



ELSEVIER

journal homepage: www.intl.elsevierhealth.com/journals/ijmi

Review

A four-phase model of the evolution of clinical decision support architectures

Adam Wright^{a,b,c,*}, Dean F. Sittig^{d,e}

^a Clinical Informatics Research and Development, Partners HealthCare, Boston, MA, United States

^b Division of General Medicine, Brigham & Women's Hospital, Boston, MA, United States

^c Department of Medicine, Harvard Medical School, Boston, MA, United States

^d Department of Medical Informatics, Northwest Permanente, PC, Portland, OR, United States

^e Department of Medical Informatics and Clinical Epidemiology, Oregon Health and Science University, Portland, OR, United States

ARTICLE INFO

Article history:

Received 2 June 2007

Received in revised form

26 January 2008

Accepted 30 January 2008

Keywords:

Decision support systems, clinical
Decision making, computer-assisted
Decision support techniques
Hospital information systems
Medical records systems,
computerized
Models, theoretical

ABSTRACT

Background: A large body of evidence over many years suggests that clinical decision support systems can be helpful in improving both clinical outcomes and adherence to evidence-based guidelines. However, to this day, clinical decision support systems are not widely used outside of a small number of sites. One reason why decision support systems are not widely used is the relative difficulty of integrating such systems into clinical workflows and computer systems.

Purpose: To review and synthesize the history of clinical decision support systems, and to propose a model of various architectures for integrating clinical decision support systems with clinical systems.

Methods: The authors conducted an extensive review of the clinical decision support literature since 1959, sequenced the systems and developed a model.

Results: The model developed consists of four phases: standalone decision support systems, decision support integrated into clinical systems, standards for sharing clinical decision support content and service models for decision support. These four phases have not heretofore been identified, but they track remarkably well with the chronological history of clinical decision support, and show evolving and increasingly sophisticated attempts to ease integrating decision support systems into clinical workflows and other clinical systems.

Conclusions: Each of the four evolutionary approaches to decision support architecture has unique advantages and disadvantages. A key lesson was that there were common limitations that almost all the approaches faced, and no single approach has been able to entirely surmount: (1) fixed knowledge representation systems inherently circumscribe the type of knowledge that can be represented in them, (2) there are serious terminological issues, (3) patient data may be spread across several sources with no single source having a complete view of the patient, and (4) major difficulties exist in transferring successful interventions from one site to another.

© 2008 Elsevier Ireland Ltd. All rights reserved.

* Corresponding author at: Division of General Medicine, Brigham & Women's Hospital, 1620 Tremont Street, Boston, MA 02120, United States. Tel.: +1 781 416 8764; fax: +1 781 416 8912.

E-mail address: awright5@partners.org (A. Wright).

1386-5056/\$ – see front matter © 2008 Elsevier Ireland Ltd. All rights reserved.

doi:10.1016/j.ijmedinf.2008.01.004

Contents

1. Introduction	642
2. Phase 1: standalone decision support systems	642
3. Phase 2: decision support integrated into clinical systems	644
4. Phase 3: standards for sharing decision support content	645
5. Phase 4: service models	646
6. Conclusions and future directions	646
Acknowledgements	647
References	647

1. Introduction

Clinical decision support has been defined in myriad ways, ranging from extremely precise and narrow definitions that may exclude broad categories of work to all-encompassing definitions. This paper will use a consensus definition of clinical decision support: “Providing clinicians, patients or individuals with knowledge and person-specific or population information, intelligently filtered or presented at appropriate times, to foster better health processes, better individual patient care, and better population health” [1].

Decision support systems have been used for a variety of purposes, ranging from quality and safety to efficiency, and across a variety of clinical domains such as screening, diagnosis and therapy. A substantial body of evidence exists to suggest that decision support systems can be extremely effective [2–8]. A recent systemic review by Garg found that in 100 studies of clinical decision support, “CDSS improved practitioner performance in 62 (64%) of the 97 studies assessing this outcome, including 4 (40%) of 10 diagnostic systems, 16 (76%) of 21 reminder systems, 23 (62%) of 37 disease management systems, and 19 (66%) of 29 drug-dosing or prescribing systems” [4].

Despite their great potential, and a history of successes, clinical decision support systems have not found wide use outside of a handful of mostly academic medical centers, the Veterans’ Health Administration (VA), and large, integrated delivery systems such as Kaiser Permanente [9].

In this paper we review the history of clinical decision support. We endeavored to cover the most influential and widely cited decision support systems throughout the history of the field and, in particular, to trace the development of architectures for clinical decision support systems. By architectures, we mean the way in which decision support systems interact (or choose not to interact) with other related systems, such as computerized physician order entry (CPOE) and electronic health record (EHR) systems. Based on our review, we have formulated a model with four distinct architectural phases for decision support:

1. Standalone decision support systems, beginning in 1959.
2. Integrated systems, beginning in 1967.
3. Standards-based systems, beginning in 1989.
4. Service models, beginning in 2005.

Fig. 1 shows a schematic of the model. These phases are described in detail in the forthcoming sections. It is

worth noting that there are nearly limitless other axes along which decision support systems can be categorized, and accordingly, models could be developed along nearly any dimension. Other potentially fruitful axes might include clinical intent (diagnosis, therapy, prevention, public health, telemedicine, etc.), mechanism of intervention (alerts, reminders, critiquing, information-based) or method of reasoning (forward-chaining, backward-chaining, rule-based, NLP-based, etc.) [10–18].

The proposed architectural model is sequential and evolutionary: the phases happened in order and it is clear that systems in each phase learned from and were influenced by prior phases. That said, just as in other evolutionary models, the earlier phases have not necessarily faced extinction: there are still viable Phase 1 systems being built today, even though the architectural state of the art has evolved significantly.

