be applied in a manner that can optimize reasoning. Bayesian systems that use more sophisticated reasoning strategies based on decision analysis (see ► Chap. 3) can use utility theory to identify the test that will provide the most useful information given the current state of reasoning. Considerations of cost, discomfort to the patient, and the availability of the test can influence the utility of the particular choice.

Machine Learning

The availability of large biomedical datasets and computers and algorithms that can process huge amounts of data are revolutionizing health care and the life sciences. Machine learning is everywhere in health care, from interpreting radiographic images to predicting utilization of health care services to identifying potential adverse drug events. It is not surprising that decision-support based on machine-learning models is becoming increasingly common in clinical settings. Such systems have been in place for decades, but now they are assuming increasing prominence, given the new opportunities that the renaissance in machine learning is offering all of biomedicine in the era of “big data.”

There are a host of **supervised learning techniques** that can determine how data are associated with hypotheses (James et al. 2013), and that consequently can be trained on EHR data to infer conclusions based on some set of input data. For example, the decision-support capabilities of the patient monitoring systems discussed in ► Chap. 21 often apply statistical methods to the current data stream to

4 *Rebound tenderness* is pain that is exacerbated when the physician presses down on the abdomen and then suddenly releases, generating a “rebound” when the abdomen returns to its baseline position.

infer corresponding classifications to inform care providers of the patient's current state. Regression analysis or more sophisticated techniques, such as artificial neural networks and support vector machines, when applied to routinely collected patient data, have enabled investigators to develop venerable decision aids such as the APACHE system (Knaus et al. 1991; Zimmerman et al. 2006), which offers prediction models providing prognostic information regarding patients in the ICU (see ► Chap. 21).

Recent work demonstrates the value of applying data-driven techniques to a wide range of clinical problems for which clinicians may benefit from decision support, from assessing newborns in the ICU (Saria et al. 2010) to development of models that can suggest which patients might benefit most from palliative care (Avanti et al. 2018). Scores of start-up companies have emerged in recent years, each hoping that different large datasets and specialized machine-learning techniques will lead to new insights about particular clinical problems in an effort to enhance decision making.

From the beginning, such machine-learning approaches have been criticized when used as the basis for CDS, primarily because of the lack of transparency in how data-driven methods reach their conclusions (Shortliffe et al. 1979). Because the associations between findings and diagnoses are inferred as the system is trained on the data and are not readily available for inspection, such systems cannot offer guidance as to *why* they might reach particular conclusions. This inability to explain the basis for their recommendations is especially important when the recommendations of a system might be overly fitted to the peculiarities of a dataset drawn from a patient population different from the one to which the system is being applied (as may have been the case with de Dombal's system). Because the output of a CDSS based on a machine-learning algorithm must be accepted at face value, there is typically no way to know what biases may exist in the data that trained the system or what clinically relevant intermediary states may have led the system to reach

its final conclusion. There is currently intense interest in being able to develop new technologies powered by machine-learning methods that in some measure can explain the basis of their reasoning and that can allow users to assess the likelihood that particular recommendations are appropriate in their clinical setting (Ribeiro et al. 2016).

24.3.2.5 Declarative Representation of Knowledge

In simple Bayesian systems and in those CDSSs based on machine-learning algorithms, data are provided as input into the system, and the output is a classification of the data—often a diagnosis on which to act. Since the 1970s, however, workers in biomedical informatics have pursued the development of decision-support technologies that attempt to encode in a more explicit way how the inputs to the system relate to the outputs. The goal is to encode models—models of reasoning, models of pathophysiology, models of probabilistic relationships, models of the evidence in support of alternative treatment options, and other relevant models—in a way that forms the basis for a system's computation to derive an appropriate recommendation from the input data. Such systems are often built with the objective that the underlying models be examinable and explainable. Often, there is a desire to make those models editable, so that the models easily can be updated in light of new discoveries and new understanding. The unifying idea in this approach is that CDSSs are built with a computer-based representation of the knowledge that drives system behavior. There are many ways to represent knowledge in computers (Musen 2014), and each strategy has different strengths and weaknesses.

Bayesian Belief Networks

Much of the early interest in the naïve, sequential Bayesian approach stemmed from a conviction that it simply was impractical to construct Bayesian systems in which the assumption of conditional independence was lifted: There would be too many probabilities to assess when building the system, and the necessary computation could be intractable.

Work on the use of **belief networks**, however, has demonstrated that it actually is realistic to develop more expressive Bayesian systems in which conditional dependencies are modeled explicitly—often by taking advantage of approximate algorithms for concluding the posterior probabilities that are computationally efficient in most cases. (Belief networks are described in detail in ► Chap. 3.) Currently, many modern CDS systems that make recommendations based on probabilistic relationships use belief networks as their primary representation of the underlying clinical situation, and then “solve” the belief network at runtime to calculate the posterior probabilities of the conditions represented in the graph. The use of belief networks is popular because the formalism makes probabilistic relationships perspicuous, overcomes the assumption of conditional independence, and enables the attendant probabilities to be learned from analysis of appropriate data sets (for example, EHR data). The approach has been demonstrated in numerous diagnostic systems, from belief networks that ascertain the status of newborns from data in the neonatal ICU (Saria et al. 2010), systems for differential diagnosis (Shwe et al. 1991; Middleton et al. 1991), to belief networks that offer interpretations of biomedical image data (Kahn et al. 1997).

Because making most decisions in medicine requires weighing the costs and benefits of actions that could be taken in diagnosing or managing a patient’s illness, researchers also have developed tools that draw on the methods of **decision analysis**. Decision analysis adds to Bayesian reasoning the idea of explicit decisions and of **utilities** associated with the various outcomes that could occur in response to those decisions (see ► Chap. 3). One class of programs for decision-analysis is designed for use by the analysts themselves; such programs are of little use to the average clinician or patient, however (Pauker and Kassirer 1981). A second class of programs uses decision-analysis concepts within systems designed to advise physicians who are not trained in these techniques. In such programs, the underlying decision models gener-

ally have been prespecified—either as decision trees that enumerate all possible decisions and all possible ramifications of those decisions or as belief networks in which explicit decision and utility nodes are added, called **influence diagrams** (Shachter 1986).

We say that belief networks and influence diagrams represent knowledge in a **declarative** manner, because a belief network provides an inspectable, editable model of the probabilistic relationships that are relevant to the decision problem under consideration. If a developer wants to designate a new relationship between two entities in the network, then she needs only to augment the model by adding a new edge to the graph that encodes the given network. The network thus provides a transparent mechanism to communicate what the system “knows” about the probabilistic relationships among the entities in the application domain, and changing the network intrinsically changes the behavior of the system when it reasons about those entities. This is different from the case of a hardcoded algorithm or a system based on machine learning, where the knowledge is not readily inspectable or editable in a direct manner.

Rule-Based Approaches

Although belief networks provide a convenient mechanism to encode knowledge about the world in a declarative fashion, they are only one of several alternative frameworks that may be used to drive CDS based on explicit models of the application area. Since the 1970s, workers in medical AI have been exploring the use of methods that emphasize the modeling of rules that describe conclusions that can be reached about the decision problem and the variables that may predicate those conclusions. Often called **knowledge-based systems**, these programs reason about the clinical situation by examining a collection of rules of the form, “If some set of conditions is true, then conclude that something else is true” (► Fig. 24.2). Although these rules may be created through machine-learning approaches, they often are built by manually encoding relationships between clinical data and corresponding conclusions that are

Rule 507

IF:

- 1) The infection that requires therapy is meningitis
- 2) Organisms were not seen on the stain of the culture
- 3) The type of infection is bacterial
- 4) The patient does not have a head injury defect, AND
- 5) The age of the patient is between 15 years and 55 years

THEN

The organisms that might be causing the infection are
Diplococcus-pneumoniae and Neisseria-meningitidis

Fig. 24.2 A rule from a rule-based system. Rules are conditional statements that indicate what conclusions can be reached or actions taken if a specified set of conditions is found to be true. This rule, taken from the CDSS known as MYCIN, is able to conclude probable bacterial causes of infection if the five conditions in the premise are all found to be true for a specific patient

offered by experts in the field or by examination of evidence reported in the scientific literature. When a knowledge-based system is encoded using rules, it is referred to as a **rule-based system** (Buchanan and Shortliffe 1984).

Rule-based systems provide an important mechanism for developers to build CDS capabilities into modern information systems. From CDS systems that interpret ECG signals to those that recommend guideline-based therapy, rules provide an extremely convenient means to encode the necessary knowledge. Rule-based systems require a formal language for encoding the rules, plus an interpreter (sometimes called an **inference engine**) that operates on the rules to generate the necessary behavior.

Perhaps the best-known rule-based CDSS is one that was never put into clinical use, but that has served as a prototype for the many rule-based systems that have followed. The program, known as MYCIN, combined a diagnostic component with an advisor component that suggested appropriate management of patients who have infections (Shortliffe 1976). MYCIN's developers believed that straightforward algorithms or probabilistic approaches were inadequate for this clinical problem in which the underlying knowledge was poorly understood and even the experts often disagreed about how best to manage specific patients, especially before definitive bacterial culture results became available. As a result, the researchers were drawn to the

use of interacting rules to represent knowledge about organisms that might be causing a patient's infection and the antibiotics that might be used to treat it.

Knowledge of infectious diseases in MYCIN was represented as *production rules* (see Fig. 24.2). A production rule is an IF-THEN conditional statement. The conclusions drawn by one production rule may be used to satisfy the premises of other rules when a system of rules is used for reasoning by an inference engine. MYCIN's power was derived from such rules in a variety of ways:

- The MYCIN program determined which rules to use and how to chain them together to make decisions about a specific case. The MYCIN reasoning program used an approach called **backward chaining**; whenever a rule was being considered and the system did not know whether the condition on the left-hand side of the rule (i.e., the premise) was true, MYCIN would look backward to see whether the knowledge base contained any other rules that, when evaluated, could conclude information that might inform the evaluation of the current rule's premise. (Nearly all contemporary rule-based systems, on the other hand, use an inference method known as **forward chaining**: Whenever a production rule “fires” and the conclusion of that rule is proven to be true, the system looks forward for other rules in the rule base that use the concluded information in their premise, and that therefore also might be able to fire now that the new conclusion is known to be true.)
- The rules often formed a coherent explanation of MYCIN's reasoning—those that applied to the current decision were displayed in response to a user's questions (Fig. 24.3). Although rules were stored in a machine-readable format, English translations could be displayed.

By removing, altering, or adding rules, system developers could modify the program's knowledge structures rapidly, without explicitly reprogramming or restructuring other parts of the knowledge base. The rule base

In this series, how many of the cultures were positive for E.coli
(ORGANISM-1)?

(typical system prompt)

****WHY**

(request from user to determine motivation for question)

[i.e. WHY is it important to determine the number of cultures which were
positive for the organism?]

(system's rephrasing of the WHY
question, and answer)

[1.0] This will help to determine whether there is significant disease
associated with this occurrence of the organism.

It has already been established that

[1.1] the culture was taken from a sterile source

[1.2] the number of cultures in the series is greater than 2

Therefore, if

[1.3] the number of cultures which were positive for ORGANISM-1
is greater than 2

Then there is suggestive evidence (.5) that there is significant disease
associated with this occurrence of the e.coli (ORGANISM-1).

[RULE049]

****WHY**

(repeated WHY to determine relevance of previous explanation)

[i.e. WHY is it important to determine if there is significant disease
associated with this occurrence of the organism?]

[2.0] This will help to determine a diagnosis for which the infection should
be treated.

It has already been established that

[2.1] the infection is not one of: primary bacteremia, meningitis

[2.2] for some current culture of the patient it is true that the site
of the culture is blood

Therefore, if

[2.3] there is significant disease associated with this occurrence of
the organism

Then

It is definite (1.0) that the diagnosis for which the infection should
be treated is secondary-bacteremia

[RULE103]

■ **Fig. 24.3** Two examples of MYCIN's explanation capabilities. User input is shown in boldface capital letters and follows the double asterisks. The system

expands each ["WHY"] question (enclosed in square brackets) to ensure that the user is aware of its interpretation of the query

thus offered the kind of inspectability and editability that we discussed when considering belief-network representations of knowledge.

The developers evaluated MYCIN's performance on therapy selection for patients with blood-borne bacterial infections (Yu et al. 1979a), and for those with meningitis (Yu et al. 1979b). In the latter study, MYCIN gave advice that compared favorably with that offered by experts in infectious diseases—results that ushered in enormous excitement about the potential of rule-based systems to offer high-level clinical advice in real-world situations.

The developers of MYCIN had to construct their own syntax for encoding rules and had to program their own inference engine to evaluate the rules. However, now there are many open-source and proprietary **rule engines** that provide custom-tailored editors for writing rules and inference engines that can execute the rules at runtime. For example, JESS is a popular Java-based rule engine that can be licensed from Sandia National Laboratory and that currently is free for academic use. Drools is an open-source rule engine developed by the JBoss community that also has had substantial adoption.

Developers use JESS, Drools, and proprietary rule engines to create CDS systems that contain multiple rules that, as with MYCIN, can chain together to generate conclusions based on a sequence of inference steps. Decision support sometimes requires multiple rules to execute at runtime, together generating a final recommendation that derives from the consequences of the rules chaining off one another.

