System-Agnostic Clinical Decision Support Services: Benefits and Challenges for Scalable Decision Support

Kensaku Kawamoto[*], Guilherme Del Fiol, Charles Orton and David F. Lobach

Division of Clinical Informatics, Department of Community and Family Medicine, Box 2914, Duke University Medical Center, Durham, NC 27710, USA

Abstract: System-agnostic clinical decision support (CDS) services provide patient evaluation capabilities that are independent of specific CDS systems and system implementation contexts. While such system-agnostic CDS services hold great potential for facilitating the widespread implementation of CDS systems, little has been described regarding the benefits and challenges of their use. In this manuscript, the authors address this need by describing potential benefits and challenges of using a system-agnostic CDS service. This analysis is based on the authors' formal assessments of, and practical experiences with, various approaches to developing, implementing, and maintaining CDS capabilities. In particular, the analysis draws on the authors' experience developing and leveraging a system-agnostic CDS Web service known as SEBASTIAN. A primary potential benefit of using a system-agnostic CDS service is the relative ease and flexibility with which the service can be leveraged to implement CDS capabilities across applications and care settings. Other important potential benefits include facilitation of centralized knowledge management and knowledge sharing; the potential to support multiple underlying knowledge representations and knowledge resources through a common service interface; improved simplicity and componentization; easier testing and validation; and the enabling of distributed CDS system development. Conversely, important potential challenges include the increased effort required to develop knowledge resources capable of being used in many contexts and the critical need to standardize the service interface. Despite these challenges, our experiences to date indicate that the benefits of using a system-agnostic CDS service generally outweigh the challenges of using this approach to implementing and maintaining CDS systems.

Keywords: Clinical decision support, decision support service, Web service, SEBASTIAN, service-oriented architecture.

1. INTRODUCTION

Recent studies have shown that traditional approaches to health care are inadequate for ensuring patient safety and care quality [1,2]. A highly promising strategy for addressing this urgent problem is the use of computer-based clinical decision support (CDS) systems, which provide clinicians, patients and other health care stakeholders with pertinent knowledge and/or person-specific information, intelligently filtered or presented at appropriate times, to enhance health and health care [3]. Indeed, when delivering patient-specific, actionable recommendations to clinicians automatically at the point of care, computer-based CDS systems consistently lead to significant improvements in clinical care [4].

While health informaticists have demonstrated repeatedly how CDS systems can improve health and health care [4], most patient care continues to be conducted with minimal CDS, if any [3]. Although many factors contribute to this limited availability of CDS capabilities [3], one important reason is the difficulty of sharing machine-executable medical knowledge across applications and care settings [3-6]. To address this challenge, attempts have been made to standardize how machine-executable medical knowledge is represented. For example, both the Arden Syntax [7] and

GELLO [8] are Health Level 7 (HL7) standards for representing clinical rules, and the Guideline Interchange Format (GLIF) [9] was designed as a sharable and machine-executable representation format for clinical practice guidelines. Despite these efforts, however, no single knowledge representation approach has been widely adopted across the health informatics community to date, and it is unlikely that such a consensus will be reached in the near future.

As an alternative and complementary strategy for knowledge sharing, CDS capabilities – and in particular the analysis of patient data to generate patient-specific inferences – can be encapsulated within independent, system-agnostic software services and then leveraged by various downstream applications. In this manuscript, such CDS services that provide patient evaluation capabilities independent of specific CDS systems or system deployment settings are referred to as system-agnostic or system-independent CDS services. A software architecture based on the coordinated use of such independent business services is known as a service-oriented architecture (SOA) [10,11]. The use of SOA is growing rapidly within enterprises across various industries, with 68% of enterprises recently surveyed by Forrester Research reporting current or planned SOA adoption by the end of 2010 [12].

As prime examples of system-agnostic CDS services, First DataBank [13] and Cerner® Multum [14] provide their machine-executable knowledge related to medications as system-independent services through application

*Address correspondence to this author at the Division of Clinical Informatics, Department of Community and Family Medicine, Duke Institute for Genome Sciences and Policy, Box 2914, Duke University Medical Center, Durham, NC 27710, USA; Tel: (919) 684-2340; Fax: (919) 684-8675; E-mail: kawam001@mc.duke.edu

programming interfaces. Moreover, we have developed a system-agnostic, standards-based CDS service for supporting this approach to CDS implementation across any medical domain. This system-agnostic CDS service is known as SEBASTIAN, and it allows machine-executable medical knowledge to be encoded in discrete modules and then leveraged across applications and care settings [5]. SEBASTIAN is implemented as a Web service, in which the client and server communicate over the Internet using extensible markup language (XML) messages [15].