2. Phase 1: standalone decision support systems

Pinpointing the beginning of the field of medical informatics is challenging, but it is perhaps best to begin any discussion of the field with the 1959 paper “Reasoning foundations of medical diagnosis; symbolic logic, probability, and value theory aid our understanding of how physicians reason” by Ledley and Lusted [19]. Ledley went on to invent the whole-body CT scanner [20], and Lusted became a leader in the field of medical decision-making. Their 1959 paper, however, laid out a probabilistic model for medical diagnosis, with grounds in set-theory and Bayesian inference. The article, published in *Science*, was a tutorial in statistical and probabilistic inference for clinicians. The paper proposed an analog computer used to sort cards. These cards would contain a diagnosis, and a series of punches which represented symptoms. By selecting the cards which matched the symptoms present in a given case, a clinician could develop a differential diagnosis. The number of cards supporting a particular diagnosis represented the likelihood of that diagnosis. The system could easily be updated as new patients were seen—the clinician simply had to fill out and punch a card for each new patient and diagnosis, and drop it into the sorter.

Two years after Ledley and Lusted published their paper, Warner of the LDS Hospital and the University of Utah published a mathematical model for diagnosing congenital heart defects [21]. This model used contingency tables to map symptoms and signs to diagnoses, based on the frequency of

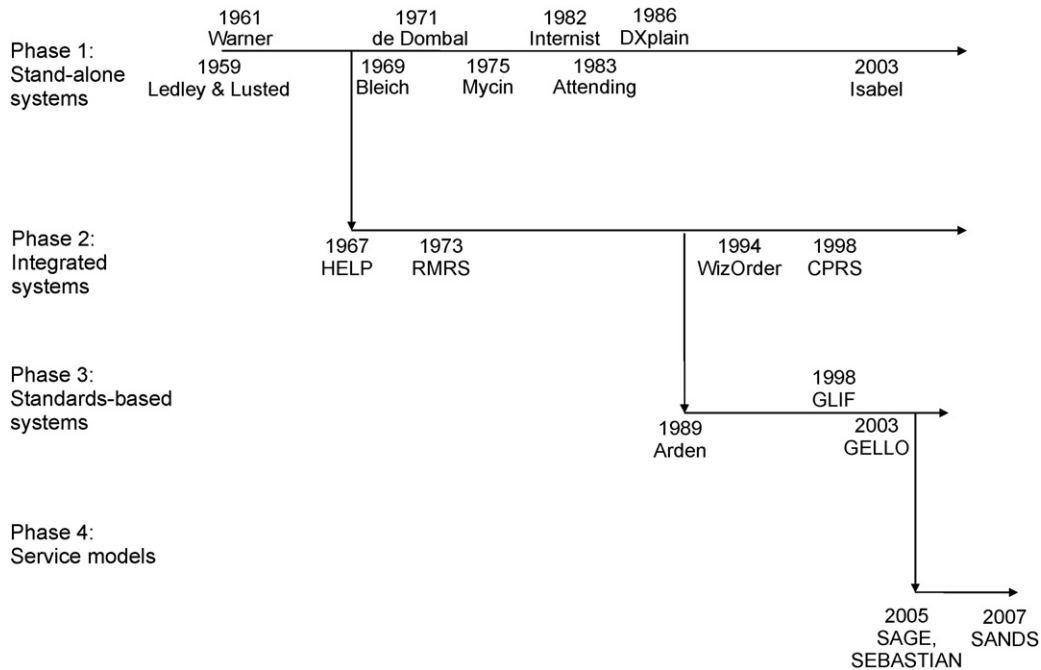


Fig. 1 – A schematic drawing of the four-phase model for clinical decision support.

manifestation for each symptom or sign given an underlying diagnosis. The system was evaluated by comparing its diagnoses with gold-standard surgical diagnoses, and was found to compare favorably with experienced cardiologists.

Shortly after Warner, Collen developed a system for “Automated Multiphasic Screening And Diagnosis” which was used at Kaiser Permanente, and which Collen described in a 1964 paper [22]. When a patient came in for an exam he or she was given a stack of cards each of which contained a symptom or a question. The patient would put the cards which indicated a symptom he or she was experiencing, or a question to which their answer was affirmative, into a designated “Yes” box, and the rest of the cards into a “No” box. These cards were then run through a computer system which proposed an initial differential diagnosis.

In 1969 Bleich developed a system to suggest therapy for acid–base disorders [23]. This system was unique because it was the first system to suggest a therapy in addition to a diagnosis. The system would take in information useful for diagnosing acid–base disorders, such as the results of arterial blood gas tests, vital signs, and clinical findings. If any information needed for decision-making was missing, the system would prompt the user to collect and enter that information. If the information was complete the system would produce an evaluation note, written in the same style as a human specialist consultant, proposing a management plan for review by the physician providing care. Acid–base disorders were a particularly apt clinical domain because their management is fairly closed and algorithmic.

Two years later, in 1971, de Dombal built a probabilistic model and computer system for diagnosing abdominal complaints [24,25]. The system was significant because evaluation showed it to be quite effective. When compared to the final gold-standard surgical diagnosis, the computer’s

preliminary diagnosis was accurate 91.8% of the time. By comparison, a group of senior clinicians was correct 79.6% of the time. Not only was the computer able to match the performance of senior clinicians, it actually improved significantly on it—cutting the error rate in half.