In most installed information systems, however, rule-based decision support is much simpler and also more limited. Most deployed CDS systems have rules that generally do not chain together, but that are triggered individually, each time that either there is a relevant change to the data in a patient database that should generate an alert, there is a time-related event that should trigger a reminder, or an action is performed (e.g., by a user, when the action is established in the workflow as a triggering event). Each rule examines the state of the database and generates a corresponding action, alert, or a reminder that is usually sent to a particular clinician or to members of the health-care team. Such rules are of the general form: Event – Condition – Action (ON Event, IF Condition, THEN Perform Action) and they are commonly referred to as **ECA rules**.

For example, **Arden Syntax** became an international standard for ECA rules known as **Medical Logic Modules** (MLMs), endorsed by HL7 and ANSI in 1999 (■ Fig. 24.4). Arden Syntax provides a standard mechanism for declaring the variables about whose values the system will perform its reasoning (values that derive from data in the clinical information system); the conditions that, if true, would predicate specific actions; the actions that should be taken, and the kinds of events that would invoke or trigger the rule. The standard was created with the hope that the informatics community would develop whole libraries of MLMs, all written in Arden Syntax, that could operate in any clinical environment where an information system could interpret the standard format.

A significant obstacle to the sharing of MLMs, however, is that Arden Syntax is, in fact, just a syntax. What is missing from the

standard is any notion of the *semantics* of the data on which the MLMs operate. When an MLM executes, the variables that are used in the logic of the rule are bound to values that derive from the patient database of the information system in which the MLMs operate. Arden Syntax specifies that the individual database queries needed to determine the values of the variables should appear within the “curly braces” of variable definitions in the portion of the MLM known as the “data slot” (see ■ Fig. 24.4). What a developer should include within the curly braces depends on the particular schema of the relevant patient database and mechanism for performing queries. EHR information models and the way in which elements are coded differ from system to system. Thus, all system-specific aspects of MLM integration need to be provided within the curly braces. To adapt an MLM for use in a new environment, a programmer needs to consider the variables on which the MLM operates, determine whether those variables have counterparts in the local patient database, and write an appropriate query that will execute at runtime.

The **curly braces problem** is compounded because there may be assumptions regarding the semantics of the variables themselves that may not be obvious to the local implementer: If the MLM refers to serum potassium, should the logic be executed if the original specimen was grossly hemolyzed?⁵ If a serum potassium value is not available in the database, but there is a value for a whole-blood potassium, should the MLM be executed using that value instead?⁶ If there is no serum potassium value available for today, but there is one from last night, should the logic execute using the most recent value? Decision rules cannot simply be dropped from one system into another and be shared effortlessly; rather, considerable thought, analysis, and computer skill needs to go into writing the appropriate database que-

5 If the red blood cells in a specimen *hemolyze* (burst), they release potassium, which can cause an inaccurate elevation in the measured potassium value.

6 The *serum* is the liquid that is left when the cells are removed from whole blood.

MAINTENANCE:

Title: Diabetic Foot Exam Reminder;;
 Mlmmname: Diabetic_Foot_Exam.mlm;;
 Arden: Version 2.8;;
 Version: 1.00;;
 Institution: Intermountain Healthcare ;;
 Author: Peter Haug (Peter.Haug@imail.org) ;;
 Specialist: Peter Haug (Peter.Haug@imail.org) ;;
 Date: 2011-11-28;;
 Validation: testing;;

LIBRARY:

Purpose: Alert for Diabetic Foot Exam Yearly;;
 Explanation: This MLM will send an alert if the patient is a diabetic (diabetes in problem list or discharge diagnoses) and Foot Exam is recorded within the last 12 months;;
 Keywords: diabetes; Foot Exam;;
 Citations: Boulton AJM, Armstrong DG, Albert SF, Frykberg RG, Richard Hellman, Kirkman MS, Lavery LA, LeMaster JW, Mills JL, Mueller MJ, Sheehan P, Dane K, Wukich DK. Comprehensive Foot Examination and Risk Assessment. Diabetes Care. 2008 August; 31(8): 1679–1685;;
 Links: http://en.wikipedia.org/wiki/Diabetic_foot_ulcer;;

KNOWLEDGE:

Type: data_driven;;
 Data: Problem_List_Problem := object [Problem, Recorder];
 Problem_List := read as Problem_List_Problem {select problem, recorded_by from Problem_List_Table};
 Patient_Dx_Object := object [Dx];
 Diabetic_Dx := read as Patient_Dx_Object {ICD_Discharge_Diagnoses};
 Foot_Examination := object [Recorder, Observation];
 Observation := object [Abnormality, Location, Size, Units];
 Foot_Exam := read as Foot_Examination latest {select Recorder, Observation.Abnormality, Observation.Location, Observation.Size, Observation.Units from PE_Table};
 Registration_Event := event { registration of patient };
 ICD_for_Diabetes := (250 , 250.0 , 250.1 , 250.2 , 250.3 , 250.4 , 250.5 , 250.6 , 250.7 , 250.8 , 250.9);;
 Evoke: Registration_Event;;
 Logic: if (Diabetic_Dx.Dx is in ICD_for_Diabetes or (exist Problem_List and "Diabetes" is in Problem_List.Problem)) then Diabetes_Present := true ;
 endif;
 if (Diabetes_Present and exist Foot_Exam and Foot_Exam occurred not within past 12 months) then
 conclude true ;
 endif;
 conclude false ; ;;
 Action: write "Patient is a diabetic with no Diabetic Foot Exam in last 12 months. Please order or perform one.";;

■ **Fig. 24.4** This medical logic module (MLM), written in the Arden syntax, prints a warning for health-care workers whenever a patient who has diabetes is registered for a clinic visit and has not had a documented foot examination in the past year. The evoke slot defines a situation that causes the rule to be triggered; the logic slot encodes the decision logic of the rule; the action slot

defines the procedure to follow if the logic slot reaches a positive conclusion. The data slot defines the variables that are to be used by the MLM; the text between curly braces must be translated into queries on the local patient database when the MLM is deployed locally. (Courtesy of P. J. Haug, Intermountain Healthcare, with permission)

ries that go within the curly braces to make such rules operational.

In the case of Arden Syntax, developers write rules to deal with one clinical problem at a time. There may be one MLM to deal with the problem of administering a drug like penicillin to a patient with a history of penicillin allergy; another MLM may report that a patient has a dangerously low serum potassium value. Unlike the rules in MYCIN, MLMs are generally not intended to interact with one another or to be chained together to generate complex inferences. MLMs may be coerced to chain together when one MLM posts to the patient database a value that can trigger another MLM. This mechanism also allows one MLM to set up information in the database that might invoke another MLM in the case of some future event, thus enabling the recommendation of actions that unfold over time, as in the case of many clinical practice guidelines for chronic diseases. Although this approach allows developers to program complex problem-solving behavior, the technique has the same disadvantages that came to light with chaining rule-based systems such as MYCIN: When the rule base grows to a large size, interactions among rules may have unanticipated side effects. Furthermore, when rules are added to or deleted from a previously debugged knowledge base, there may be unexpected system behaviors that emerge as a result (Clancey 1983; Heckerman and Horvitz 1986).

For MLMs to work well in practice, moreover, the rules need to be tailored to the particular clinical environment—triggered by appropriate workflow events, interacting with particular kinds of participants, customizing logic to account for various business and workflow processes, and notifying the user in setting-specific ways. To customize an MLM to account for such considerations requires that it become less portable. Much of the effort required to introduce CDS systems into the health-care enterprise involves precisely such adaptations. To accelerate portability, MLM developers must seek a balance between a generic specification of logic that is widely agreed upon, and site-specific customizations that will facilitate the use of that

logic. Achieving the right balance will always remain an elusive target (see also ► Sect. 24.5.4).

More General Representations of Knowledge

The variety of approaches for building CDS systems that we have described so far suggests that each method has significant strengths and weaknesses. Hardcoded branching-logic systems can be very easy to build but difficult to update and maintain when the program code becomes complicated. Belief networks and influence diagrams offer precision in probabilistic reasoning when the goal is to make a classification of some clinical phenomena, but they offer limited capabilities if the goal is to generate a plan for medical therapy on the fly or to simulate some biomedical situation. Rule-based systems can help to decompose decision problems into tractable IF–THEN chunks, but they typically are limited in what these chunks can express about a clinical problem. All of these approaches are constrained in their inability to reason about clinical abstractions (e.g., knowing that elevated serum potassium is *a kind of* electrolyte abnormality) and by their inability to make inferences about situations that entail nuance or complexity.

In recent years, developers of CDS systems have become increasingly interested in the use of more general representations of clinical knowledge to provide more sophisticated capabilities for decision support. These interests have paralleled the surge of enthusiasm in the World Wide Web community for the notion of developing **knowledge graphs** that encode facts about the world in a lattice of nodes that represent the kinds of entities in the world and links between them that represent relationships among those entity types (Noy et al. 2019). Unlike belief networks, which typically contain dozens of nodes, knowledge graphs often contain hundreds or thousands or even millions of nodes—or more. Whereas belief networks store numeric, probabilistic relationships among the entities in the graph, knowledge graphs store symbolic, logical connections among the enti-

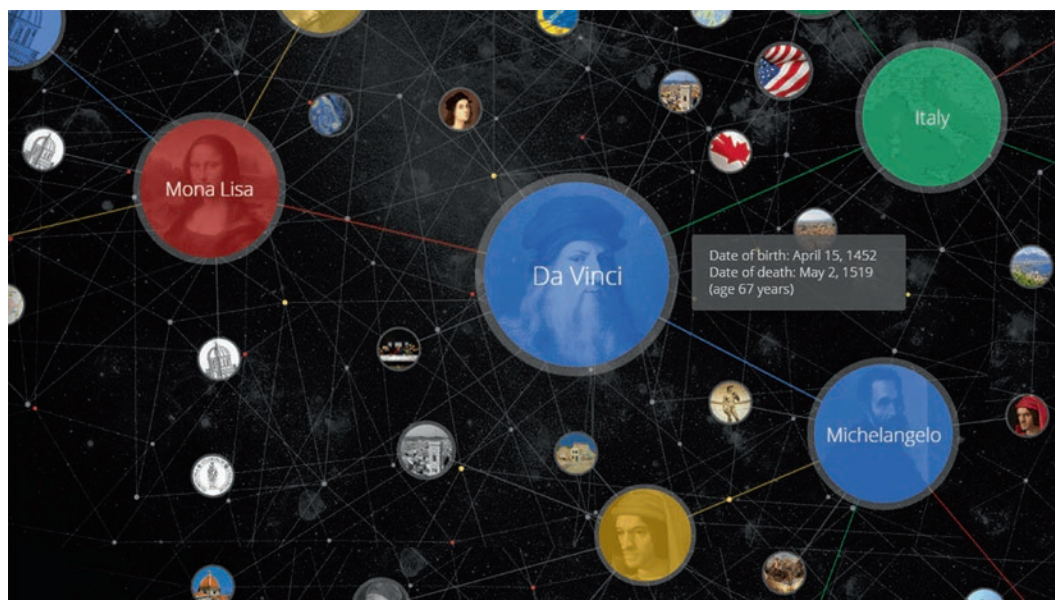


Fig. 24.5 Google depicts its knowledge graph artistically as a vast collection of nodes representing entities in the world, linked to other entities in a rich network. The graph allows Google's search engine to highlight attributes of entities for which users might search, to

expand searches by including synonyms, and to collect information about entities related to the subject of a search for presentation to the user. (Source: ► <https://searchengineland.com/laymans-visual-guide-googles-knowledge-graph-search-api-241935>)

ties in addition to various properties of each entity. There currently is no standard way of creating a knowledge graph and there is not even consensus on what features a knowledge graph should support. Nevertheless, virtually all well-known e-commerce sites—from Google to Facebook to Bing to eBay—use a graph-based representation of knowledge to provide key functionality for users hoping to perform a variety of tasks.

Google, for example, has a knowledge graph that comprises more than one billion classes of entities and instances of those entities, and more than 70 billion facts about those entries (► Fig. 24.5) (Noy et al. 2019). The graph makes it possible for Google to know that, when a user searches for some abstraction, such as “medications for diabetes,” the user may be interested in different *kinds of* medications for diabetes, such as insulin and hypoglycemic drugs. Thus, the knowledge graph causes the search engine to bring up links to related entities when a user performs a search, and the graph helps to disambiguate the user's query when the objective of the search may not be clear.

Many commercial CDS systems are adopting knowledge-graph technology as a component of their software architecture. Because there is no standard knowledge-graph approach, these systems adopt different knowledge-graph formalisms and perform different tasks using their graphs. “IBM Watson,” for example, is a name that encompasses a family of CDS systems that have been deployed for a range of clinical domains in recent years. Although these IBM Watson systems have had a variety of capabilities and evolving computational architectures, a common theme has been their use of knowledge graphs to implement precise information retrieval and to assist with natural language understanding. Thus, a knowledge graph can allow an IBM Watson system to interpret a user's natural language query and to locate a document or a fact that is itself stored in a knowledge graph to respond to that query. Indeed, the use of knowledge graphs in general search engines such as Google and Bing provides such technologies with capabilities that make them useful for decision support. When reporting search results, both

Google and Bing display the contents of their knowledge graphs for the indicated item in a “knowledge box” at the upper right of the page. These knowledge boxes often can be very effective in offering specific information to address user queries.

The simplest form of knowledge graph is one that encodes an enumeration of the kinds of entities in an application area and that places them in an abstraction hierarchy, indicating when elements of one entity form a subclass (or superclass) of another. This type of data structure is often referred to as an **ontology**. Ontologies are like **controlled terminologies** (see ► Chap. 7), in that they provide a standard means for referring to the types of entities that comprise a domain, but they organize those entities using a graph that makes explicit the semantics of the relationships among the entities. **SNOMED CT** and the **NCI Thesaurus** are examples of commonly used controlled terminologies that are represented using knowledge graphs in a manner that also makes them ontologies.