SEBASTIAN is the core decision engine being used to enable multiple operational CDS systems, including the point-of-care chronic disease management system for the Duke University Health System [16], the enterprise care quality reporting system for the Duke University Health System, multiple population health management interventions for Medicaid beneficiaries residing in a six-county region in North Carolina [17], a results-based breast cancer management system at a large tertiary care hospital in Argentina [18], and multiple additional CDS systems under development. Furthermore, the SEBASTIAN service interface has served as the basis of the HL7 Decision Support Service specification [19], which specifies a standard service interface for system-agnostic CDS services and was adopted as an international draft standard in 2006. Also of note, a detailed Web service specification based on the HL7 draft standard was adopted by the Object Management Group (OMG) in December 2009 as a part of a joint effort between HL7 and OMG to specify standard interfaces for software services important to health care [20,21].

The widespread use of system-agnostic CDS services could significantly facilitate the more widespread adoption of advanced CDS capabilities, especially if (i) such CDS services were deployed in the context of an overall SOA and (ii) the services implemented standard interfaces and semantics. Indeed, we have previously proposed how such an architecture based on the use of standard Decision Support Services and other standard services could fulfill the strategic objectives of the *Roadmap for National Action on CDS* commissioned by the U.S. Office of the National Coordinator for Health Information Technology [6].

Despite the great potential for system-independent CDS services to enable the deployment of CDS capabilities on a widespread scale, relatively little has been described regarding practical considerations on the use of this approach to CDS implementation. Therefore, we present in this manuscript an analysis of the benefits and challenges of implementing CDS using this architectural pattern, as well as potential approaches to overcoming the challenges identified.

The assessment provided in this manuscript is based primarily on the authors' experience designing, implementing, and operationally leveraging the SEBASTIAN CDS Web service described earlier [5,16-18]. In addition, this analysis is informed by KK and GDF's experience leading the development of several relevant health IT standards as co-chairs of the HL7 CDS Work Group, including the HL7 Decision Support Service draft standard [19], the OMG Decision Support Service standard [20], and the HL7 Infobutton standard [22]. Also, GDF was

actively involved in enterprise knowledge management efforts at Intermountain Healthcare [23], and KK has analyzed alternate CDS implementation architectures to synthesize lessons learned from the adoption patterns of various approaches to CDS implementation [24]. Thus, this analysis is based largely on operational implementation experiences with a service-oriented approach to CDS, supplemented with theoretical analyses and significant experience developing relevant standards in this area.

2. EXAMPLE CDS ARCHITECTURES USING SYSTEM-AGNOSTIC CDS SERVICES

In order to facilitate the discussion of the benefits and challenges of using system-agnostic CDS services, we provide in Fig. (**1**) high-level architectures for two example CDS systems that utilize the design pattern of interest. In this figure, letters correspond to high-level system components, and numbers refer to the system processes described below. These example system architectures are based on the approach we have used to implement two similar, operational CDS systems [16,17]. In considering these examples, it is important to note that these architectures represent one of potentially many valid approaches for leveraging a system-agnostic CDS service, as a central principle for SOA-based design is the ability to flexibly orchestrate the use of various services to efficiently meet business needs. Also of note, details such as caching and multi-threading are intentionally omitted from these examples to ensure that the larger architectural vision is not obscured by excessive technical details.

System Components. Two distinct CDS systems are supported by a common CDS service in this example: a point-of-care disease management module of an electronic health record (EHR) system (component A in the figure), as well as a population health management system (component B). Services that enable these CDS systems include data access services for retrieving relevant clinical data (components C and D); a terminology service enabled by a terminology resource such as the National Library of Medicine's Unified Medical Language System (component E) [25]; a common CDS service such as SEBASTIAN (component F); other CDS services, such as a commercial medication CDS service (component G); a service for converting structured content into formatted documents such as HTML pages and PDF documents (component H); and a service for supporting tele-communications such as email, fax, and page (component I).