In the 1970s, the field of artificial intelligence began to influence medical informatics. In 1975 Shortliffe applied the relatively new techniques of expert system modeling using backward chaining to the problem of antibiotic prescribing with his system, MYCIN [26]. With MYCIN, a clinician would enter what was known about a patient’s infectious process. The system would then apply to these facts a series of rules from its knowledge base. The system could then request additional information from the clinician, or suggest what it considered optimal antibiotic therapy based on the facts presented. Early evaluation showed it suggested acceptable therapy 75% of the time, and it got better as more rules were added.

Most of these early systems would, broadly, take input of clinical parameters and make suggestions of diagnoses or therapy. The ATTENDING system [27,28] by Perry Miller of Yale, however, took a different approach. The user of ATTENDING would, as with the other systems, input clinical parameters, but he or she would also enter a proposed plan. The system would then make comments and suggestions about the plan, and it would be up to the user to change the plan based on these suggestions. This method of interaction was called critiquing, and critiquing systems were eventually developed for ventilator management, hypertension, and other clinical domains.

The systems discussed heretofore all have one thing in common: they are limited to one specific area of medicine, such as antibiotic prescribing, or congenital heart defects. The INTERNIST-I system [29], developed by Randolph Miller et al. in

the early 1980s attempted to provide diagnostic decision support across the entire field of internal medicine. The system's knowledge base comprised "15 person-years of work, 500 disease profiles [and], 3550 manifestations of disease" [29]. It was tested on 19 standardized clinical exercises published in the *New England Journal of Medicine*, and did about as well as an average doctor in proposing the correct diagnosis for the case, but not as well as the experts who wrote the cases up. One key intellectual contribution of the INTERNIST system was the way it abstracted the complex field of diagnosis into three concepts: evoking strength, frequency and import. Shortly after INTERNIST, Barnett, and others, released the DXplain system [30]. Unlike INTERNIST, DXplain was designed to explain its diagnostic reasoning process. The DXplain system is still available today, and has been updated with a web-interface [31]. INTERNIST eventually evolved into the QMR system, though neither system is commercially available today.

All of the systems reviewed in this section fall into the category of standalone decision support systems. That is, they were systems (usually, but not exclusively, computer programs) which ran separately from any other system. To employ one, a clinician had to intentionally and purposefully seek the system out, log into it, enter information about his or her case, and then read and interpret the results. Such systems have several key advantages over other types of clinical decision support systems. First, anyone with access to clinical knowledge and computing skills can make one. Doing so requires no special access to patient-specific clinical data or clinical systems. There is no need to standardize on anything: completely arbitrary systems can be used for terminology, input structure, output format and knowledge representation. Such systems are also easy to share—the creator could simply mail a copy of the program to anyone who wished to use his or her system (or, in the modern analog, upload the system to a website).

Along with these advantages, though, standalone clinical decision support systems have some significant, and potentially disqualifying, disadvantages. First and foremost, people have to seek the systems out, so the system cannot be proactive. But the cause of many medical errors is lack of knowledge—people do not know what they do not know. A system which is not proactive cannot be of assistance in such a case. The other disadvantage is more practical; these systems were extremely inefficient to use. It could take well over an hour to enter a case into the INTERNIST system, so its use was, naturally, quite narrow. This limitation was especially bothersome where systems were dealing with data, such as laboratory results, that were likely available in electronic form in another system, but which still had to be manually entered due to a lack of system integration.

3. Phase 2: decision support integrated into clinical systems

To surmount the significant problems with standalone clinical decision support systems, developers began integrating such systems into other clinical systems, such as CPOE and EHR systems. The HELP system [32], developed at the LDS Hospital in Salt Lake City, Utah starting in 1967 was the first example of such integration. The system was first used in the

cardiac catheterization laboratory and in the cardiac intensive care unit. Subsequently, it was expanded to provide sophisticated clinical decision-support capabilities to a wide variety of clinical areas such as the clinical laboratory, nurse charting, radiology, pharmacy, respiratory therapy, ICU monitoring, and the emergency department to name just a few [32]. The HELP system is currently operational in most of Intermountain Health Care's (IHC) over 30 hospitals. HELP had advanced Bayesian decision support capabilities, and included modules for a variety of functions, including many of the functions described above. HELP also served as the substrate for many successful projects in clinical decision support, such as Sittig's COMPAS system [33] for ventilator management, a system for blood product ordering developed by Gardner [34,35] and a well-known antibiotic advising system developed by Evans [36] among many others.

In 1973, McDonald of the Regenstrief Institute in Indiana was developing the Regenstrief Medical Record System (RMRS) [37]. This system used a large base of rules to make suggestions about care. The system was evaluated in an experimental trial. In the first half of the trial, half of the users of the RMRS system received suggestions based on the database of rules while half received no suggestions. Dr. McDonald found that physicians carried out the suggestions they received 51% of the time. However, when they did not receive suggestions, they carried out the appropriate care only 22% of the time [38]. Even more interesting, when the system was turned off, performance went back to baseline almost immediately—the physicians who had been receiving the alerts, dropped down to pre-trial levels: there was no learning effect. In this landmark paper, McDonald argued that medicine had become so complex, and the amount of information required to practice it effectively so expansive, that no un-aided human could provide perfect care. Instead, some sort of ancillary aid, like a computer, was needed.

In addition to HELP and RMRS, a variety of other clinical systems, mostly at academic medical centers, have been used for clinical decision support, beginning in 1980s and 1990s. The WizOrder system [39], in use at Vanderbilt, and now available commercially as Horizon Expert Orders from McKesson has been a fruitful development platform, as has the Brigham Integrated Computing System (BICS) in use at the Brigham & Women's Hospital [40,41]. BICS includes support for a variety of CDS interventions including clinical pathways which can guide clinicians through data entry and ordering tasks according to a specific clinical purpose, as well as supplemental time-based pathway support, such as reminding clinicians to complete orders relevant to a pathway which are not yet done. The Veterans Health Administration has also been a leader in the field of clinical decision support with their Computerized Patient Record System (CPRS) [42], reporting spectacular outcomes for even simple interventions, and quickly rocketing from a position as one of the lower-performing healthcare systems, to a quality leader [43].