Ontologies are important in biomedical informatics for their explicit representation of abstraction relationships, thus facilitating interpretation of high-throughput experiments, aiding information retrieval and natural language processing, and indexing data (Bodenreider and Stevens 2006). CDS systems can use ontologies that encode knowledge about different application areas in a manner that aids reuse of that knowledge in new settings and that makes it easy for developers to update the knowledge as their understanding of the application area evolves. Such systems use ontologies (or knowledge graphs more generally) to encode clinical knowledge in a manner that may overcome some of the limitations of the more prevalent CDS architectures. These systems make an explicit distinction between the *static knowledge* of the clinical domain (e.g., knowledge of the specifications entailed by a clinical practice guideline) and the *problem-solving knowledge* needed to apply the static knowledge to a particular patient (e.g., the means to generate specific prescriptions for medications based on the general guideline recommendations and the particular clinical situation

that the patient is experiencing). This distinction makes it possible for system builders to address different elements of the knowledge needed to be represented in the computer using tailored approaches and tools (Musen 1998; deClerq et al. 2004).

The ATHENA-CDS system exemplifies this component-oriented approach (Goldstein et al. 2000). ATHENA-CDS is a computer system that is integrated with the HIS that has been used by the U.S. Department of Veterans Affairs (VA), known as VistA.⁷ ATHENA-CDS has been installed at several VA medical centers and has remained in continuous use at the Palo Alto VA medical center since the 1990s. ATHENA-CDS offers advice regarding patients who have certain chronic diseases, whose physicians would like to treat those patients in accordance with recognized evidence-based clinical practice guidelines (■ Fig. 24.6). ATHENA-CDS draws on several electronic knowledge bases, each one constituting a knowledge graph that encodes the knowledge of a particular guideline (e.g., for hypertension, for hyperlipidemia, for diabetes, and so on). Each time that a patient with a relevant diagnosis (e.g., hypertension) is seen in the outpatient clinic, ATHENA-CDS takes as input the corresponding guideline knowledge base and patient-specific data from the VistA EHR and generates as output suggestions to the clinician for treating the patient to ensure that the treatment is consistent with the care that the guideline would recommend. Because the standard documents that define clinical practice guidelines can be long and complicated, it is extremely helpful for the computer to focus the clinician’s attention on precisely which interventions should be considered to guarantee that the patient’s care is consonant with the medical evidence captured by a given guideline (■ Fig. 24.7).

ATHENA-CDS was engineered using an approach that separates out static knowledge about the clinical application area from knowledge about problem solving (i.e., knowledge about generating a situation-specific clin-

7 The VA is in the process of replacing VistA with a commercial HIS offered by Cerner.

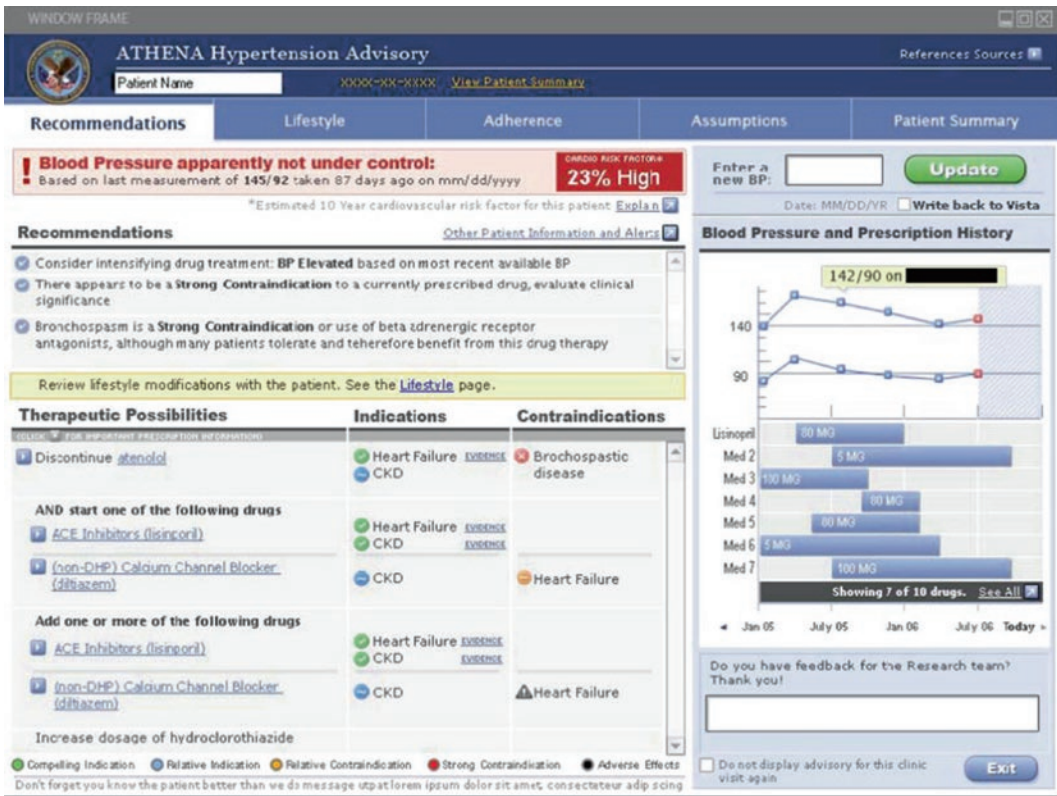
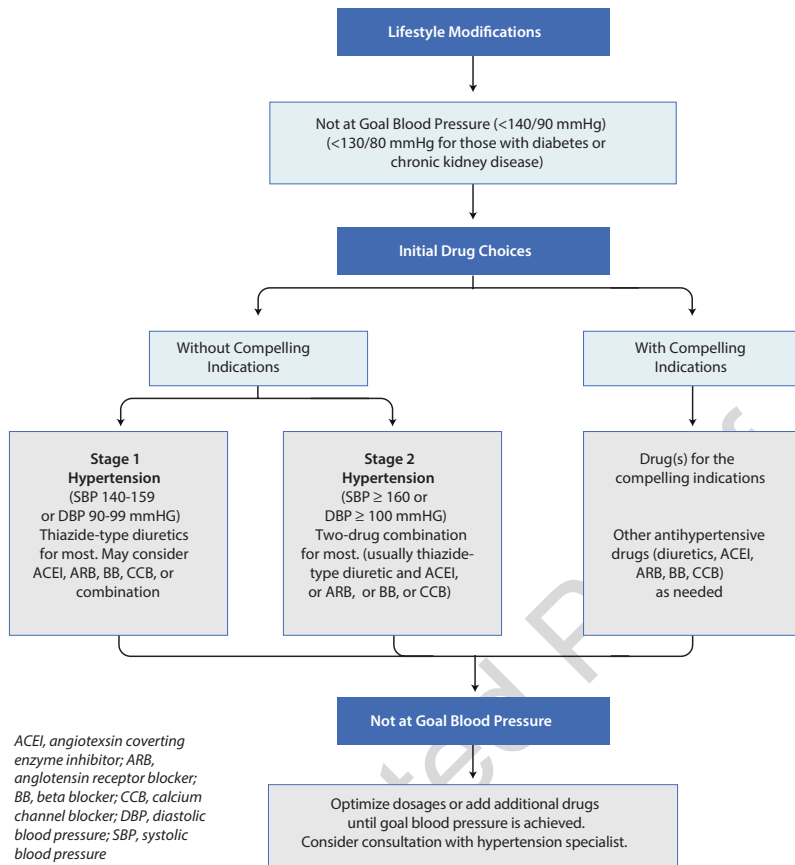


Fig. 24.6 An example of the ATHENA-CDS system interface. ATHENA-CDS provides decision-support for the management of hypertension and several other chronic diseases by using a declarative knowledge base created as an instantiation on a generic guideline ontology. In the screen capture, the provider has entered

the patient's most recent blood pressure, and is offered advice about possible alterations in therapy based on the relevant clinical-practice guideline. The screen image depicts only simulated patient data (Courtesy of M. K. Goldstein, VA Palo Alto Healthcare System)

ical recommendation; Musen et al. 1996). To construct ATHENA-CDS, it was necessary first to define an ontology of clinical practice guidelines (Fig. 24.8). The guideline ontology makes it clear that all guidelines must include *eligibility criteria* that indicate which patients should be treated in accordance with the guideline, a *clinical algorithm* that specifies the sequence of treatments recommended by the guideline, and *guideline drugs* that represent all the medications that patients might be given when their provider follows the guideline. Because the guideline ontology is general, the graph does not contain information about any *particular* clinical algorithm, any *particular* eligibility criteria, and so on. The ontology merely states that all guidelines for management of chronic diseases have such characteristics.

Developers of ATHENA-CDS used the Protégé ontology-development system (Musen 2015) to create the ontology of clinical practice guidelines, which constitutes a knowledge graph. The developers then used Protégé to create subgraphs that represent distinct knowledge bases that define how to manage patients in accordance with particular guidelines. The developers created a knowledge base for management of hypertension reflecting the guideline that is used by the VA and the Department of Defense (DOD), supplemented with recommendations from the Joint National Commission on Hypertension (National High Blood Pressure Education Program 2004; Fig. 24.9). They instantiated the ATHENA-CDS guideline ontology to build a knowledge base for management of congestive heart failure based on the guideline



■ Fig. 24.7 Professional societies, health-care practices, private foundations, and other organizations are all working to capture “best practices” for managing patients in accordance with scientific evidence in terms of clinical practice guidelines. Unfortunately, nearly all these guidelines are published initially as large paper

documents. Here is a high-level, paper-based flowchart from the guideline developed by Joint National Commission on Hypertension. The flowchart summarizes detailed recommendations that the guideline document specifies in many pages of text

developed by the American Heart Association and the American College of Cardiology. The developers built a knowledge base for management of chronic pain, based on the guideline promoted by the VA and the DOD (Trafton et al. 2010). Other knowledge bases for guideline-based care of diabetes, hyperlipidemia, and chronic kidney disease were created in a similar manner.

The ontology-driven approach makes it possible to start with a particular ontology (in this case, one for clinical practice guidelines for management of chronic disease) to create multiple knowledge bases, each one instanti-

ating the ontology to specify the knowledge required for particular guidelines. Similarly, the different knowledge bases can be mapped to different problem-solving programs, such that each problem solver automates a different task associated with guideline-based care (therapy planning, eligibility determination, and so on). The ability to “mix and match” knowledge bases and problem solvers offers considerable flexibility, and it enables developers to reuse elements of previous solutions to address new CDS problems that require different domain knowledge or different problem-solving procedures.

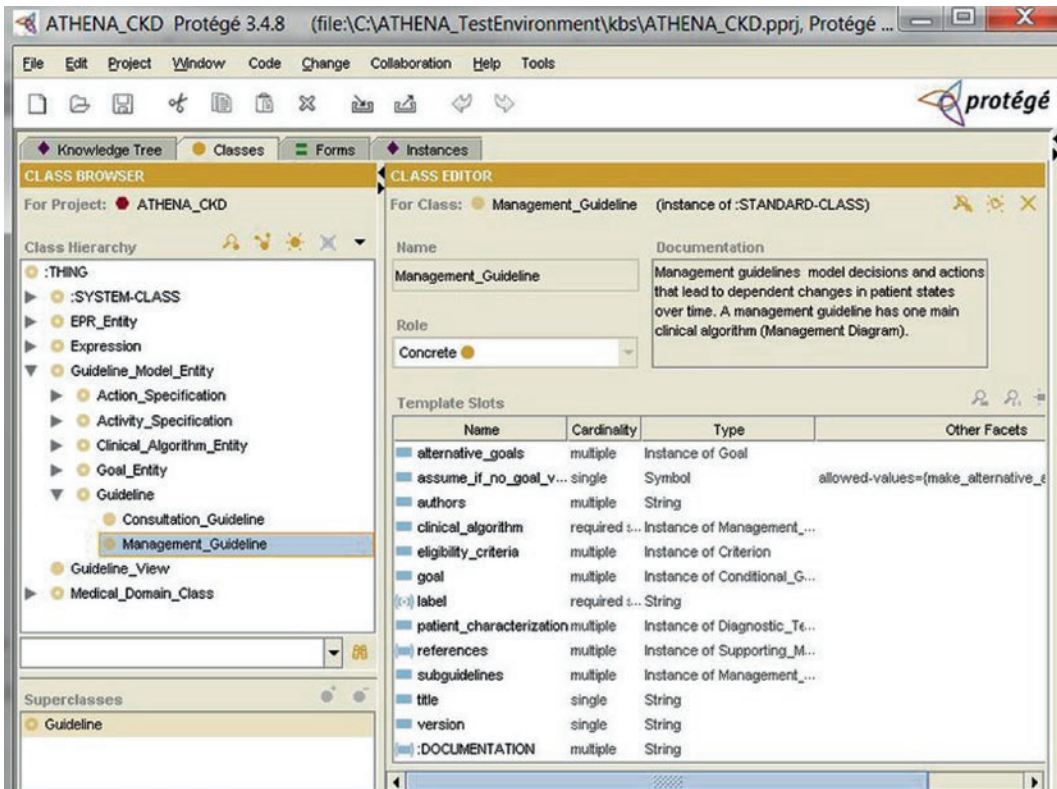


Fig. 24.8 A small portion of the ontology of clinical guidelines used by ATHENA-CDS as entered into the Protégé ontology-editing system. The hierarchy of entries on the left includes entities that constitute building blocks for constructing guideline descriptions. The panel on the right shows the attributes of whatever entity is highlighted on the left. Here, *goal*, *eligibility_criteria*, and *clinical_algorithm*, for example, are attri-

butes of the entity known as *Management_Guideline*. The ontology entered into Protégé reflects concepts believed to be common to all guidelines, but does not include specifications for any guidelines in particular. The complete domain model is used to generate automatically a graphical knowledge-acquisition tool, such as the one shown in Fig. 24.7

24.3.3 Coda

There is no standard way to build a decision-support system. Developers need to make choices, based on the nature of the decision task to be performed, the data and knowledge that are available, and the software tools with which they are most familiar. There are a variety of approaches from which to choose, and each one entails different kinds of trade-offs. A CDSS for providing advice about something potentially as complicated as a clinical practice guideline can be hardcoded in software, implemented as a rule-based system, or driven by knowl-

edge-graph technology. In cases when there is no appropriate evidence-based guideline, a CDSS can use probabilistic approaches or machine learning to suggest reasonable treatment decisions. Although interoperability standards such as SMART-on-FHIR are offering the opportunity to embed custom-tailored CDS technology within proprietary Health IT systems (see Chap. 7), the monolithic nature of most installed systems makes it challenging to achieve this kind of flexibility in the real world. Nevertheless, new IT standards are improving the landscape considerably, as we discuss in the next section.