As noted in the figure, one CDS service can leverage another CDS service (components F and G). Moreover, a terminology service can be used by multiple system components. For example, consider a case in which a CDS service (component F) defines required data on patient problems in terms of SNOMED CT [26] and ICD9 [27] concepts, and an EHR system (component A) defines problem list data in terms of ICD10 [27] concepts. In this example, the terminology service could be used by the CDS service to identify all SNOMED CT codes that are descendants of the top-level concept of diabetes mellitus, or to translate this SNOMED-based set of concepts to a comparable ICD9-based set of concepts. Moreover, the data access layer of the EHR system could use a terminology
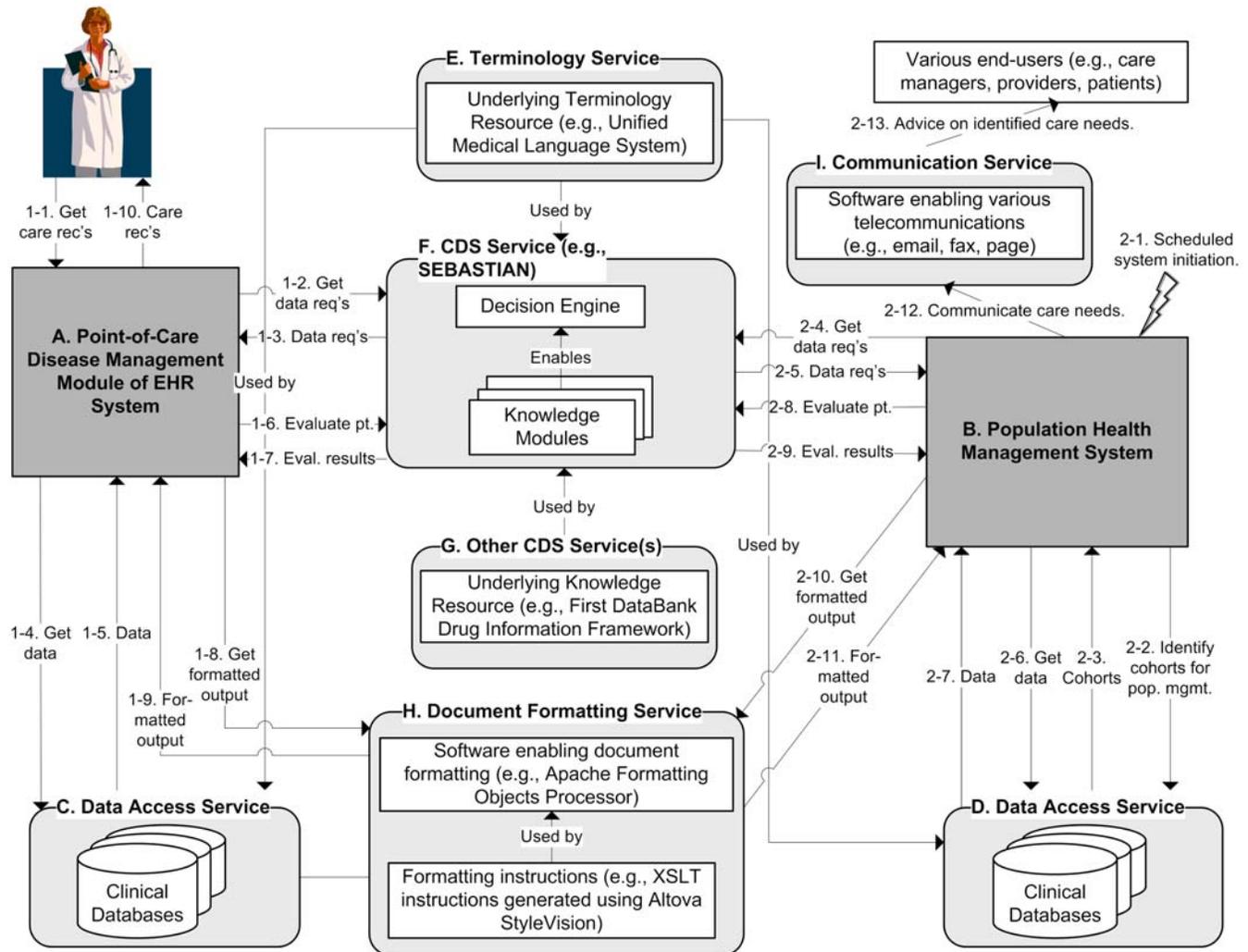
Fig. (1). Sample CDS system architectures using system-agnostic CDS services. Letters and numbers refer to system components and processes referenced in the text.

service to translate problems encoded in ICD10 into appropriate ICD9 concepts for the purposes of communicating with the CDS service. Of note, the degree to which a CDS service minimizes the terminology manipulations required by a client (e.g., by allowing clients to submit patient data using a large variety of terminologies) is an important consideration when designing a CDS service. In the case of SEBASTIAN, we have generally sought to provide explicit support for terminologies in common use in order to limit the need for clients to perform terminology manipulations on their natively available data [5].