Integrating CDS into clinical information systems solved some problems, while creating others. Integrated systems have two key advantages over standalone systems: first, the user does not have to reenter information which is already stored electronically; and, second, such systems can be proactive—they can alert a user to a dangerous drug-drug

interaction or a dosing error without the user seeking assistance. The major downside of integrated systems is that there is no easy way to share them or reuse their content. Standalone systems can be easily shared; often by simply mailing a tape or emailing an executable program. In contrast, since integrated systems are built directly into larger clinical systems they cannot be directly shared with others who are not using the same clinical system. Also, integrating decision support into a clinical system can create knowledge-management problems. It is hard to separate knowledge from code; if a clinical guideline is updated, it may be necessary to review the entire source code for a clinical system to find the places where this guideline is used. And finally, nearly everyone who has been successful at developing integrated decision support systems has also developed their own clinical system. However, the vast majority of hospitals and doctors buy rather than build clinical systems so this limitation has kept this type of decision support from seeing wide adoption.

4. Phase 3: standards for sharing decision support content

In response to the inability to share decision support content, a variety of efforts have been undertaken to standardize clinical decision support content. Foremost among these is the Arden Syntax [44–48]. The initial version of the Arden Syntax was developed at a 3-day consensus meeting in June of 1989 held at the Arden Homestead in New York from which the standard got its name. The standard combined the syntaxes used by the HELP system and the RMRS system because these systems were (and to an extent, still are) the two most prominent clinical systems. Both systems are discussed above.

Rules encoded in Arden Syntax are called Medical Logic Modules. The Arden Syntax divides rules into three sections, called the “maintenance”, “library” and “knowledge” sections. The maintenance section contains meta-data about the rule, such as who owns it, when it was created, when it was last reviewed or updated, and its validation status. The library section contains meta-data describing the clinical role of the rule, its purpose, an explanation, keywords, and a citation to the original source of the guideline or best practice that the rule encodes. The computable portion of the rule is encoded in the knowledge section. The knowledge section contains subsections called “type”, “data”, “evoke”, “logic”, “action” and “urgency”. In the current version of Arden Syntax, type is always set to “data-driven” because this is the only mode of decision support offered. The data section is used to read data values, such as recent lab tests, medications lists, or clinical problems from the encompassing clinical system. The evoke section contains one or more triggers that might cause the rule to fire, such as “new potassium value stored”. The logic section encodes the rule, generally as a series of if-then statements, and the action section encodes what the rule does when its logic section is satisfied—in general, Arden Syntax has only been used to raise alerts. The urgency section contains a number between 1 and 100 to encode how important that rule is. The guidelines used to assign urgencies is implementation dependent, and not fully defined in the specification.

Arden Syntax has had some limited commercial success. Three clinical system vendors (Eclipsys, McKesson and Siemens), which together represent about a quarter of the overall clinical information system market, offer some support for the Arden Syntax, and a number of vendors, most notably Thomson Micromedex (Denver, CO) and Zynx (Los Angeles, CA) sell Medical Logic Modules.

Arden syntax has two key limitations: first, it can only be used to encode event-driven, patient-specific rules. For use cases such as drug–drug interaction checking, or panic lab value alerting, this modality is sufficient. However, because Arden Syntax is patient-specific, it cannot be used for population-based decision support (such as a quality-of-care dashboard), and because it is event-driven, it cannot be used for point-of-care reference or information retrieval support. The other key limitation relates to vocabulary: Arden Syntax does not define a standard vocabulary for things like lab tests, drugs or procedures. As a result, even if two clinical systems support the Arden Syntax, if they use different clinical terminologies, Arden Syntax rules from one system cannot be used in the other system without modification. For example, if one hospital’s clinical system stored a blood test result as “Serum Potassium” and another hospital’s clinical system stored the same result as “K+” a human-guided mapping would be needed. To assist in this mapping, Arden Syntax wraps system-specific terminological expressions in curly braces, and automated tools exist to help the implementer disambiguate these bracketed terms, but human intervention is still required. This problem is so limiting and well-known that it is referred to simply as the “curly braces problem.” The Arden Syntax has been revised several times since it was first created in 1989, and its most recent version (Version 2.6) has been accepted as a standard by both the American National Standards Institute (ANSI), and Health Level 7 (HL7), a healthcare standards body [49].

Since the creation of Arden Syntax, numerous other standards for representing and sharing decision support content and knowledge have been created. Many of these efforts have stalled, but one effort in particular, the Guideline Interchange Format [50] (GLIF), developed over the past decade, has gained some traction. Unlike Arden Syntax, which is mostly designed for Alerts and Reminders, GLIF focuses on more complex multi-part guidelines, including complex clinical pathways that take place in phases or over time. A general-purpose execution engine for executing GLIF guidelines has been described [51], but it has not yet been implemented in any commercially available system.

Although GLIF is not currently available in any commercial system, it was pilot tested from 2000 to 2003 by a group of three universities: Stanford, Columbia and Harvard, calling themselves the InterMed Consortium [52]. The consortium received a grant to practice encoding and sharing rules in GLIF, and had some success in doing so. By pilot testing the standard, the consortium learned many lessons about the scope and potential of the language, and used many of these findings to refine and improve the language.