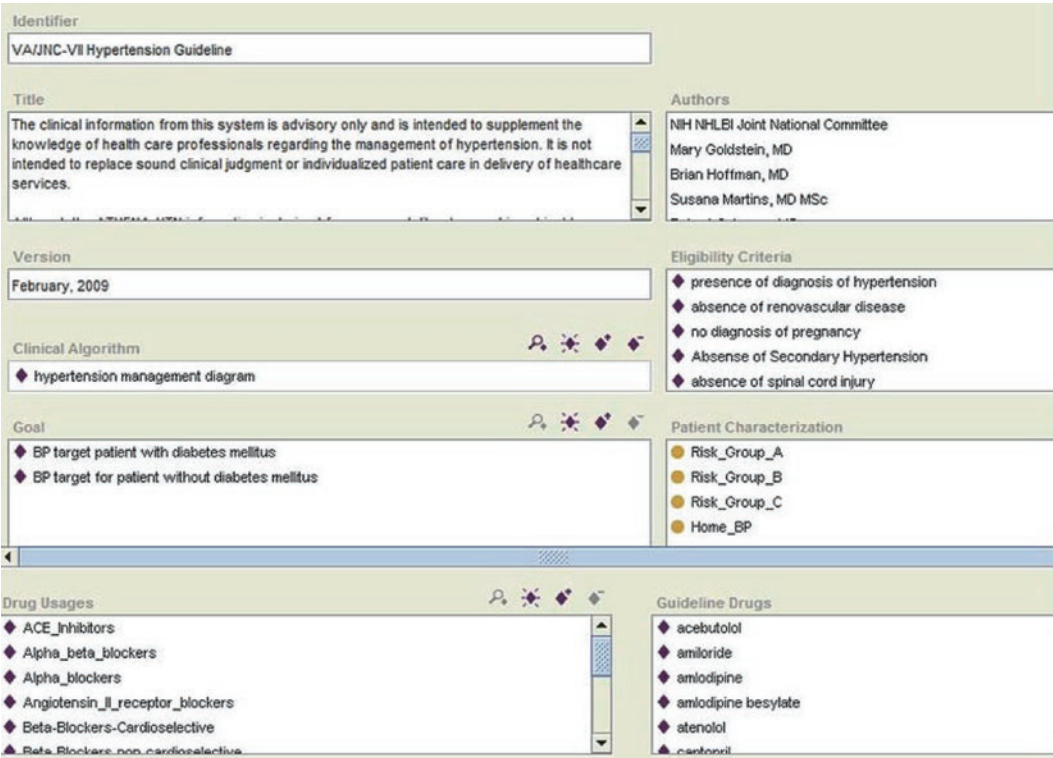


Fig. 24.9 A screen from a Protégé-generated knowledge-acquisition tool for entry of clinical-practice guidelines. The tool is generated automatically from a domain ontology, part of which appears in Fig. 24.7. The entries into the tool specify the knowledge required to treat patients in accordance with the guideline for chronic hypertension adopted by the Department of Veterans Affairs

24.4 Translating CDS to the Clinical Enterprise

Over the past several decades, advanced CDS systems have been developed and deployed in a number of academic medical centers. The technology has subsequently diffused into commercial EHR systems and into routine practice (Chaudhry et al. 2006). The uptake has been greater in medium-to-large hospitals and in medical-center-based networks, including affiliated practices, and has been much less in smaller hospitals, clinics, and independent practices. Although these trends had been sluggish, the “meaningful use” regulations for HIT and other regulatory and incentive-based approaches in the United States as well as elsewhere accelerated the adoption of CDS

technology (Blumenthal and Tavenner 2010; Blumenthal 2010).

In general, the CDS systems deployed to date in vendor EHR systems are quite varied, and relatively limited in scope, and their capabilities in knowledge management are also varied (Wright et al. 2009). The greatest uptake has been in the form of simple alerts and reminders, standard physician order sets, CPOE-based prescription templates with dose checks, allergy checks, identification of drug–lab and drug–drug interactions, and some use of info-buttons or access to context-specific knowledge resources. In some specific settings, rule-based systems have been used to drive the intelligent collection of clinical information in a comprehensive, structured clinical documentation form (Schnipper et al. 2008).

A few vendors have been successful at distributing knowledge resources, making available shareable clinical knowledge in the form of drug-interaction databases, order sets for common indications, rule-based knowledge, documentation templates, and information resources for infobutton-based queries (Middleton et al. 1998). More recent research has examined opportunities for creating knowledge repositories to make these resources more readily available in both the public and private sector (Osheroﬀ et al. 2007; Kawamoto et al. 2013).

Despite growing demand, many years of research and development, and the broad adoption of EHRs in recent years, CDS has had relatively limited adoption to date. There are several reasons for the slow uptake of CDS (Wright et al. 2009), which need to be addressed by approaches such as those enumerated in the sections that follow.

24.4.1 Standard Patient Information Model

CDS rules and other problem-solving approaches need to operate on specific patient data with a clear understanding of the patient data model and semantics of the terms. If those data are stored in a proprietary format and with non-standard encodings, then a set of rules needs to be customized to use data in that form, or the data need to be translated to the information model of the CDS rules, quality measure, or other uses. The customization of rules for each EHR (or EHR implantation) has been the prevailing mode, typified by the “curly braces problem” of Arden Syntax rules, described previously. As a result, vendor EHR systems tend to have libraries of rules that operate only in their own systems, using their proprietary data dictionaries and data models. Knowledge sharing across platforms and systems has been limited, and considerable work is required to integrate vendor HIT products with external CDS systems.

One approach is to develop a canonical information model that both the knowledge source systems, and the knowledge artifacts

can use. This strategy can facilitate a common understanding of semantics in the data and the semantics of the knowledge artifact, and it alleviates mapping CDSS variables to local data or developing a set of custom rules to access the local data in each and every setting. This approach does not eliminate what some describe as an “irreducible mapping problem” in Health IT, but it does distribute it in an effective manner such that, as data representation evolves more toward a canonical form, we can approach **iso-semantic mappings** of data from data source to knowledge artifact—that is, where the semantics of the source data are identical to the semantics of the data expected in the CDS analytic. This approach has been pursued in the very successful Observational Health Data Sciences and Informatics (OHDSI) project in research informatics (Hripscak et al. 2015), which adopts a Common Data Model for source systems to map to, and for analytics (and federated queries) to run against (Jiang et al. 2017a, b).

As discussed below in ► Sect. 26.4.2, in 2012, a **virtual medical record (vMR)** based on the HL7 version 3.0 **Reference Information Model (RIM)** was an initial effort to arrive at the notion of a canonical information model. It was approved by HL7 as a draft standard for trial use for linking dynamically at runtime the arbitrary data elements available in the patient database of an EHR to CDS systems that assume the standard vMR data model for data encoding (Kawamoto et al. 2010). The vMR was designed to serve as an intermediary data model—a canonical form—between proprietary database formats and standards-based CDS systems that developers might plug into any EHR that can make its data available in a vMR-compliant manner. HL7 supported work to map the vMR to standard terminologies and clinical data element definitions. So, for example, data elements in an ECA rule could be referred to in an interoperable manner, rather than, say, the individual mapping of data elements to access methods inside the curly braces of Arden Syntax MLMs (see Rule-Based Systems in ► Sect. 26.3.2).

Newer work focuses on the development of the **Quality Data Model (QDM)** to represent clinical data and concepts used in specifying quality measures, and ultimately clinical decision support logic. The QDM is an information model that describes the relationships between patients and clinical concepts in standardized formats. QDM allows the definition of a clinical concept used in a specification to measure quality of patient care via defined data elements, and it provides the vocabulary needed to relate concepts to each other. For example, QDM, at the highest level of abstraction, defines categories such as Medication, Procedure, Condition, and so on. Within a category, the Datatype definition gives the context of the clinical care process being assessed such as “medication - active,” or “medication - administered.” Further Datatype details are defined with Attributes that can further define the Datatype, or define the expected source for the data. For example, a diagnosis of “Diabetes Mellitus Type I” is an *active diagnosis* datatype, a “Metformin prescription” is a *medication prescribed*, and a “Hemoglobin A1c value” is a *laboratory result*. Each of these elements can be related to terms in controlled terminologies via specification of corresponding code sets. By relating attributes between data elements, the QDM provides a method to construct complex clinical representations both for electronic clinical quality measures and for clinical decision-support logic. Because of this ability to share common logic elements (expressions, value sets, terminology) between quality measure and CDS specifications, QDM has become popular for specifying quality measures among a wide variety of measure developers and CDS implementers (Pathak et al. 2013; Hong et al. 2016).

More recently, the **Fast Healthcare Interoperability Resource (FHIR)** standard from HL7 has emerged from a multi-vendor collaboration in the Argonaut Project (HealthMgt 2015). FHIR is designed specifically for the Web and provides resources and foundations based on common methods and technologies used in the Web (XML, JSON, HTTP, and OAuth) (Bender and Sartipi

2013). FHIR resources are accessed through what is known as a **RESTful API**, which can be defined as one that uses HTTP requests to GET, PUT, POST and DELETE data. FHIR frameworks are built around the concept of resources—basic units of interoperability and modular components that can be assembled into working systems to try to resolve clinical, administrative and infrastructural problems in health care. This capability has partially addressed the heterogeneity of different systems, and it has dramatically improved interoperability between systems.

24.4.2 Adoption of Standard Knowledge-Representation Models

Although Arden Syntax has been an HL7 and ANSI standard since 1999, only a few vendor systems manage their libraries of decision rules using Arden Syntax. Even doing so, of course, the rules still need to be customized for use with vendor-specific patient databases on a tedious, rule-by-rule basis to overcome the curly-braces problem described in ► Sect. 24.3.2. HL7 has pursued the development of more interoperable models not only for data query but also for defining elements of ECA rules. A succession of efforts led to a specification for a query language called GELLO (Sordo et al. 2004), and then the Health eDecisions rule formalism,⁸ and more recently the Clinical Query Language, discussed below.

24.4.2.1 Standards for Encoding Clinical Guideline Models

Particularly challenging in the standards-development world is creation of a standardized, shared model for representing clinical practice guidelines in a form suitable for execution at run time. The **Guideline Element Model (GEM)**; Shiffman et al. (2000)

8 HL7 International (2014). Health eDecisions. Retrieval February 19, 2020: ► https://wiki.hl7.org/index.php?title=Health_eDecisions

is an XML mark-up specification that is an American National Standards Institute (ANSI) standard, now in its third revision, that guideline authors can use to annotate their narrative guidelines to identify key elements for both quality assessment and execution. GEM allows authors to demarcate the text that identifies guideline actions or eligibility criteria, and thus can serve an intermediary purpose in work to transform a prose guideline into a computable specification, but the standard does not itself provide a mechanism to translate a marked-up guideline document into a structure that a computer can interpret and execute. Such systems may be criticized as not explicitly representing the underlying ontology of the guideline components to be used at run time.

Other efforts have focused on creation of a guideline ontology, such as the one adopted by ATHENA-CDS (see Fig. 24.8), that can inform the creation of computer-understandable knowledge bases that are able to capture knowledge about specific guidelines. Such knowledge bases then could allow a CDS system to use knowledge about the guideline, data from the EHR, and information concerning patient preferences and available resources to offer situation-specific, guideline-directed advice. As we have noted, an underlying infrastructure known as EON (Musen et al. 1996) drives the ATHENA-CDS system. Other ontology-based approaches have appeared over the years, including GLIF, GUIDE, PRODIGY, Asbru, and GLARE. Peleg and colleagues (2003) compared many of these guideline models, and they showed significant commonalities among them. Despite the large degree of agreement, however, work in this area has not yet led to anything near a standard that is widely adopted. Part of the problem is that there is wide variation in the structure, granularity, and specificity of existing clinical practice guidelines, making it difficult to develop a single comprehensive and yet readily applicable guideline model. Analysis of the use of guidelines also indicates that guidelines themselves are rarely “executed” without considerable adaptation or localization, except in situations such as protocol-driven care (for example, in clinical trials

or in very specific procedures such as renal dialysis). ATHENA-CDS thus dispenses with offering specific guideline-based recommendations, and instead suggests to the clinicians when certain treatment options might be “compellingly indicated” or “relatively contraindicated.” In highly regimented settings such as the administration of chemotherapy for cancer, however, a CDS system generally would need to be much more “prescriptive” in offering recommendations to clinicians.