Process Flow for Example Disease Management System using Synchronous Interaction Model. In this example, a system-agnostic CDS service is used in a synchronous manner. Here, the CDS process begins when a clinician seeks on-demand disease management recommendations for a patient within an EHR system (process 1-1 in the figure). When the clinician invokes the disease management module (DMM) of the EHR system, the DMM communicates with the CDS service to identify the patient data required for evaluating the patient (processes 1-2 and 1-3), retrieves the required patient data from relevant

clinical databases (processes 1-4 and 1-5), and submits the patient data to the CDS service for evaluation (process 1-6). The CDS service then uses its knowledge modules, as well as any relevant external CDS services, to analyze the submitted patient data and to identify the patient's care needs. Following this analysis, the CDS service communicates its evaluation results back to the DMM (process 1-7). The DMM then compiles the care recommendations into a structured document (e.g., an XML document which lists the patient's care needs, required actions, relevant data, and the underlying clinical guideline). Next, the DMM uses a document formatting service to convert the structured document into an appropriately rendered, human-readable output (e.g., an HTML page) (processes 1-8 and 1-9). The appropriately formatted care recommendations are then presented to the clinician for review (process 1-10).

Process Flow for Example Population Health Management System using Asynchronous Interaction Model. In this second example, the population health management system (PHMS) is invoked asynchronously on a scheduled basis (process 2-1). As the first step, the PHMS

queries its clinical databases to identify cohorts of patients for population health management (e.g., patients with a targeted disease) (processes 2-2 and 2-3). Then, for each patient identified, the PHMS determines the data required by the CDS service to identify the patient's care needs (processes 2-4 and 2-5), obtains the required data through its data access layer (processes 2-6 and 2-7), and uses the CDS service to identify each patient's care needs (processes 2-8 and 2-9). Then, following the evaluation of all individuals in the cohort, the PHMS uses the patient-level evaluation results to generate population-level CDS interventions (e.g., patient follow-up lists for community-based care managers, clinic-based feedback reports, and patient-directed care reminder letters; processes 2-10 and 2-11) [17]. These population-level interventions are then communicated to appropriate end-users through a telecommunication service (processes 2-12 and 2-13).

With these example CDS architectures in mind, the remainder of the manuscript discusses benefits and challenges of using system-agnostic CDS services to implement various CDS applications. Of note, our analysis of benefits and challenges is largely based on our implementation experiences and theoretical analyses, rather than on empirical studies directly comparing service-oriented versus non-service-oriented approaches to CDS deployment. Therefore, these benefits and challenges should be considered potential benefits and challenges that will hopefully serve as useful insights but which require further validation and investigation. At the same time, our analysis is fully aligned with a well-respected analysis of the benefits and challenges of SOA in general [10].

As a final note on the analysis that follows, for both benefits and challenges, these aspects of system-agnostic CDS services are not exclusive to the use of this design pattern. For example, the potential for centralized knowledge management and sharing is a potential benefit of using system-agnostic CDS services, but such scalable knowledge management and sharing could also potentially be achieved using a different design pattern, such as one based on the use of standardized knowledge resources such as Arden Syntax modules.

3. BENEFITS

Potential benefits of using a system-agnostic CDS service are outlined in Table **1** and described in further detail below. For each benefit, examples are provided to ground the benefit in practical terms.

Relative Ease and Flexibility of Use. A primary potential benefit of using system-agnostic CDS services is the relative ease and flexibility with which such services can be used to facilitate the implementation of CDS capabilities across information systems and clinical care settings. Because system-agnostic CDS services make minimal assumptions about how they will be utilized, CDS system implementers have significant flexibility in designing how a CDS application makes use of the service to meet end-user needs. Moreover, because system-agnostic CDS services are self-contained, designed for external integration, and make minimal assumptions about their deployment context, they can be integrated into various applications with relative ease. For example, using SEBASTIAN, one of the authors (KK) was able to develop a prototype chronic disease management system and three population health management interventions by himself in approximately six months [5]. Such rapid development was possible because SEBASTIAN was able to fulfill a core functional requirement of the CDS systems – namely, the analysis of patient data to generate patient-specific inferences. As a result, the downstream applications only needed to concern themselves with the

Table 1.   Potential Benefits of Using a System-Agnostic CDS Service

Potential Benefit	Examples
Relative ease and flexibility with which a CDS service can be leveraged across applications and care settings to implement CDS capabilities	Using SEBASTIAN, four CDS applications were developed across two care settings by one health informaticist (KK) in about six months [5] Commercial medication knowledge resources can be adapted for use within various CDS applications
Facilitation of centralized knowledge management and sharing	Commercial medication knowledge vendors manage knowledge centrally for large numbers of client institutions SEBASTIAN supports multiple CDS applications deployed across divergent care settings [5,16,17]
Potential to support multiple underlying knowledge representations and knowledge resources through a common service interface	SEBASTIAN can leverage any knowledge resource accessible through the Java programming language [5] HL7 and OMG Decision Support Service standards do not restrict the underlying knowledge representation formalism [19,20]
Improved simplicity and componentization (separation of concerns)	A system-agnostic CDS service does not need to be concerned with when it is leveraged; how required data are retrieved; or how patient-specific inferences are communicated to end-users [24] A system-agnostic CDS service can be loosely coupled with other functionally independent system components and services to fulfill various CDS needs [16,17]
Easier testing and validation	SEBASTIAN test cases can focus solely on the underlying clinical decision logic [5] SEBASTIAN clinical decision rules can undergo batch regression testing [5]
Enabling of distributed CDS development	Medication CDS capabilities are developed across numerous vendors and institutions using commercial medication knowledge bases Duke enterprise care quality reporting system developed using SEBASTIAN, but with minimal need for involvement by SEBASTIAN development team