Many other standards for representing decision support content in a standardized way exist. OpenClinical.org [53] provides an overview of guideline and knowledge modeling standards, such as Arden Syntax, GLIF and GELLO. HL7 man-

ages a number of these projects, including Arden Syntax, a Decision Support Service (DSS), which will be discussed in the next section, and a promising new order set standard.

The use of standards to represent, encode, store and share knowledge overcomes many of the disadvantages of the natively integrated decision support systems described in Phase 2 above. It provides a method for sharing the decision support content, and separates the code describing such content from the code which implements the clinical information system. However, standards also have some inherent limitations and disadvantages. First, there are often too many standards to choose from several dozen standards are available to represent simple alerts and reminders. The second problem is that any encoding standard inherently constrains what a user can encode in the standard to that which the standard-writer intended and included in the scope of the standard. At the time that Arden Syntax was developed, the goal was to develop a system for patient-specific event-driven decision support and, as such, this is the only type of decision support that can be encoded in Arden Syntax. This limitation is not present in standalone or natively integrated decision support systems since their scope is limited only by the underlying capabilities of the programming languages and data models used. Terminology issues have also significantly limited the adoption of these standards—unless clinical systems and decision support systems share a common terminology standard, cumbersome mappings must be undertaken. Finally, even if there were a perfect standard for sharing decision support rules and guidelines, there are significant unanswered questions about where such guidelines would be kept, how they would be owned, who would be liable for them, how they would be evaluated, or who would keep them up to date.

5. Phase 4: service models

More recent efforts have separated the clinical information system and clinical decision support system components of an integrated decision support system, and recombined them by using a standard application programming interface (API). The first effort along this front was the Shareable Active Guideline Environment project (SAGE) [54,55]. SAGE placed an API in front of the clinical system. A properly designed SAGE rule could interact with any clinical system that supported this SAGE-compliant API. The approach that SAGE took, placing a standardized interface in front of the clinical system, has been termed a Virtual Medical Record (VMR) approach [56]. The principle advantage of this approach is that it solves the vocabulary problem—the SAGE virtual medical record specifies the vocabularies that will be used to access and process the medical record, and to the extent that a clinical system uses different terminologies, it is required to provide a suitable mapping. Like Arden Syntax, SAGE requires a standard guideline format, necessarily constraining the type of decision support that can be implemented in SAGE.

SEBASTIAN, a more recent system first described in 2005, has taken the opposite approach from SAGE [57]. It places a standardized interface in front of clinical decision support modules, and makes only limited demands on the clinical sys-

tem to store data in any particular way. In this model, any clinical system which understands the SEBASTIAN protocol can make queries of centralized Decision Support Services. SEBASTIAN maintains most of the same advantages of something like the Arden Syntax, while freeing the user from the restrictions that statically defined knowledge representation formats impose. Moreover, since the modules are located on the Internet, they can be shared by more than one hospital, allowing for greater efficiency. However, SEBASTIAN makes significant demands of the consumers (i.e. clinical systems) of the services it provides. First, although SEBASTIAN is standardized, each knowledge module is free to require any given set of data, which the clinical system must fetch and provide. Moreover, two SEBASTIAN knowledge modules may use different vocabulary standards, forcing the consumer of the service to provide the same data more than once in different encodings (and necessitating that the clinical systems provide support for all the vocabulary standards chosen). Also, SEBASTIAN requires that a service consumer move patient data to the service, which some hospitals or providers may be reluctant to do. Further, because the amount of patient data needed may potentially be large, performance issues may manifest, although in early testing to this point, performance has been acceptable [57]. These limitations aside, the SEBASTIAN architecture is promising. SEBASTIAN began as a Ph.D. student project, but progress is now continuing through HL7 [58] as the HL7 DSS. The HL7 DSS uses the HL7 Version 3 Reference Information Model (RIM) as its patient data model. If adopted, this would resolve the vocabulary challenges described above. Perhaps more concerning is a recent revelation that the creators of SEBASTIAN have filed for a patent on their model [59-61], creating a degree of intellectual property and licensing uncertainty for other potential implementers of the approach.

Both of these interface-oriented systems provide significant advantages over the systems which require a standard representation, and both are promising. However, each system constrains itself to fully standardizing only one of the two interfaces at the junction between a clinical decision support system and a clinical system, limiting their potential for success. Also, both systems principally look at only one clinical system and one decision support system at a time, although, in the real world, knowledge about a patient (that is stored in a clinical system) and knowledge of medicine (that is stored in a decision support system) can be fragmented across many sites.

6. Conclusions and future directions

Although the four phases we describe here happened in roughly sequential order, it is important to note that decision support systems are still being developed in all four of them. There appears to be some drift, over time, towards more use of the later phases (particularly integrated and standards-based approaches) and less use of entirely stand-alone systems, but these later phases have not, yet, fully supplanted the earlier ones. This is analogous to biological evolution, where evolutionarily favored variant alleles coexist with ancestral alleles for quite some time until a point of fixation is reached. We suspect that the point of fixation for decision support archi-

tectures is distant, and that all four models will continue to be employed for quite awhile, and perhaps further models will emerge.

In the nearer term, we do expect to see increased use of service models (Phase 4) as their modularity allows for the benefits of integrated and standard-based systems without their respective inherent limitations of non-portability and limited range of representable knowledge. As performance of, and access to, high-speed computer networks increase we expect this trend to accelerate. We are already beginning to see a trend towards wider use of service-oriented paradigms in clinical systems and clinical decision support [60–67], although others have expressed their concerns [61].

Ongoing work in HL7 towards developing a standard for service-oriented decision support is encouraging, and successful adoption of such a standard in commercial clinical systems would be a significant step forward. We are hopeful for, but less optimistic about, wide-spread support of a knowledge-representation standard such as Arden Syntax in commercial clinical systems.