24.4.2.2 Standards for Encoding ECA Rules

As we have noted, the ability to share decision rules is hindered in the absence of standards to encode both the logic of the rules (how the IF and the THEN components are to be evaluated) and the clinical data on which the rules depend. Recently, the **Clinical Quality Language (CQL)** has emerged as an expression language that addresses these challenges. CQL is intended to characterize both quality-measure logic and decision-support logic and the data that such logic processes (Odigie et al. 2019). For example, CQL expressions define the expected input data model, library resources that may be called, parameters, value sets, and code sets used in definitions of patient conditions, and additional concepts needed to specify and encode the clinical logic and data types used in a clinical quality measure—or in a CDS rule (Jiang et al. 2015).

Expression languages are commonly used to represent the logic to be used at the presentation layer of an application, and the methods that may be used to interact with standard data models. CQL grew out of prior efforts such as GELLO and the Health eDecisions framework, from work in the U.S. Office of the National Coordinator’s Clinical Quality Framework Initiative. It was focused on identifying, defining, and harmonizing standards and specifications that promote integration and reuse between clinical decision support and clinical quality measurement (CQM) knowledge-representation formalisms. It strives to be more clinician friendly and accessible to subject-matter experts. However, the standards used for the electronic representation of CDS and CQM artifacts have not been

developed in consideration of each other, and the domains use different approaches to the representation of patient data and computable expression logic. Harmonization of these approaches now is focused on clearly identifying the various components involved in the specification of quality artifacts, and then establishing as a principle the notion that they should be treated independently (separation of concerns). Broadly, the components of a CQL artifact involve specifying:

- Metadata – Information about the knowledge artifact (whether CQM or CDS) such as its identifier and version, what health topics it covers, supporting evidence, related artifacts, dependencies, etc.
- Clinical Quality Information – The structure and content of the clinical data (data model) involved in the artifact
- Expression Logic – The actual knowledge and reasoning being communicated by the artifact

The CQL specification is an approved HL7 standard endorsed by the U.S. Centers for Medicare and Medicaid Services, the U.S. Centers for Disease Control and Prevention, and others for the representation of either CDS or CQM knowledge artifacts. It is part of the Clinical Quality Framework effort supported by the Standards and Interoperability Framework of the U.S. Office of the National Coordinator for Health IT. As such, it aims to promote the interoperability of knowledge artifacts through standardization of the data models, logic statements, and controlled medical terminology (including value sets). Because CQL is also fully specified in a machine-readable way, it may be programmatically converted into what HL7 calls an *expression logical model* (ELM), which can then be interpreted in an execution environment.

The convergence of a common formalism that accommodates the use of standardized data models, facilitates clinical query and computation through a library of methods, and which is machine interpretable dramatically improves the portability of clinical knowledge artifacts. In addition, this specification

is designed to be data model-independent, meaning that CQL has no explicit dependencies on any aspect of any particular data model. Rather, the specification allows for any data model to be used, so long as a suitable description of that data model is supplied.

Current efforts now focus on further standardization of input data models, as well as alignment with the FHIR API standard. For example, the QUICK specification and the Quality Improvement Core (QICore) are being developed concurrently with the CQL specification to ensure that the two specifications interoperate effectively. QUICK is a logical model consisting of clinical objects, attributes, and relationships. QUICK provides a uniform way for clinical decision support and quality measures to refer to clinical data. This initiative began in 2013 with the creation of the Quality Improvement Domain Analysis Model (QIDAM), which drew on the vMR and QDM as sources of requirements. Originally, the QUICK data model was developed entirely independently of FHIR. However, recognizing the broader community focus on FHIR, QUICK was aligned, structurally and semantically, as closely as possible to FHIR. This alignment not only creates a common model for quality and interoperability, but also it will make it easier in the future to leverage other FHIR-related efforts, such as Clinical Document Architecture (CDA) on FHIR, or CQL on FHIR. Authors of future quality measures and clinical decision support artifacts may use QUICK, together with the Clinical Quality Language (CQL), to create interoperable and executable knowledge artifacts, thus dramatically the ability to share computable biomedical knowledge artifacts.

24.4.3 Modes of Deployment of CDS

Even with the emergence of shareable, computable, biomedical knowledge artifacts, one of the key impediments to widespread adoption of CDS, particularly the use of rules

and alerts, is clinician annoyance with pop-ups, messages, emails, and other notifications that interrupt workflow. Ideally CDS systems should be integrated into the organization and presentation of information to facilitate workflow and decision making, by anticipating what information is needed for a decision, pre-fetching it, displaying it in ways that support visualization of trends or relationships, and tying these analyses to care plans or actions that can be offered immediately and quickly selected by the user. Order sets, as stated in the beginning of this chapter, form a good example of use of CDS both to suggest appropriate actions in a given setting and to make it easy to accomplish those actions, by immediately enabling the orders in the set to be entered automatically into the EHR, perhaps with modification.

There is much ongoing research to develop methods for managing the processes of data capture, data presentation, data visualization, and selection of actions, but this work is usually being done outside of vendor EHRs. Given limited interoperability and access to the internals of proprietary systems, this kind of experimentation is now tending to take place in the form of apps and services that operate on externally extracted data (Mandl and Kohane 2012). There is growing support for the notion of SMART apps—Substitutable Medical Applications and Reusable Technologies—that use standard API methods to access electronic health record data, perform clinical inferences external to the EHR, and return insights in a standardized container (whether iFrame, Web page, or free-standing application; Mandel et al. 2016).

24.4.4 Workflow and Setting-Specific Factors

As noted in ► Sect. 24.3.2, applications based on single-step situation–action rules are among the most prevalent and useful types of CDS systems. Such systems can be invoked in many contexts to provide either recommenda-

tions in real time or reminders or alerts that are processed in batch, based on time-oriented triggers or data-evaluation events. Rules can invoke other knowledge resources—providing new information content, triggering other rules, or offering order sets.

Rule content is ideally based on analysis of clinical evidence, such as recommendations or guidelines emanating from the U.S. Preventive Services Task Force, or from professional-society studies of best practices for specific diseases. The job of formalizing these recommendations into executable logic requires that they be expressed in a specified way, but even having done so, such rules are not typically ready to execute in a particular environment, even if they are expressed in a rule execution language “understood” by an EHR system, and if they refer to the data elements in the EHR in their expected format. The reason the rules are not readily executable is also the reason that rules that work well in one environment are often not able to be successfully deployed elsewhere without substantial modification (even if in the same representation format and if using the same data model).

The reason for the failure is lack of adaptation to what we refer to as *setting-specific factors* (SSFs; Greenes et al. 2010). To work effectively, rules need to integrate well with the clinical setting, workflow, users, application environment, and other factors. These requirements are reflected in how and when the rule should be triggered—on various events such as examination of some element of the EHR, on login to the system, or on the availability of laboratory test results. Rules may also be developed in the form of reminders that are triggered when the CDSS evaluates on a batch basis a practice’s list of patients to be seen on a given day, the patients who have a birthday in a given month, the passage of a specific interval of time since a previous comparison event, and so on. The rules additionally may vary based on the practice setting (e.g., the emergency department, an office practice, or an inpatient unit); particular inclusion or exclusion criteria or threshold

modifications that may be site-specific; how the recommendation should be transmitted (e.g., via electronic mail, popup windows, or sidebar messages); whether the recommendation requires acknowledgment by the recipient; whether it can be overridden; whether the alert should be escalated to supervising clinicians, and so on. Rules that have been custom tailored in such ways by means of executable code naturally are less sharable than are generic rules. Failure to capture the kinds of customizations that are needed, however, makes it time consuming for individual sites to adapt generic medical recommendations to their particular requirements or to capitalize on the experiences of others. What is needed is a way to represent useful experience in terms of SSF combinations that work, without needing to do so at the level of detailed code that is difficult for users to visualize and modify.

24.4.5 Sharing of Best-Practice Knowledge for CDS

The methods described above for sharing computable biomedical knowledge are now gaining momentum (viz., SMART, CQL, FHIR). Historically, it generally has fallen on each health-care organization, user group, or other entity to undertake its own process of identifying and managing the best-practice knowledge it wants to deploy in its CDS systems. Most institutions lack the expertise, or the resources, to accomplish this task. Even having a national or international repository of such knowledge would not preclude the need for customization, but it would certainly make it easier for each health-care entity to start with a trusted source. Over the past decade, the U.S. Agency for Healthcare Quality and Research has funded efforts to create such a public repository, known as CDS Connect (Lomatan et al. 2019). In CDS Connect's repository, the goal is to archive knowledge artifacts represented in the CQL formalism to promote sharing and interoperability. Where such a repository should be hosted, how it might integrate public and pri-

vate knowledge sources, who would have oversight over it, how its knowledge would be peer reviewed and quality-rated, and how it would be sustained are among the many questions that have not yet been answered, but this is an area of intense research and development. While many health-care organizations continue to perform this kind of knowledge-curation work for their own constituencies, several initiatives show a clear pathway to becoming viable alternatives for knowledge aggregation and dissemination.

24.5 Future Research and Development for CDS

Workers in biomedical informatics have studied problems in assisting with complex decision making for more than half a century. It seems that it is only now, with the very recent adoption of HIT on a widespread basis, that the foundations are finally in place for the rapid advance of CDS technology in clinical settings. Although considerable logistical problems still must be surmounted as outlined in ► Sect. 24.4, this is an exciting time in which to study CDS and its translation from the laboratory to the point of care.

24.5.1 Standards Harmonization for Knowledge Sharing and Implementation

Many implementation challenges remain for the broad adoption and effective use of CDS in EHR systems. As mentioned above, one of the most active areas of current research focuses on development of standard approaches to knowledge sharing for CDS. Knowledge sharing may take the form of human-readable artifacts, machine-interpretable artifacts, or executable Web services (Osheroff et al. 2007; Goldberg et al. 2014; Dixon et al. 2013; Kawamoto et al. 2013). A capability for CDS sharing, as well as CDS functionality itself, would be substantially facilitated by the continued develop-

ment and use of common standards designed to serve CDS needs in health care.

As noted, several standards currently exist that are aimed at specific areas of CDS and types of CDS artifacts, or that could be leveraged to benefit CDS. For example, the Clinical Decision Support Consortium, a large collaborative research and development group supported by the U.S. Agency for Healthcare Research and Quality, adopted an enhanced version of the Continuity of Care Document (CCD) to serve as the foundation for input data for multi-institutional trials of CDS technology (Middleton 2009). When taking advantage of the most current standards such as CQL and FHIR, systems developers tend to adopt not only the standards but also arrive at more common implementation approaches or patterns, that promote knowledge sharing and reuse, and which may decrease the implementation burden if EHR vendors accommodate the standardized API and interaction models, as well as the data model expectations of the standards.

24.5.2 Context-based Knowledge Selection

Much of the above effort is aimed at overcoming the non-portability of event-condition-action (ECA) rules, such as alerts and reminders, because of the need to localize and adapt triggering conditions, modes of interaction, and specific workflow processes of sites (Setting-specific factors, or SSFs, as discussed in ► Sect. 24.4.4), leading to many variations that must be laboriously designed and tracked.

One of the possible modes for reducing this need for customization and localization is to use the context, state, and activity (CSA) of a CDSS user to automatically identify appropriate knowledge artifacts to be made available. The idea relies on maintaining patient state, user role, expertise, setting, and specific tasks being performed to determine when the knowledge might be pertinent. The latter also requires a rich multi-axial set of metadata for the knowledge artifact repository that allow selection based on user CSA.

Other ways in which context could be used might be to create specialized views or filters of available data, based on CSA. For example, selection of data to be viewed could be based on a user's role and domain expertise. Associations (and, ideally, typed relations) among data items could be highlighted—for example, among problems, findings, and actions. Such relations could also potentially be used to anticipate assessment and plan entries in notes based on data items present or being focused on. Such approaches undoubtedly will only be scalable in generalized web-services implementations. For example, infobutton managers already use context such as user type, app/function being performed, and specific item being evaluated to identify retrieval keys for external knowledge resources. The idea behind the expansion of context as a mode for invocation of CDS more generally is to create a more detailed and continually updated model of context, state, and activity. Such efforts are just getting underway.

24.5.3 Representation Models

To date, standards and related efforts addressing CDS have heavily emphasized specific CDS execution methods and the representation of the clinical context of the patient. For example, as we have noted previously, a variety of frameworks for working with rules, including Arden Syntax, Drools, JESS, CQL, along with several proprietary formats, have worked their way into vendor offerings. This diversity has inhibited the exchange of best-practice knowledge to date, but progress toward more effective knowledge interchange is being made with CQL in particular. The current situation is that both public and private knowledge repositories of knowledge artifacts that address specific pieces of the CDS problem exist (e.g., NLM Value sets, CMS eCQMs, standardized data models, professional society clinical pathways, vendor implementations of algorithms and analytics, and SMART applications) and we see growing adoption given the increased pressures on clinical reasoning and operations in practice.

In addition to reducing unwarranted clinical variability in practice, future work needs to establish a canonical patient information model with a formal ontology, an event model for triggering conditions, an action model for CDS intervention recommendations, a workflow model for appropriately inserting CDS interventions into the routines of clinical practice, a knowledge-representation schema with a standard regular expression language, and, ideally, a measurement standard to assess CDS performance in use.