other tasks related to CDS delivery – the identification of when to evaluate patients, the retrieval of relevant patient data from clinical data repositories, and the appropriate communication of the patient-specific inferences to end-users following interaction with SEBASTIAN [5,24]. Today, the four CDS systems developed in this manner continue to be used to support hundreds of clinical users on an operational basis.

As another example of the relative ease and flexibility with which system-agnostic CDS services can be leveraged, consider the example of commercial medication knowledge resources. In our experience, it took an experienced health informaticist (KK) less than a week to gain a working knowledge of the application programming interface to a commercial medication knowledge service and to configure a CDS system to make use of the service. Moreover, commercial medication knowledge services impose few constraints on how they are to be used. Therefore, these resources can be leveraged for such varied downstream CDS applications as electronic prescribing systems, pharmacy error checking modules, and patient safety surveillance systems. While medication knowledge services cover a limited medical domain and therefore cannot be considered to be a direct analogue to a more broadly scoped CDS service such as SEBASTIAN, we do believe that these medication knowledge services provide excellent examples of how system-agnostic CDS services can be designed and supported to enable CDS on a widespread scale.

Facilitation of Centralized Knowledge Management and Sharing. Another potential benefit of using system-agnostic CDS services is that this architectural pattern can facilitate the centralized management and sharing of machine-executable medical knowledge. By their very nature, system-agnostic CDS services provide access to knowledge resources that are independent of specific CDS system implementations, as opposed to knowledge resources that incorporate implementation-specific information such as when it should be triggered, how the required patient data should be retrieved, and how the patient-specific inferences should be acted upon [24]. Consequently, each knowledge resource is potentially applicable across a wide range of CDS deployment settings, thereby making the knowledge resources more conducive to centralized management and sharing across multiple deployment contexts. Commercial medication knowledge vendors, for example, maintain their knowledge bases centrally on behalf of a large number of clients. Similarly, centrally managed knowledge within SEBASTIAN is used to support multiple CDS applications deployed across divergent care settings [5,16,17].

Centralized knowledge management has many inherent benefits. These benefits include a reduction in redundant knowledge management efforts, as well as an increased capacity for developing and supporting effective knowledge management processes and tools. Furthermore, centralized knowledge management enables an increase in the effort and resources that can be dedicated towards the development of the knowledge base. Also, centralized knowledge management can improve quality control due to increased vetting of the knowledge by both the creators of the knowledge as well as by the many users of the knowledge. Moreover, centralized knowledge management enables the

development of knowledge components that can be reused across many different decision support modules. Examples of such reusable knowledge components include clinical decision rules that detect if a patient has a specific condition or calculate derived patient attributes such as body mass index or creatinine clearance.

Potential to Support Multiple Knowledge Representations and Resources through Common Service Interface. As another important benefit, system-agnostic CDS services have the potential to support a variety of knowledge representation formalisms and knowledge resources through a common service interface. For example, SEBASTIAN can leverage any knowledge resource that can be accessed through the Java programming language [5], and the HL7 and OMG Decision Support Service specifications place no constraints on the underlying knowledge representation formalisms that are used to instantiate the service [19,20]. Thus, for example, a Decision Support Service compliant with the HL7 and OMG standards may use rules-based knowledge as well as knowledge based on decision trees, probabilistic systems, and/or neural networks, as long as external systems can interact with the service through a standard service interface. Given the significant heterogeneity that currently exists, and is likely to continue to exist, in how machine-executable medical knowledge is represented, this flexibility is an important potential advantage for a services-based approach to CDS.

Improved Simplicity and Componentization (Separation of Concerns). As any experienced CDS implementer can attest, implementing CDS can require intricate coordination among various health information systems and knowledge resources. Thus, if CDS capabilities are implemented without an explicit intention to define and separate out functions into independent components through a process known as "separation of concerns," a CDS implementation can quickly become highly complex and difficult to maintain. By explicitly separating out patient data analysis and inferencing as a separate and independent functional component of an overall CDS system, the use of system-agnostic CDS services can help to simplify the CDS deployment architecture. For example, whereas a clinical decision engine under other CDS architectures may need to include specifications on such issues as when it is to be triggered, how required data are to be retrieved, and how patient-specific inferences are to be communicated to end-users [24], a clinical decision engine implemented as a system-agnostic CDS service needs to only be concerned with how to evaluate patient data to generate patient-specific inferences. Moreover, as outlined in Fig. (**1**), because a system-agnostic CDS service is a functionally independent system component, it can be loosely coupled with other functionally independent system components and services to fulfill a given CDS need such as point-of-care disease management [16], while at the same time being coupled with a different set of system components to fulfill a different CDS need such as population health management [17].