Formal adoption of either a knowledge representation standard (Phase 3) or a service protocol (Phase 4) through the Healthcare Information Technology Standards Panel (HITSP) would be helpful in this regard, as would addition of this as a requirement for EHR certification by the Certification Commission for Health Information Technology. While neither HITSP nor CCHIT have announced any formal plans in this regard, both have expressed an interest in this area. The American Health Information Community (AHIC)'s newly formed *ad hoc* Committee on Clinical Decision Support may be an important driver of these efforts going forward.

We expect that broader efforts to standardize data representation formats and terminologies within EHRs, as led by HITSP, will facilitate increased use of clinical decision support. Phases 2–4 all depend on availability of patient data, and these efforts have often been hampered by the variable terminologies and data representation formats used across commercial products (and even across multiple implementations of a single product). As these models are further standardized, the task of developing decision support becomes easier.

Each of the four evolutionary approaches to decision support architecture has had unique advantages and disadvantages, which are described in detail above. That said, there were some common limitations that almost all the approaches faced, and that no single approach has been able to entirely surmount: first, fixed knowledge representation systems inherently circumscribe the type of knowledge that can be represented in them, second, there are serious terminological issues, third, patient data may be spread across several sources with no single source having a complete view of the patient, and fourth, and perhaps most significant, major difficulties exist in transferring successful interventions from one site to another. Although a small number of institutions had great success with decision support, this success could not be (or was not) widely replicated in community settings. We are hopeful that, as these models for decision support evolve and mature, and as new models emerge, these past difficulties can be overcome and adoption of effective decision support interventions will increase across the entire spectrum of care.

Summary points

What was known before this paper

- Clinical decision support systems can be effective tools for improving the quality of healthcare.
- A variety of such systems have been developed and described in the literature, frequently yielding significant improvements in domains ranging from diagnosis to therapy and prevention.
- A common feature of many clinical decision support systems is a mechanism for integrating such systems into clinical information systems such as electronic health records and computerized physician order entry systems.

What is learned as a result

- In this work, the authors review the history of evolution of clinical decision support systems from 1959 to present with a particular emphasis on the mechanism these systems used to integrate with clinical information systems.
- As part of the review, the authors propose a four-phase architectural model for how these mechanisms of integration have evolved, beginning with standalone systems, and continuing through increasingly sophisticated levels of integration.

Acknowledgements

Funding: This work was funded in part by NLM Training Grant 1T15LM009461 and NLM Research Grant R56-LM006942-07A1.

REFERENCES

- [1] J.A. Osheroff, J.M. Teich, B. Middleton, E.B. Steen, A. Wright, D.E. Detmer, A roadmap for national action on clinical decision support, *J. Med. Inform. Assoc.* 14 (2) (2007) 141–145.
- [2] E.A. Balas, S. Weingarten, C.T. Garb, D. Blumenthal, S.A. Boren, G.D. Brown, Improving preventive care by prompting physicians, *Arch. Intern. Med.* 160 (3) (2000) 301–308.
- [3] M.D. Cabana, C.S. Rand, N.R. Powe, et al., Why don't physicians follow clinical practice guidelines? A framework for improvement, *JAMA* 282 (15) (1999) 1458–1465.
- [4] A.X. Garg, N.K. Adhikari, H. McDonald, et al., Effects of computerized clinical decision support systems on practitioner performance and patient outcomes: a systematic review, *JAMA* 293 (10) (2005) 1223–1238.
- [5] J.M. Grimshaw, I.T. Russell, Effect of clinical guidelines on medical practice: a systematic review of rigorous evaluations, *Lancet* 342 (8883) (1993) 1317–1322.
- [6] D.L. Hunt, R.B. Haynes, S.E. Hanna, K. Smith, Effects of computer-based clinical decision support systems on physician performance and patient outcomes: a systematic review, *JAMA* 280 (15) (1998) 1339–1346.