24.5.4 Externalizing CDS

Standardization of methods for externalized CDS (CDS performed outside of the EHR, in the cloud) is becoming more commonplace due to emerging standards described above (CQL, CQM, QUICK, FHIR). Prior research and development efforts have demonstrated the feasibility of accessing clinical data from the electronic health record via standards-based (viz., FHIR) and proprietary RESTful APIs, and running executable knowledge artifacts – both quality measures, and clinical decision support – on the data in the secure cloud environment and returning insights into the app being used in the clinical workflow (Wright et al. 2015; Dixon et al. 2013). Current work is focused on representing all of the requisite knowledge artifacts required to create a computable practice guideline fully represented in CQL, and on running those knowledge artifacts via FHIR data-access methods. Use of services enables considerable flexibility and breadth of kinds of CDS that can be made available widely, and ongoing work with FHIR resource specifications is improving the ability to gather the data needed and deliver it, and to integrate with clinical workflow processes more smoothly.

We have described the many efforts underway to standardize data models for information exchange, CQL for knowledge representation, and functional integration with EHR workflows using SMART, and FHIR. A deeper functional integration model between externalized services and EHR work-

flow is provided by **CDS Hooks**. CDS Hooks (Dolin et al. 2018) is a method for enabling an event in the host system to determine when an app (such as provided by SMART-on-FHIR) should be launched programmatically, thus complementing the need to launch apps directly by users. The EHR detects an event such as a physician beginning to write an order, and it then can invoke an external decision-support service. That service can determine what task is being performed and return information in the form of a “card” (a phrase or text snippet containing an inference or assessment, or suggested action) that will be displayed within the EHR. CDS Hooks may also provide a link to an external app. The main downside of these approaches to CDS is that, although they have formal methods for launching apps, they basically have no constraints on what happens within the apps, or support for linking to the intrinsic functionality and workflows of a host EHR. Thus, the methods can result in a proliferation of SMART apps and CDS Hooks modules in an organization’s library, with a number of them that may be of limited utility. This situation could cause significant challenges for an enterprise seeking to manage and update its CDS capabilities on a regular basis.

24.5.5 Usability Research and CDS

The use of CDS within EHRs, and health IT in general, have been identified as double-edged swords: technology may provide benefit, but it also may cause considerable harm. Clinician error when using information systems that may result in untoward outcomes and unintended consequences (Karsh et al. 2010; Sittig and Singh 2009) may be an emerging property that is demonstrated only after system implementation or widespread use. Medical errors related to use of Health IT are problematic, not only for clinical and quality of care reasons, but technically, since they may represent a mismatch between the user’s model of the task being performed and the model used in a computation (National Research Council (U.S.) Committee on Engaging the Computer

Science Research Community in Health Care Informatics et al. 2009), a mismatch between the application's intended functionality and the resulting action or event (Harrison et al. 2007), or a latent health IT-related error yet to happen (Ash et al. 2007). Excessive alert fatigue can undermine the efficacy of clinical decision support in CPOE (Isaac et al. 2009; Strom et al. 2010), and in other IT functions (Chused et al. 2008), and result in very high user override rates (Shah et al. 2006; van der Sijs et al. 2006; Weingart et al. 2003). Critical research questions need to focus on the potential mismatch between the user's mental model or intent and the application design, use case, or workflow model (Zhang and Walji 2011; Patel et al. 2010). Further attention needs to be given to basic principles of human-factors engineering, such as the use of colors and layout within the application interface. Additional questions remain regarding the ideal design of methods and controls with which a user might interact to choose a medication from a long list, or identify and encode patient problems. More advanced research will enable visualization and decision making by matching problems with care plans, and facilitation of continuity and coordination of care based on underlying CDS rules and guideline-based workflows. Especially challenging is addressing the need for structured data to support clinical decision support and quality reporting, in a manner that is efficient for the end-user. This goal might be achieved by combining structured documentation during data entry, and natural language processing for data abstraction from the clinical narrative. Most important, however, are methods to direct CDS to the right user, at the right time, in the right workflow, with the right level of alerting or intervention, and the right information (Osheroff et al. 2012).

24.5.6 Data-Driven CDS

As we have discussed, a major area of research in informatics concerns methods for deriving knowledge from large data sets using a variety of techniques. With the adoption of health IT broadly, investigators are drawing on large-

scale data-mining methods to provide CDS for population monitoring, public health surveillance, and even to offer patient-specific recommendations based on cohort data when there is no specific evidence that could otherwise guide therapy. With the increasing availability of data from diverse sources relevant to patient care, large data sets may be created and used for both discovery of previously unknown associations, and novel clinical predictions (Frankovich et al. 2011; Longhurst et al. 2014). Critical research questions here will include how to define like cohorts of patients, how to structure and frame the index decision, what methods to use to assess the likelihood of alternate prediction scenarios, and how to model and elicit the patient's preferences for each scenario. The Institute of Medicine (2011b) articulated a long-term vision for a Learning Health System, in which clinical and administrative data of all kinds will begin to inform and enhance clinical practice on a national level in a wide variety of ways.

24.6 Conclusions

The future of CDS systems inherently depends on progress in developing useful computer programs and in reducing logistical barriers to implementation. Although ubiquitous computer-based decision aids that routinely assist clinicians in most aspects of their practice are currently the stuff of science fiction, progress has been real and the potential remains inspiring. Early predictions about the effects that such innovations would have on medical education and practice have not yet come to pass (Schwartz 1970), but growing successes support an optimistic view of what technology will eventually do to assist practitioners with the processing of complex data and knowledge. The research challenges have been identified much more clearly, legislative mandates are creating not only new financial incentives but also the practical substrate of increased EHR adoption and convergence toward data interoperability, and the implications for health-science education are much better understood. The basic com-

puter literacy of health professional students can be generally assumed, but health-science educators now must teach the conceptual foundations of biomedical informatics if their graduates are to be prepared for the technologically sophisticated world that lies ahead.

Equally important, we have learned much about what is not likely to happen. The more that investigators understand the complex and changing nature of medical knowledge, the clearer it becomes that trained practitioners of biomedical informatics will always be required as participants in fostering a cooperative relationship between physicians and computer-based decision tools. There is no evidence that machine capabilities will ever equal the human's ability to deal with unexpected situations, to integrate visual and auditory data that reveal subtleties of a patient's problem, to work with patients to incorporate their values and priorities in care plans, or to deal with social and ethical issues that are often key determinants of proper medical decisions. Considerations such as these will always be important to the humane practice of medicine, and practitioners will always have access to information that is meaningless to the machine. Such observations argue cogently for the discretion of health-care workers in the proper use of decision-support tools.

? Suggested Readings

Bright, T. J., Wong, A., Dhurjati, R., Bristow, E., Bastian, L., Coeytaux, R. R., Samsa, G., Hasselblad, V., Williams, J. W., Musty, M. D., Wing, L., Kendrick, A. S., Sanders, G. D., & Lobach, D. (2012). Effect of clinical decision-support systems: A systematic review. *Annals of Internal Medicine*, 157(1), 29–43 This thorough analysis of studies of CDS systems demonstrates that there is good evidence that CDS technology can alter clinician behavior in positive ways, but that evidence that CDS systems can improve long-term patient outcomes is still inconclusive. The paper is also useful for its comprehensive bibliography.

Greenes, R. A. (Ed.). (2014). *Clinical Decision Support*, 2nd Edition: The Road to Broad

Adoption. New York: Elsevier This book offers a comprehensive discussion of the nature of medical knowledge and of information technology to assist with medical decision making. It provides detailed discussions of the computational, organizational, and strategic challenges in the design, development, and deployment of CDS systems.

Institute of Medicine. (2011). *Digital infrastructure for the learning health system: The foundation for continuous improvement in health and healthcare*. Workshop Series Summary. Washington, DC: The National Academies Press. This monograph summarizes the vision for a national Learning Health System and offers the perspective of a wide range of thought leaders on the work required to achieve that vision.

Ledley, R., & Lusted, L. (1959). Reasoning foundations of medical diagnosis. *Science*, 130, 9–21 This is the paper that started it all. This classic article provided the first influential description of how computers might be used to assist with the process of diagnosis. The flurry of activity applying Bayesian methods to computer-assisted diagnosis in the 1960s was largely inspired by this provocative paper.

Sittig, D. F., Wright, A., Osheroff, J. A., Middleton, B., Teich, J. M., Ash, J. A., Campbell, E., & Bates, D. W. (2008). Grand challenges in clinical decision support. *Journal of Biomedical Informatics*, 41(2), 387–392 A rank-ordered list of some of the principal challenges for CDS technology development and implementation, intended “to educate and inspire researchers, developers, funders, and policy makers”.

? Questions for Discussion

1. Some researchers in medical AI have argued that CDS systems should reason from clinical data in a way that closely matches the reasoning strategies of the very best clinical experts, as such experts are the most clever diagnosticians and the most experienced treatment specialists that there are. Other researchers maintain that expert reasoning, no matter how excellent, is

- at some level inherently flawed, and that CDS systems must be driven from the mining of large amounts of solid data. How do you account for the apparent difference between these views? Which view is valid? Explain your answer.
2. Transitioning CDS systems from one clinical setting to another has always been problematic. The Leeds Abdominal Pain System was installed in several major clinical settings, and yet the system never performed as well elsewhere as it had done in Leeds. The Arden Syntax, created expressly to facilitate knowledge sharing across institutions, failed to meet this goal to a significant degree. Why kinds of setting-specific factors make it difficult to transplant decision-support technology from one environment to another? What kinds of research might lead to better methods for knowledge sharing in the future?
 3. In one early evaluation study, the decision-support system ONCOCIN provided advice concerning cancer therapy that was approved by experts in only 79% of cases (Hickam et al. 1985). In another study, the HyperCritic CDS system for the management of hypertension offered the same comments that were generated by a panel of experts in only 45% of cases (Van der Lei et al. 1991). Even today, such system performance is fairly typical for computer programs that suggest patient therapy. Do you believe that this performance is adequate for a computational tool that is designed to help physicians to make decisions regarding patient care? What problems might CDS systems encounter as their developers attempt to make the systems more comprehensive in the advice that they offer? Why might it be more difficult for computer systems to offer acceptable recommendations for patient therapy than seems to be the case for diagnosis? What safeguards, if any, would you suggest to ensure the proper use of such systems? Would you be willing to visit a particular physician if you knew in advance that she made decisions regarding treatment that were approved by expert colleagues less than 80% of the time? If you would not, what level of performance would you consider adequate? Justify your answers.
 4. A large international organization once proposed to establish an independent laboratory—much like Underwriters Laboratory in the United States—that would test CDS systems from all vendors and research laboratories, certifying the effectiveness and accuracy of those systems before they might be put into clinical use. What are the possible dimensions along which such a laboratory might evaluate decision-support systems? What kinds of problems might such a laboratory encounter in attempting to institute such a certification process? In the absence of such a credentialing system for CDS systems, how can health-care workers feel confident in using a clinical decision aid?
 5. There is considerable untapped potential for CDS to help in managing patients with multiple complex conditions. What are the challenges in dealing with such patients, and how can CDS be helpful? What are the features required of an algorithm that might integrate recommendations from the multiple clinical-practice guidelines that a CDS system could apply?
 6. CDS is often implemented poorly, resulting in dissatisfaction, if not outright annoyance. What are the human factors that need to be taken into consideration in implementing CDS effectively? Discuss issues and approaches to enhancing usability. What are situations in which graphics and visualization might be used? How can CDS be used to enhance rather than to impede workflow? What are strategies to help avoid unintended consequences of poorly implemented CDS?