Easier Testing and Validation. As a corollary to the improved simplicity and componentization just discussed, the modular and functionally independent nature of system-agnostic CDS services can reduce the effort required for testing and validation. For example, SEBASTIAN test cases

can focus on verifying that knowledge modules return patient-specific assessments that are consistent with the underlying clinical decision logic, rather than needing to be concerned with context-specific issues such as whether the resulting assessments were communicated to the appropriate end-users [5]. Moreover, combined with SEBASTIAN's capability to evaluate patient data "as if" it was any specified date and time, over a thousand static test cases can be run against SEBASTIAN rules in batch to enable comprehensive regression testing within a matter of minutes [5]. Following such highly scalable testing of the core decision logic, more labor-intensive testing is conducted at the application level by human users to verify the appropriate functioning of the application as a whole.

Enabling of Distributed CDS Development. Finally, an additional potential benefit of using system-agnostic CDS services is that this approach can enable the distributed development of application-level CDS capabilities. Whereas the knowledge itself in a CDS service must undergo central coordination and quality control, the development of downstream CDS applications can be conducted in a highly decentralized and distributed manner. For example, medication CDS capabilities are developed by various independent vendors and health care institutions using commercial medication knowledge bases. As another example, SEBASTIAN patient evaluation capabilities developed for the purposes of supporting point-of-care chronic disease management [16] were efficiently leveraged by a separate project team to generate enterprise care quality reports, with minimal need for involvement by the SEBASTIAN development team. Thus, an important benefit of the use of this approach to CDS is that available health informatics resources can be more efficiently leveraged to develop and deploy CDS capabilities in a scalable manner.

4. CHALLENGES AND POTENTIAL SOLUTIONS

Despite the many potential benefits to using a system-agnostic CDS service, several potential challenges must also be considered. Table **2** outlines these potential challenges, along with descriptive examples and potential solutions. These challenges and potential approaches to overcoming these challenges are described below.

Increased Effort Required for Developing and Supporting Knowledge Resources. Because knowledge resources within a context-independent CDS service are designed to support a variety of CDS applications across multiple clinical contexts, developing and supporting such knowledge resources can require substantially more effort than a similar resource designed to support a specific CDS application in a specific implementation setting. For example, whereas a given CDS application may only need to determine if a woman who is 40 years of age or older is overdue for a mammogram, a context-independent knowledge resource designed to support this and other CDS applications would need to consider providing additional information that may be needed for other CDS applications, such as the date and result of the last mammogram, and whether the mammogram is almost due (which, for example, may trigger a reminder letter to be sent in a different CDS system). Furthermore, whereas the need for past mammograms may only need to be identified in terms of

concept set A in terminology A' for the particular deployment setting, a context-independent knowledge resource would need to consider describing the need for mammograms in terms of concept sets A through Z in terminologies A' through Z', including standard terminologies such as SNOMED CT [26]. As noted earlier in the discussion of the sample CDS architectures, the terminologies supported by a CDS service, as well as the manner in which terminology services are leveraged, are both important considerations when designing and implementing system-agnostic CDS services.

In addressing the need for increased effort to make underlying knowledge resources more capable of deployment across applications and care settings, one potential solution is to balance generalizability with the realities of resource availability. For example, whereas the above knowledge resource may eventually need to have its concepts mapped to all terminologies of significance used within the global health care community, a reasonable approach would be to support major standard terminologies in conjunction with any non-standard terminologies that are in use by existing clients. Furthermore, to address the need for additional effort, the development cost for knowledge resources could be spread out across multiple deployment settings to support the increased effort requirement.

Need for Standardized Service Interface. Another important challenge to the use of system-agnostic CDS services is that the full potential of such services may only be realized if different services use a common interface for communicating with clients. Without such standardization, clients may need to manage multiple, non-compatible interfaces to various CDS services. For example, current CDS systems designed to use one commercial medication knowledge service cannot be easily adapted to use a different knowledge service, as these services utilize proprietary application programming interfaces. Moreover, a similar problem with interoperability would arise if a CDS system currently designed to use SEBASTIAN needed to access a different CDS service with a non-compatible service interface.