- [7] M.E. Johnston, K.B. Langton, R.B. Haynes, A. Mathieu, Effects of computer-based clinical decision support systems on clinician performance and patient outcome. A critical appraisal of research, *Ann. Intern. Med.* 120 (2) (1994) 135–142.
- [8] K. Kawamoto, C.A. Houlihan, E.A. Balas, D.F. Lobach, Improving clinical practice using clinical decision support systems: a systematic review of trials to identify features critical to success, *BMJ* 330 (7494) (2005) 765.
- [9] B. Chaudhry, J. Wang, S. Wu, et al., Systematic review: impact of health information technology on quality, efficiency, and costs of medical care, *Ann. Intern. Med.* 144 (10) (2006).
- [10] R.A. Miller, Medical diagnostic decision support systems—past, present, and future: a threaded bibliography and brief commentary, *J. Am. Med. Inform. Assoc.* 1 (1) (1994) 8–27.
- [11] J.M. Teich, M.M. Wrinn, Clinical decision support systems come of age, *MD Comput.* 17 (1) (2000) 43–46.
- [12] A. Wright, H. Goldberg, T. Hongsermeier, B. Middleton, A description and functional taxonomy of rule-based decision support content at a large integrated delivery network, *J. Am. Med. Inform. Assoc.* (2007).
- [13] I. Sim, P. Gorman, R.A. Greenes, et al., Clinical decision support systems for the practice of evidence-based medicine, *J. Am. Med. Inform. Assoc.* 8 (6) (2001) 527–534.
- [14] R.A. Miller, L.R. Waitman, S. Chen, S.T. Rosenbloom, The anatomy of decision support during inpatient care provider order entry (CPOE): empirical observations from a decade of CPOE experience at Vanderbilt, *J. Biomed. Inform.* 38 (6) (2005) 469–485.
- [15] J.A. Osheroff, E.A. Pifer, J.M. Teich, D.F. Sittig, R.A. Jenders, Improving Outcomes with Clinical Decision Support: An Implementer's Guide, HIMSS, Chicago, IL, 2005.
- [16] J.K. Wang, M.M. Shabot, R.G. Duncan, J.X. Polaschek, D.T. Jones, A clinical rules taxonomy for the implementation of a computerized physician order entry (CPOE) system, *Proc. AMIA Annu. Symp.* (2002) 860–863.
- [17] T. Falas, G. Papadopoulos, A. Stafylopatis, A review of decision support systems in telecare, *J. Med. Syst.* 27 (4) (2003) 347–356.
- [18] P.L. Miller, Critiquing: a different approach to expert computer advice in medicine, *Med. Inform. (Lond.)* 11 (1) (1986) 29–38.
- [19] R.S. Ledley, L.B. Lusted, Reasoning foundations of medical diagnosis; symbolic logic, probability, and value theory aid our understanding of how physicians reason, *Science* 130 (3366) (1959) 9–21.
- [20] D.F. Sittig, J.S. Ash, R.S. Ledley, The story behind the development of the first whole-body computerized tomography scanner as told by Robert S. Ledley, *J. Am. Med. Inform. Assoc.* 13 (5) (2006) 465–469.
- [21] H.R. Warner, A.F. Toronto, L.G. Veasey, R. Stephenson, A mathematical approach to medical diagnosis. Application to congenital heart disease, *JAMA* 177 (1961) 177–183.
- [22] M.F. Collen, L. Rubin, J. Neyman, G.B. Dantzig, R.M. Baer, A.B. Siegelau, Automated multiphasic screening and diagnosis, *Am. J. Public Health Nations Health* 54 (1964) 741–750.
- [23] H.L. Bleich, Computer evaluation of acid-base disorders, *J. Clin. Invest.* 48 (9) (1969) 1689–1696.
- [24] F.T. de Dombal, J.C. Horrocks, J.R. Staniland, P.J. Guillou, Construction and uses of a 'data-base' of clinical information concerning 600 patients with acute abdominal pain, *Proc. R. Soc. Med.* 64 (9) (1971) 978.
- [25] F.T. de Dombal, D.J. Leaper, J.R. Staniland, A.P. McCann, J.C. Horrocks, Computer-aided diagnosis of acute abdominal pain, *Br. Med. J.* 2 (5804) (1972) 9–13.
- [26] E.H. Shortliffe, R. Davis, S.G. Axline, B.G. Buchanan, C.C. Green, S.N. Cohen, Computer-based consultations in clinical therapeutics: explanation and rule acquisition capabilities of the MYCIN system, *Comput. Biomed. Res.* 8 (4) (1975) 303–320.
- [27] P.L. Miller, Critiquing anesthetic management: the "ATTENDING" computer system, *Anesthesiology* 58 (4) (1983) 362–369.
- [28] P.L. Miller, Extending computer-based critiquing to a new domain: ATTENDING, ESSENTIAL-ATTENDING, and VQ-ATTENDING, *Int. J. Clin. Monit. Comput.* 2 (3) (1986) 135–142.
- [29] R.A. Miller, H.E. Pople Jr., J.D. Myers, Internist-1, an experimental computer-based diagnostic consultant for general internal medicine, *N. Engl. J. Med.* 307 (8) (1982) 468–476.
- [30] G.O. Barnett, J.J. Cimino, J.A. Hupp, E.P. Hoffer, DXplain, An evolving diagnostic decision-support system, *JAMA* 258 (1) (1987) 67–74.
- [31] MGH Laboratory of Computer Science, DXplain, Available from <http://www.lcs.mgh.harvard.edu/dxplain.asp> (cited on December 1, 2006) 2005.
- [32] G.J. Kuperman, R.M. Gardner, T.A. Pryor, HELP: A Dynamic Hospital Information System, Springer-Verlag, New York, 1991.
- [33] D.F. Sittig, R.M. Gardner, N.L. Pace, A.H. Morris, E. Beck, Computerized management of patient care in a complex, controlled clinical trial in the intensive care unit, *Comput. Methods Programs Biomed.* 30 (2–3) (1989) 77–84.
- [34] E.F. Lepage, R.M. Gardner, R.M. Laub, O.K. Golubjatnikov, Improving blood transfusion practice: role of a computerized hospital information system, *Transfusion* 32 (3) (1992) 253–259.
- [35] R.M. Gardner, O.K. Golubjatnikov, R.M. Laub, J.T. Jacobson, R.S. Evans, Computer-critiqued blood ordering using the HELP system, *Comput. Biomed. Res.* 23 (6) (1990) 514–528.
- [36] R.S. Evans, S.L. Pestotnik, D.C. Classen, et al., A computer-assisted management program for antibiotics and other anti-infective agents, *N. Engl. J. Med.* 338 (4) (1998) 232–238.
- [37] C.J. McDonald, R. Murray, D. Jeris, B. Bhargava, J. Seeger, L. Blevins, A computer-based record and clinical monitoring system for ambulatory care, *Am. J. Public Health* 67 (3) (1977) 240–245.
- [38] C.J. McDonald, Protocol-based computer reminders, the quality of care and the non-perfectability of man, *N. Engl. J. Med.* 295 (24) (1976) 1351–1355.
- [39] A. Geissbuhler, R.A. Miller, Clinical application of the UMLS in a computerized order entry and decision-support system, *Proc. AMIA Symp.* (1998) 320–324.
- [40] J.M. Teich, C.D. Spurr, S.J. Flammini, et al., Response to a trial of physician-based inpatient order entry, *Proc. Annu. Symp. Comput. Appl. Med. Care* (1993) 316–320.
- [41] D.W. Bates, J.M. Teich, J. Lee, et al., The impact of computerized physician order entry on medication error prevention, *J. Am. Med. Inform. Assoc.* 6 (4) (1999) 313–321.
- [42] T.G. Thompson, D.J. Brailer, The Decade of Health Information Technology: Delivering Consumer-centric and Information-rich Health Care, US Department of Health and Human Services, Washington, DC, 2004.
- [43] A.K. Jha, J.B. Perlin, K.W. Kizer, R.A. Dudley, Effect of the transformation of the Veterans Affairs Health Care System on the quality of care, *N. Engl. J. Med.* 348 (22) (2003) 2218–2227.
- [44] Health Level 7, Arden Syntax for Medical Logic Systems Standard Version 2.5, American National Standards Institute, Washington, DC, 2004.
- [45] G. Hripcsak, Arden Syntax for medical logic modules, *MD Comput.* 8 (2) (1991) 76,8.