References

- Adler-Milstein, J., Bu, D., & Pan, E. (2007). The cost of information technology-enabled diabetes management. *Disease Management*, 10(3), 115–128.
- Adler-Milstein, J., Embi, P. J., Middleton, B., et al. (2017). Crossing the health IT chasm: Considerations and policy recommendations to overcome current challenges and enable value-based care. *Journal of the American Medical Informatics Association*, 24(5), 1036–1043.
- Alamri, Y., Frizelle, F., Al-Mahrouqi, H., et al. (2016). Surgical ward round checklist: Does it improve medical documentation? A clinical review of Christchurch general surgical notes. *ANZ Journal of Surgery*, 86(11), 878–882.
- Algaze, C. A., Wood, M., Pageler, N. M., et al. (2016). Use of a checklist and clinical decision support tool reduces laboratory use and improves cost. *Pediatrics*, 137(1), e20143019.
- Ash, J. S., Sittig, D. F., Poon, E. G., et al. (2007). The extent and importance of unintended consequences related to computerized provider order entry. *Journal of the American Medical Informatics Association: JAMIA*, 14(4), 415–423. <https://doi.org/10.1197/jamia.M2373>.
- Avanti, A., Jung, K., Harman, S., et al. (2018). Improving palliative care with deep learning. *BMC Medical Informatics and Decision Making*, 18(Suppl 4), 122.
- Baars, M. J. H., Henneman, L., & Ten Kate, L. P. (2005). Deficiency of knowledge of genetics and genetic tests among general practitioners, gynecologists, and pediatricians: A global problem. *Genetics in Medicine*, 7(9), 605.
- Barnett, G. O., Cimino, J. J., Hupp, J. A., et al. (1987). *Journal of the American Medical Association*, 258(1), 67–74. PMID: 32953167.
- Baron, R. J. (2010). What's keeping us so busy in primary care? A snapshot from one practice. *The New England Journal of Medicine*, 362(17), 1632–1636. <https://doi.org/10.1056/NEJMon0910793>.
- Bates, D. W., & Gawande, A. A. (2003). Improving safety with information technology. *New England Journal of Medicine*, 348(25), 2526–2534. <https://doi.org/10.1056/NEJMSa020847>.
- Beasley, J. W., Wetterneck, T. B., Temte, J., et al. (2011). Information chaos in primary care: Implications for physician performance and patient safety. *Journal of American Board of Family Medicine*, 24(6), 745–751.
- Bender, D., & Sartipi, K. (2013). HL7 FHIR: An Agile and RESTful approach to healthcare information exchange. *2013 IEEE 26th international symposium on computer-based medical systems (CBMS)*, IEEE, pp. 326–331.
- Bleich, H. (1972). Computer-based consultation: Electrolyte and acid-base disorders. *American Journal of Medicine*, 53, 285–291.
- Bloomrosen, M., Starren, J., Lorenzi, N. M., et al. (2011). Anticipating and addressing the unintended consequences of health IT and policy: A report from the AMIA 2009 Health Policy Meeting. *Journal of the American Medical Informatics Association*, 18(1), 82–90.
- Blumenthal, D. (2009). Stimulating the adoption of health information technology. *New England Journal of Medicine*, 360(15), 1477–1479. <https://doi.org/10.1056/NEJMp0901592>.
- Blumenthal, D. (2010). Launching HITECH. *New England Journal of Medicine*, 362(5), 382–385.
- Blumenthal, D., & Glaser, J. P. (2007). Information technology comes to medicine. *New England Journal of Medicine*, 356(24), 2527–2534. <https://doi.org/10.1056/NEJMHpr066212>.
- Blumenthal, D., & Tavenner, M. (2010). The “meaningful use” regulation for electronic health records. *New England Journal of Medicine*, 363(6), 501–504. <https://doi.org/10.1056/NEJMp1006114>.
- Bodenreider, O., & Stevens, R. (2006). Bio-ontologies: Current trends and future directions. *Briefings in Bioinformatics*, 7(3), 256–274. <https://doi.org/10.1093/bib/bbl027>.
- Bright, T. J., Wong, A., Dhurjati, R., et al. (2012). Effect of clinical decision-support systems: A systematic review. *Annals of Internal Medicine*, 157(1), 29–43. <https://doi.org/10.1059/0003-4819-157-1-201207030-00450>.
- Bu, D., Pan, E., Walker, J., et al. (2007). Benefits of information technology-enabled diabetes management. *Diabetes Care*, 30(5), 1137–1142.
- Buchanan, B. G., & Shortliffe, E. H. (1984). *Rule-based expert systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*. Reading: Addison-Wesley.
- Chan, I. S., & Ginsburg, G. S. (2011). Personalized medicine: Progress and promise. *Annual Review of Genomics and Human Genetics*, 12, 217–244. <https://doi.org/10.1146/annurev-genom-082410-101446>.
- Chaudhry, B., Wang, J., Wu, S., et al. (2006). Systematic review: Impact of health information technology on quality, efficiency, and costs of medical care. *Annals of Internal Medicine*, 144(10), 742–752.
- Chused, A. E., Kuperman, G. J., & Stetson, P. D. (2008). Alert override reasons: A failure to communicate. *Proceeding of the AMIA Annual Symposium*, pp. 111–115.
- Cimino, J. J., Li, J., Bakken, S., & Patel, V. (2002). Theoretical, empirical and practical approaches to resolving unmet information needs of clinical information system users. *Proceeding of the AMIA Annual Symposium*, pp. 170–174.
- Clancey, W. J. (1983). The epistemology of a rule-based system: A framework for explanation. *Artificial Intelligence*, 20(3), 215–251.
- Clancy, C. M., Anderson, K. M., & White, P. J. (2009). Investing in health information infrastructure: Can it help achieve health reform? *Health Affairs*, 28(2), 478–482. <https://doi.org/10.1377/hlthaff.28.2.478>.
- Colombet, I., Bura-Rivière, A., Chatila, R., et al. (2004). Personalized versus non-personalized computerized decision support system to increase therapeutic qual-

- ity control of oral anticoagulant therapy: An alternating time series analysis. *BMC Health Services Research*, 4(1), 27. <https://doi.org/10.1186/1472-6963-4-27>.
- Covell, D., Uman, G., & Manning, P. R. (1985). Information needs in office practice: Are they being met? *Annals of Internal Medicine*, 103(4), 596–599.
- De Dombal, F. T., Leaper, D. J., Staniland, J. R., McCann, A. P., & Horrocks, J. C. (1972). Computer-aided diagnosis of acute abdominal pain. *British Medical Journal*, 1, 376–380.
- Decker, S. L., Jamoom, E. W., & Sisk, J. E. (2012). Physicians in nonprimary care and small practices and those age 55 and older lag in adopting electronic health record systems. *Health Affairs*, 31(5), 1108–1114. <https://doi.org/10.1377/hlthaff.2011.1121>.
- DeClerq, P. A., Blom, J. A., Korsten, H. H., et al. (2004). Approaches for creating computer-interpretable guidelines that facilitate decision support. *Artificial Intelligence in Medicine*, 31(1), 1–27.
- DesRoches, C. M., Worzala, C., Joshi, M. S., et al. (2012). Small, nonteaching, and rural hospitals continue to be slow in adopting electronic health record systems. *Health Affairs*, 31(5), 1092–1099. <https://doi.org/10.1377/hlthaff.2012.0153>.
- Dixon, B. E., Simonaitis, L., Goldberg, H. S., et al. (2013). A pilot study of distributed knowledge management and clinical decision support in the cloud. *Artificial Intelligence in Medicine*, 59(1), 45–53.
- Dolin, R. H., Boxwalla, A., & Shalaby, J. (2018). A Pharmacogenomics clinical decision support service based on FHIR and CDS Hooks. *Methods of Information in Medicine*, 57(S 02), e115–e123.
- Engel, G. L. (1977). The need for a new medical model: A challenge for biomedicine. *Science*, 196, 129–136.
- Fargher, E. A., Eddy, C., Newman, W., et al. (2007). Patients' and healthcare professionals' views on pharmacogenetic testing and its future delivery in the NHS. *Pharmacogenomics*, 8(11), 1511–1519. <https://doi.org/10.2217/14622416.8.11.1511>.
- Flexner, A. (1910). *Medical education in the United States and Canada: A report to the Carnegie Foundation for the Advancement of Teaching*, Bulletin No. 4., New York City: The Carnegie Foundation for the Advancement of Teaching, OCLC 9795002.
- Frankovich, J., Longhurst, C. A., & Sutherland, S. M. (2011). Evidence-based medicine in the EMR era. *The New England Journal of Medicine*, 365(19), 1758–1759.
- Gawande, A. (2009). *The checklist manifesto: How to get things right*. New York: Metropolitan Books.
- Ginsburg, G. S., & Willard, H. F. (2009). Genomic and personalized medicine: Foundations and applications. *Translational Research*, 154(6), 277–287. <https://doi.org/10.1016/j.trsl.2009.09.005>.
- Goldberg, H. S., Paterno, M. D., Rocha, B. H., et al. (2014). A highly scalable, interoperable clinical decision support service. *Journal of the American Medical Informatics Association*, 21(e1), e55–e62.
- Goldstein, M. K., Hoffman, B. B., Coleman, R. W., et al. (2000). Implementing clinical practice guidelines while taking account of evidence: ATHENA, an easily modifiable decision-support system for management of hypertension in primary care. In *Proceedings of the AMIA Fall Symposium* (pp. 300–304). Philadelphia: Hanley & Belfus.
- Gorry, G. A., & Barnett, G. O. (1968). Sequential diagnosis by computer. *JAMA*, 205(12), 849–854.
- Greenes, R., Bloomrosen, M., Brown-Connolly, N. E., et al. (2010). The Morningside Initiative: Collaborative development of a knowledge repository to accelerate adoption of clinical decision support. *The Open Medical Informatics Journal*, 4, 278–290.
- Greenes, R. A., Ed. (2014). *Clinical Decision Support: The Road to Broad Adoption*. New York, Elsevier.
- Greenes, R. A., Bates, D. W., Kawamoto, K., et al. (2018). Clinical decision support models and frameworks: Seeking to address research issues underlying implementation successes and failures. *Journal of Biomedical Informatics*, 78, 134–143.
- Grimm, R. H., Shimoni, K., Harlan, W. R., et al. (1975). Evaluation of patient-care protocol use by various providers. *New England Journal of Medicine*, 282(10), 507–511.
- Harrison, M. I., Koppel, R., & Bar-Lev, S. (2007). Unintended consequences of information technologies in health care—an interactive sociotechnical analysis. *Journal of the American Medical Informatics Association*, 14(5), 542–549. <https://doi.org/10.1197/jamia.M2384>.
- Haynes, R. B., Wilczynski, N. L., & Computerized Clinical Decision Support System (CCDSS) Systematic Review Team. (2010). Effects of computerized clinical decision support systems on practitioner performance and patient outcomes: Methods of a decision-maker-researcher partnership systematic review. *Implementation Science*, 5(1), 12. <https://doi.org/10.1186/1748-5908-5-12>.
- HealthMgt. (2015). HL7 launches Argonaut Project to advance FHIR interoperability standard. *Health Management Technology*, 36(2), 26.
- Heckerman, D., & Horvitz, E. (1986). The myth of modularity in rule-based systems for reasoning with uncertainty. In J. Lemmer & L. Kanal (Eds.), *Uncertainty in artificial intelligence 2*. Amsterdam: North Holland.
- Hickam, D. H., et al. (1985). “The treatment advice of a computer-based cancer chemotherapy protocol advisor.” *Ann Intern Med*. 103(6 (Pt 1)): 928–936.
- Hoffman, M. A., & Williams, M. S. (2011). Electronic medical records and personalized medicine. *Human Genetics*, 130(1), 33–39. <https://doi.org/10.1007/s00439-011-0992-y>.
- Hong, N., Li, D., Yu, Y., et al. (2016). A computational framework for converting textual clinical diagnostic criteria into the quality data model. *Journal of Biomedical Informatics*, 63, 11–21.
- Hripcsak, G., Duke, J. D., Shah, N. H., et al. (2015). Observational Health Data Sciences and Informatics (OHDSI): Opportunities for observational research-

- ers. In I. N. Sarkar et al. (Eds.), *Proceedings of Medinfo 2015*. IOS Press. <https://doi.org/10.3233/978-1-61499-564-7-574>.
- Institute of Medicine. (2011a). *Health IT and patient safety: Building safer systems for better care*. Washington, DC: The National Academies Press.
- Institute of Medicine. (2011b). *Digital infrastructure for the learning health system: The foundation for continuous improvement in health and healthcare. Workshop Series Summary*. Washington, DC: The National Academies Press.
- Isaac, T., Weissman, J., Davis, R., et al. (2009). Overrides of medication alerts in ambulatory care. *Archives of Internal Medicine*, 169(3), 305.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning: With applications in R*. (Springer Texts in Statistics). New York: Springer.
- Jiang, G., Solbrig, H. R., Pathak, J., et al. (2015). Developing a standards-based information model for representing computable diagnostic criteria: A feasibility study of the NQF Quality Data Model. *Studies in Health Technology and Informatics*, 216, 1097.
- Jiang, G., Kiefer, R. C., Sharma, D. K., et al. (2017a). A consensus-based approach for harmonizing the OHDSI common data model with HL7 FHIR. *Studies in Health Technology and Informatics*, 245, 887–891.
- Jiang, G., Kiefer, R. C., Prud'hommeaux, E., et al. (2017b). Building interoperable FHIR-based vocabulary mapping services: A case study of OHDSI vocabularies and mappings. *Studies in Health Technology and Informatics*, 245, 1327.
- Jing, X., Cimino, J. J., & Del Fiol, G. (2015). Usability and acceptance of the librarian infobutton tailoring environment: An open access online knowledge capture, management, and configuration tool for OpenInfobutton. *Journal of Medical Internet Research*, 17(11), e272.
- Kahn, C. E., Roberts, L. M., Shaffer, K. A., et al. (1997). Construction of a Bayesian network for mammographic diagnosis of breast cancer. *Computers in Biology and Medicine*, 27, 19–29.
- Karsh, B.-T., Weinger, M. B., Abbott, P. A., et al. (2010). Health information technology: Fallacies and sober realities. *Journal of the American Medical Informatics Association*, 17(6), 617–623. <https://doi.org/10.1136/jamia.2010.005637>.
- Kaushal, R., Jha, A., Franz, C., et al. (2006). Return on investment for a computerized physician order entry system. *Journal of the American Medical Informatics Association*, 13(3), 261–266.
- Kawamoto, K., Del Fiol, G., Strasberg, H. R., et al. (2010). Multi-national, multi-institutional analysis of clinical decision support data needs to inform development of the HL7 Virtual Medical Record standard. *AMIA Annual Symposium Proceedings*, 2010, 377–381.
- Kawamoto, K., et al. (2013). “Key principles for a national clinical decision support knowledge sharing framework: synthesis of insights from leading subject matter experts.” *J Am Med Inform Assoc*. 20(1): 199–207.
- Knaus, W. A., Wagner, D. P., & Lynn, J. (1991). Short-term mortality predictions for critically ill hospitalized adults: Science and ethics. *Science*, 254, 389–394.
- Kohn, L., Corrigan, J., & Donaldson, M. (Eds.). (2002). *To err is human: Building a safer health system*. Institute of Medicine. Washington, DC: National Academy Press.
- Komaroff, A., Black, W., & Flatley, M. (1974). Protocols for physician assistants: Management of diabetes and hypertension. *New England Journal of Medicine*, 290, 370–312.
- Langheier, J. M., & Snyderman, R. (2004). Prospective medicine: The role for genomics in personalized health planning. *Pharmacogenomics*, 5(1), 1–8. <https://doi.org/10.1517/phgs.5.1.1.25685>.
- Leape, L. L. (1994). Error in medicine. *JAMA: The Journal of the American Medical Association*, 272(23), 1851–1857. <https://doi.org/10.1001/jama.1994.03520230061039>.
- Ledley, R., & Lusted, L. (1959). Reasoning foundations of medical diagnosis. *Science*, 130, 9–21.
- Lindgren, H. (2008). Decision support system supporting clinical reasoning process - an evaluation study in dementia care. *Studies in Health Technology and Informatics*, 136, 315–320.
- Lomatan, E. A., Meadows, G., Michaels, M., et al. (2019). To share is human! CDS Connect: A growing national repository of shareable, interoperable clinical decision support. *Proceedings of the AMIA Spring Congress*.
- Longhurst, C. A., Harrington, R. A., & Shah, N. H. (2014). A ‘green button’ for using aggregate patient data at the point of care. *Health Affairs*, 33(7), 1229–1235.
- Mandel, J. C., Kreda, D. A., Mandl, J., et al. (2016). SMART on FHIR: A standards-based, interoperable apps platform for electronic health records. *Journal of the American Medical Informatics Association*, 23(5), 899–908.
- Mandl, K. D., & Kohane, I. S. (2012). Escaping the EHR trap — the future of health IT. *The New England Journal of Medicine*, 366, 2240–2242.
- Marcial, L. H., Richardson, J. E., Lasater, B., et al. (2018). The imperative for patient-centered clinical decision support. *eGEMS*, 6(1), 12.
- McClellan, M. (2015). Accountable care organizations and evidence-based payment reform. *JAMA: The Journal of the American Medical Association*, 313(21), 2128–2123.
- Melnick, E. R., Dyrbye, L. N., Sinsky, C. A., et al. (2019). The association between perceived electronic health record usability and professional burnout among US physicians. *Mayo Clinic Proceedings*. <https://doi.org/10.1016/j.mayocp.2019.09.024>.
- Middleton, B., Shwe, M. A., Heckerman, D. E., et al. (1991). Probabilistic diagnosis using a reformulation of the INTERNIST-1/QMR knowledge base.