The HL7 and OMG Decision Support Service standards [19,20] begin to address this need for standardization by defining a standard interface for system-agnostic CDS services, including a mechanism for defining common service semantics through the use of profiles. The widespread adoption of these standards would significantly facilitate the ability of system-agnostic CDS services to enable CDS on a much larger scale.

Need for Customization. As another potential challenge, the outputs of CDS services may need to be customized to meet the unique needs of client systems in various contexts of use. For example, different clients may have different preferred screening intervals for the primary prevention of cancer. Similarly, different health care organizations may wish to base the clinical guidance provided to their clinicians on different clinical practice guidelines. Even within the same organization, different workflows, preferences, and clinical protocols may require a CDS service to respond in slightly different ways. In addressing this need for customization, one potential solution is to incorporate customization parameters as inputs into the service. For

Table 2.   Potential Challenges to Using a System-Agnostic CDS Service

Potential Challenge	Examples	Potential Solution
Increased effort required to develop and support knowledge resources for use in multiple contexts	A given knowledge resource may be used for very different types of applications A given knowledge resource may be used in settings using different terminologies	Balance generalizability with resource realities Spread knowledge development cost over multiple deployment settings to support increased effort requirement
Need for service interface to be standardized	A CDS system designed to use one commercial medication knowledge service cannot be easily adapted to use a different knowledge service A CDS system designed to use SEBASTIAN cannot be easily adapted to use a different CDS service with a non-compatible service interface	Facilitate widespread use of HL7 and OMG Decision Support Service standards [19,20]
Service output may need to be customized to meet the needs of clients	Clients may differ in the preferred screening frequency for primary cancer prevention Different health care organizations may wish to provide care guidance according to different clinical practice guidelines	Incorporate customization parameters (e.g., preferred screening frequency for mammograms) as service inputs
CDS service fulfills only one of the several tasks required for delivering CDS	A population health management system using a system-agnostic CDS service still needs to determine who should be evaluated, how patients should be triaged, and how identified care needs should be addressed A point-of-care CDS system using a system-agnostic CDS service still needs to determine when it should be invoked, how data are retrieved, and how care recommendations are communicated	Develop other system components in a modular, scalable, standards-based, and service-oriented manner
"Black-box" nature of service may be unacceptable	A client may wish to know exactly how a clinical decision was reached for cancer chemotherapy A client may require that underlying clinical logic be formally reviewed prior to operational use	Provide detailed meta-data on underlying clinical algorithms Make underlying service implementation open-source
Clients may insist on having local service instance	A client may be uncomfortable relying on a CDS service hosted externally A client may wish to avoid transmitting patient data to a third party A client may wish to minimize the time required for transmitting data to and from the CDS service	Service instances may be deployed to clients and synchronized
Need to account for different data availability and data models	Some clients may have access to only claims data, while others may have access to claims and laboratory data, and still other may have access to various types of electronic health record data Different clients may use different information models and different terminologies	Standardize expected data availability for CDS, along with associated information models and terminologies Create different sets of knowledge resources for different data availability contexts Structure knowledge resources to accommodate different data availability
Limited content availability	Outside of basic medication knowledge resources, only limited clinical content is available through system-agnostic CDS services Fundamental CDS capabilities (e.g., for calculating pediatric vital sign percentiles) are not generally available through CDS services	Create an interoperable, standards-based market for such knowledge Provide federal funding for the development of clinical content for system-agnostic CDS services

example, the client's preferred screening interval for breast cancer prevention may be provided as an input to a CDS service to influence its determination of whether a patient is in need of a screening mammogram.

Need for Other System Components. By design, a system-agnostic CDS service fulfills only one of the tasks required for delivering CDS – namely, the evaluation of patient data to generate patient-specific inferences. Consequently, other system components must be established for delivering CDS, including components for interacting with users, retrieving required clinical data, and parsing and presenting CDS service results. For example, a population health management system that uses a system-agnostic CDS service to identify the care needs of individual patients still needs to include system components that determine how a number of issues are resolved, such as when patients are evaluated; which patients are included in the evaluation; how relevant patient data are retrieved; how patients are triaged; and how identified care needs are addressed and communicated (Fig. **1**). While less complex, similar tasks related to system initiation, data retrieval, and end-user communication would need to be addressed by point-of-care CDS systems such as the one described earlier (Fig. **1**). Given the potential complexity of these various tasks, substantial additional system components may be required to

optimally fulfill end-user needs using a system-agnostic CDS service.

In developing these various system components beyond the CDS service, it is important that good system design principles are followed. When possible, such system components should be implemented in a manner that allows for efficient maintenance and re-use. One recommended approach for achieving such reuse is to implement needed functionality as independent software services that are leveraged in conjunction with system-agnostic CDS services within a standards-based SOA [6]. Coupled with the use of service interface standards such as the HL7/OMG Decision Support Service standards [19,20], the HL7 Common Terminology Services standard [28], and the HL7/OMG Retrieve, Locate, and Update Service standards [29,30], the CDS architectures described in Fig. (**1**) serve as potential examples for how CDS systems could be implemented using the recommended standards-based approach.