- [46] G. Hripcsak, P. Ludemann, T.A. Pryor, O.B. Wigertz, P.D. Clayton, Rationale for the Arden Syntax, *Comput. Biomed. Res.* 27 (4) (1994) 291–324.
- [47] R.A. Jenders, G. Hripcsak, R.V. Sideli, et al., Medical decision support: experience with implementing the Arden Syntax at the Columbia-Presbyterian Medical Center, *Proc. Annu. Symp. Comput. Appl. Med. Care* (1995) 169–173.
- [48] H.C. Karadimas, C. Chailloleau, F. Hemery, J. Simonnet, E. Lepage, Arden/J: an architecture for MLM execution on the Java platform, *J. Am. Med. Inform. Assoc.* 9 (4) (2002) 359–368.
- [49] Health Level 7, Arden Syntax for Medical Logic Systems Standard Version 2.6, Health Level 7, Ann Arbor, MI, 2007.
- [50] L. Ohno-Machado, J.H. Gennari, S.N. Murphy, et al., The guideline interchange format: a model for representing guidelines, *J. Am. Med. Inform. Assoc.* 5 (4) (1998) 357–372.
- [51] D. Wang, M. Peleg, S.W. Tu, et al., Design and implementation of the GLIF3 guideline execution engine, *J. Biomed. Inform.* 37 (5) (2004) 305–318.
- [52] M. Peleg, A.A. Boxwala, S. Tu, et al., The InterMed approach to sharable computer-interpretable guidelines: a review, *J. Am. Med. Inform. Assoc.* 11 (1) (2004) 1–10.
- [53] OpenClinical, Summaries of guideline representation methods, Available from <http://www.openclinical.org/gmmsummaries.html> (cited on May 3, 2006) 2006.
- [54] C.G. Parker, R.A. Rocha, J.R. Campbell, S.W. Tu, S.M. Huff, Detailed clinical models for sharable, executable guidelines, *Medinfo 11 (Pt 1)* (2004) 145–148.
- [55] P. Ram, D. Berg, S. Tu, et al., Executing clinical practice guidelines using the SAGE execution engine, *Medinfo 11 (Pt 1)* (2004) 251–255.
- [56] P.D. Johnson, S.W. Tu, M.A. Musen, I. Purves, A virtual medical record for guideline-based decision support, *Proc. AMIA Symp.* (2001) 294–298.
- [57] K. Kawamoto, D.F. Lobach, Design, implementation, use, and preliminary evaluation of SEBASTIAN, a standards-based web service for clinical decision support, *Proc. AMIA Symp.* (2005).
- [58] Health Level 7, Patient Evaluation Service Draft Standard, Health Level 7, Ann Arbor, MI, 2005.
- [59] D. Lobach, K. Kawamoto, inventors, Clinical Decision Support System [provisional patent application], World Intellectual Property Organization, Available from <http://www.wipo.int/pctdb/en/wo.jsp?wo=2007030425> (cited on June 01, 2007) 2006.
- [60] K. Kawamoto, D.F. Lobach, Proposal for fulfilling strategic objectives of the U.S. roadmap for national action on clinical decision support through a service-oriented architecture leveraging HL7 services, *J. Am. Med. Inform. Assoc.* 14 (2) (2007) 146–155.
- [61] P. Nadkarni, R. Miller, Service-oriented architecture in medical software: promises and perils, *J. Am. Med. Inform. Assoc.* 14 (2) (2007).
- [62] J. Lamont, Decision support systems prove vital to healthcare, *KMWorld*, February 01, 2007.
- [63] J. Glaser, S. Flammioni, The Service-Oriented Solution for Health IT, *HHN Most Wired*, November 09, 2006.
- [64] HL7 Services Specification Project Workgroup, OMG Healthcare Domain Task Force, Healthcare Services Specification Project: The Business Case and Importance of Services (presentation), Available from http://hssp.wikispaces.com/space/showimage/2007-01_07_HSSP_Public_Slide_Deck_Version_1.2.ppt (cited on February 09, 2007) 2005.
- [65] A. Wright, SANDS: an architecture for clinical decision support in a National Health Information Network, Dissertation, Oregon Health & Science University; Portland, OR: 2007.
- [66] V. Kashyap, A. Morales, T. Hongsermeier, On implementing clinical decision support: achieving scalability and maintainability by combining business rules and ontologies, *AMIA Annu. Symp. Proc.* (2006) 414–418.
- [67] H.S. Goldberg, M. Vashevko, A. Pastilnik, K. Smith, N. Plaks, B.M. Blumenfeld, Evaluation of a commercial rule engine as a basis for a clinical decision support service, *AMIA Annu. Symp. Proc.* (2006) 294–298.