- Part II: Evaluation of diagnostic performance. *Meth Inf Medicine*, 4, 256–267.
- Middleton, B., Anderson, J., Fletcher, J., et al. (1998). Use of the WWW for distributed knowledge engineering for an EMR: The KnowledgeBank concept. *Proceedings of the AMIA Annual Symposium*, pp. 126–130.
- Middleton, B. (2009). “The clinical decision support consortium.” *Stud Health Technol Inform*. 150, 26–30.
- Musen, M. A. (1998). Domain ontologies in software engineering: Use of PROTÉGÉ with the EON architecture. *Methods of Information in Medicine*, 37(4–5), 540–550.
- Musen, M. A. (2014). Knowledge representation. In I. N. Sarkar (Ed.), *Methods in biomedical informatics: A pragmatic approach*. Waltham, MA: Academic Press, Elsevier.
- Musen, M. A. (2015). The Protégé project: A look back and a look forward. *AI Matters. Association of Computing Machinery Special Interest Group in Artificial Intelligence*, 1(4), 4–12. <https://doi.org/10.1145/2557001.25757003>.
- Musen, M. A., Tu, S. W., Das, A. K., et al. (1996). EON: A component-based approach to automation of protocol-directed therapy. *Journal of the American Medical Informatics Association*, 3(6), 367–388.
- National High Blood Pressure Education Program. (2004). *The Seventh Report of the Joint National Committee on prevention, detection, evaluation, and treatment of high blood pressure*. National Heart, Lung, and Blood Institute, National Institutes of Health.
- National Research Council (US) Committee on Engaging the Computer Science Research Community in Health Care Informatics, Stead, W. W., & Lin, H. S. (2009). *Computational technology for effective health care: Immediate steps and strategic directions* (The National Academies Collection: Reports funded by National Institutes of Health). Washington, DC: National Academies Press (US).
- Noy, N., Gao, Y., Jain, A., et al. (2019). Industry-scale knowledge graphs: Lessons and challenges. *Communications of the ACM*, 62(8), 36–43.
- Odigie, E., Lacson, R., Raja, A., et al. (2019). Fast healthcare interoperability resources, clinical quality language, and systematized nomenclature of medicine-clinical terms in representing clinical evidence logic statements for the use of imaging procedures: Descriptive study. *JMIR Medical Informatics*, 7(2), e13590.
- Osheroff, J. A., Teich, J. M., Middleton, B., et al. (2007). A roadmap for national action on clinical decision support. *Journal of the American Medical Informatics Association*, 14(2), 141–145. <https://doi.org/10.1197/jamia.M2334>.
- Osheroff, J. A., Teich, J. A., Levick, D., et al. (2012). *Improving outcomes with clinical decision support: An implementer's guide*. Chicago, IL: HIMSS: Healthcare Information Management and Systems Society.
- Overby, C. L., Tarczy-Hornoch, P., Hoath, J. I., et al. (2010). Feasibility of incorporating genomic knowledge into electronic medical records for pharmacogenomic clinical decision support. *BMC Bioinformatics*, 11(Suppl 9), S10. <https://doi.org/10.1186/1471-2105-11-S9-S10>.
- Pageler, N. M., Longhurst, C. A., Wood, M., et al. (2014). Use of electronic medical record-enhanced checklist and electronic dashboard to decrease CLABSI. *Pediatrics*, 133(3), e738–e746.
- Patel, V. L., Arocha, J. F., & Zhang, J. (2010). Medical reasoning and thinking. In K. J. Holyoak & R. G. Morrison (Eds.), *Oxford handbook of thinking and reasoning* (pp. 736–754). Oxford: Oxford University Press.
- Pathak, J., Bailey, K. R., Beebe, C. E., et al. (2013). Normalization and standardization of electronic health records for high-throughput phenotyping: The SHARPN consortium. *Journal of the American Medical Informatics Association*, 20(e2), e341–e348.
- Pauker, S. G., & Kassirer, J. P. (1981). Clinical decision analysis by computer. *Archives of Internal Medicine*, 141(13), 1831–1837.
- Peleg, M., Tu, S., Bury, J., et al. (2003). Comparing computer-interpretable guideline models: A case-study approach. *Journal of the American Medical Informatics Association*, 10, 52–68. <https://doi.org/10.1197/jamia.M1135>.
- Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). “Why should I trust you?” Explaining the predictions of any classifier. *Proceedings of the Conference on Knowledge Discovery in Databases*, San Francisco, CA.
- Saria, S., Rajani, A. K., Gould, J., et al. (2010). Integration of early physiological responses predicts later illness severity in preterm infants. *Science Translational Medicine*, 2(48), 48ra65.
- Schedlbauer, A., Prasad, V., Mulvaney, C., et al. (2009). What evidence supports the use of computerized alerts and prompts to improve clinicians’ prescribing behavior? *Journal of the American Medical Informatics Association*, 16(4), 531–538. <https://doi.org/10.1197/jamia.M2910>.
- Schnipper, J. L., Linder, J. A., Palchuk, M. B., et al. (2008). “Smart Forms” in an Electronic Medical Record: Documentation-based clinical decision support to improve disease management. *Journal of the American Medical Informatics Association*, 15(4), 513–523. <https://doi.org/10.1197/jamia.M2501>.
- Schwartz, W. B. (1970). Medicine and the computer: The promise and problems of change. *New England Journal of Medicine*, 283(23), 1257–1264.
- Shachter, R. D. (1986). Evaluating influence diagrams. *Operations Research*, 34(6), 871–882. <https://doi.org/10.1287/opre.34.6.871>.
- Shah, N. R., Seger, A. C., Seger, D. L., et al. (2006). Improving acceptance of computerized prescribing alerts in ambulatory care. *Journal of the American Medical Informatics Association : JAMIA*, 13(1), 5–11. <https://doi.org/10.1197/jamia.M1868>.

- Shahar, Y., & Musen, M. A. (1996). Knowledge-based temporal abstractions in clinical domains. *Artificial Intelligence in Medicine*, 8(3), 267–298.
- Shiffman, R. N., Karras, B. T., Agrawal, A., et al. (2000). GEM: A proposal for a more comprehensive guideline document model using XML. *Journal of the American Medical Informatics Association*, 7(5), 488–498.
- Shortliffe, E. H. (1976). *Computer-based medical consultations: MYCIN*. New York: Elsevier/North Holland.
- Shortliffe, E. H., Buchanan, B. G., & Feigenbaum, E. (1979). Knowledge engineering for medical decision making: A review of computer-based clinical decision aids. *Proceedings of the IEEE*, 67, 1207–1224.
- Shwe, M. A., Middleton, B., Heckerman, D. E., et al. (1991). Probabilistic diagnosis using a reformulation of the INTERNIST-1/QMR knowledge base. I. The probabilistic model and inference algorithms. *Methods of Information in Medicine*, 30(4), 241–255.
- Sinsky, C. A., & Privitera, M. R. (2018). Creating a “manageable cockpit” for clinicians: A shared responsibility. *JAMA Internal Medicine*, 178(6), 741–742.
- Sittig, D., & Singh, H. (2009). Eight rights of safe electronic health record use. *JAMA*, 302(10), 1111–1113.
- Sittig, D. F., & Singh, H. (2011). Defining health information technology-related errors: New developments since to err is human. *Archives of Internal Medicine*, 171(14), 1281–1284. <https://doi.org/10.1001/archinternmed.2011.327>.
- Sordo, M., Boxwala, A. A., Ogunyemi, O., et al. (2004). Description and status update on GELLO: A proposed standardized object-oriented expression language for clinical decision support. *Studies in Health Technology and Informatics*, 107, 164–168.
- Strom, B. L., Schinnar, R., Aberra, F., et al. (2010). Unintended effects of a computerized physician order entry nearly hard-stop alert to prevent a drug interaction: A randomized controlled trial. *Archives of Internal Medicine*, 170(17), 1578–1583. <https://doi.org/10.1001/archinternmed.2010.324>.
- Tang, P. C., Fafchamps, D., & Shortliffe, E. H. (1994). Traditional medical records as a source of clinical data in the outpatient setting. *Proceedings of the Annual Symposium on Computer Applications in Medical Care*, pp. 575–579.
- Tang, P. C., Jaworski, M. A., Fellencer, C. A., et al. (1996). Clinician information activities in diverse ambulatory care practices. *Proceedings of the AMIA Annual Fall Symposium*. pp. 12–16.
- Trafton, J., Martins, S., Michel, M., et al. (2010). Evaluation of the acceptability and usability of a decision support system to encourage safe and effective use of opioid therapy for chronic, noncancer pain by primary care providers. *Pain Medicine*, 11(4), 575–585.
- Ullman-Cullere, M. H., & Mathew, J. P. (2011). Emerging landscape of genomics in the Electronic Health Record for personalized medicine. *Human Mutation*, 32(5), 512–516. <https://doi.org/10.1002/humu.21456>.
- van der Lei, J., Musen, M. A., van der Does, E., et al. (1991). Comparison of computer-aided and human review of general practitioners’ management of hypertension. *Lancet*, 338(8781), 1504–1508.
- van der Sijs, H., Aarts, J., Vulto, A., et al. (2006). Overriding of drug safety alerts in computerized physician order entry. *Journal of the American Medical Informatics Association*, 13(2), 138–147. <https://doi.org/10.1197/jamia.M1809>.
- Wang, S. J., Middleton, B., Prosser, L. A., et al. (2003). A cost-benefit analysis of electronic medical records in primary care. *The American Journal of Medicine*, 114(5), 397–403.
- Wang, J. K., Hom, J., Balasubramanian, S., et al. (2018). An evaluation of clinical order patterns machine-learned from clinician cohorts stratified by patient mortality outcomes. *Journal of Biomedical Informatics*, 86, 109–119.
- Weingart, S. N., Toth, M., Sands, D. Z., et al. (2003). Physicians’ decisions to override computerized drug alerts in primary care. *Archives of Internal Medicine*, 163(21), 2625–2631. <https://doi.org/10.1001/archinte.163.21.2625>.
- Wright, A., Sittig, D. F., Ash, J. S., et al. (2009). Clinical decision support capabilities of commercially-available clinical information systems. *Journal of the American Medical Informatics Association*, 16(5), 637–644.
- Wright, A., Sittig, D. F., Ash, J. A., et al. (2015). Lessons learned from implementing service-oriented clinical decision support at four sites: A qualitative study. *International Journal of Medical Informatics*, 84(11), 901–911.
- Yu, V. L., Fagan, L. M., Wraith, S. M., et al. (1979a). Antimicrobial selection by a computer. A blinded evaluation by infectious disease experts. *Journal of the American Medical Association*, 242(12), 1279–1282.
- Yu, V. L., Buchanan, B. G., Shortliffe, E. H., et al. (1979b). Evaluating the performance of a computer-based consultant. *Computer Programs in Biomedicine*, 9(1), 95–102.
- Zhang, J., & Walji, M. F. (2011). TURF: Toward a unified framework of EHR usability. *Journal of Biomedical Informatics*, 44(6), 1056–1067.
- Zimmerman, J. E., Kramer, A. A., McNair, D. S., et al. (2006). Acute Physiology and Chronic Health Evaluation (APACHE) IV: Hospital mortality assessment for today’s critically ill patients. *Critical Care Medicine*, 34(5), 1297–1310. <https://doi.org/10.1097/01.CCM.0000215112.84523.F0>.