"Black-box" Evaluation may be Unacceptable. Given the potentially high stakes of CDS in many clinical settings, the black-box nature of a CDS service may be unacceptable in certain cases. For example, a client may wish to inspect the underlying CDS code to know exactly how a clinical decision was reached with regard to cancer chemotherapy. Moreover, on a broader level, a given health care institution may require that the underlying clinical logic be formally reviewed by an appropriate committee (e.g., the health system's Pharmacy and Therapeutics committee) prior to the CDS being deployed for operational use.

One potential solution to this challenge is to provide detailed meta-data on how a clinical decision is reached by a CDS service, including a human-readable description of the clinical guideline or algorithm used and a detailed explanation of how input data are processed to generate the service output. Of note, the HL7/OMG Decision Support Service standards [19,20] include a mechanism for adding such detailed meta-data to knowledge modules as needed. Moreover, an additional potential approach to this challenge is to make the underlying service implementation open-source.

Desire for Local Service Installation. As a practical matter, health care organizations and health information technology vendors may insist on having a local service instantiation. This situation may arise, for example, if a client is uncomfortable relying on a CDS service that is hosted externally. Moreover, even though a CDS service can be configured so that no patient identifiers are exchanged, a client may wish to avoid transmitting patient data to a third party. Finally, CDS applications that are accessed synchronously may require performance levels that can only be achieved when the CDS service is hosted locally. To address such concerns, CDS services can be designed for distributed deployment. In such a distributed deployment model, it is important that processes be established to ensure synchronization of the client instances with the centrally managed CDS service.

Different Data Availability and Data Models. As with any CDS deployment strategy that spans diverse deployment settings, a significant challenge to the use of system-agnostic CDS services is that there can be significant heterogeneity in the information models and terminologies used at different care settings, as well as significant heterogeneity in the types of data available across these settings [3]. For example, different clients may have access to (i) claims data only; (ii) claims data plus laboratory data; or (iii) various types of electronic health record data. Moreover, even when clients have access to similar types of electronic data, these data may use different information models and terminologies. This heterogeneity of the data can make development of CDS resources that can be used across these different deployment contexts quite challenging.

In addressing this challenge, the most desirable long-term solution is to standardize the data expected to be available for CDS, including which information models and terminologies are used to capture such data. Such an effort is currently underway within the HL7 virtual medical record project [31]. In addition, another potential solution to this challenge is to create different sets of knowledge resources for different contexts of data availability, while factoring out common inferencing capabilities into software modules that can be re-used by related resources. Furthermore, knowledge resources could potentially be designed to accommodate different levels and types of available data.

Limited Content Availability. Finally, an important potential challenge to the use of system-agnostic CDS services is the limited availability of content within such services. Beyond knowledge resources for basic medication inferencing, there are currently only limited medical knowledge resources available through system-agnostic CDS services. For example, when we recently expanded the Duke University Health System's disease management system to provide pediatric vital sign percentiles for height, weight, head circumference, body mass index, and blood pressure, we looked for a public or commercial CDS service that we could leverage for this purpose. However, we were unable to find a suitable CDS service providing this desired capability, and we had to implement the capability in-house within SEBASTIAN. Thus, limited content availability is still a significant current challenge to the use of this architectural pattern from the perspective of a CDS implementer.

One way to address this challenge would be to create an interoperable, standards-based market for such knowledge, in which knowledge resources from various organizations are available through a common Decision Support Service interface. Moreover, federal funding could be provided to stimulate the development of clinical content that is available through standard CDS services either for free or for a nominal charge. Once such a market for standards-compliant CDS services is established, market forces could potentially enable a highly scalable model for the widespread deployment of advanced CDS capabilities.

5. BALANCE OF BENEFITS AND CHALLENGES

In assessing the value of any approach to CDS design and implementation, we believe it is critical that the assessment be primarily based on how the approach facilitates the efficient implementation and maintenance of desired CDS capabilities. Based on this very practical measure, we believe that the benefits of system-agnostic CDS services clearly outweigh the challenges. Indeed, we have found that SEBASTIAN has allowed us to develop,

deploy, and maintain CDS capabilities across various applications and care settings very efficiently, and commercial medication knowledge service providers have also proven the value of this approach through the establishment of large client bases. Thus, the use of system-agnostic CDS services represents a highly promising strategy for enabling robust CDS on a wider scale.

Of note, a potential concern with the use of system-agnostic CDS services may be that the use of a network-based approach to CDS may introduce significant latency that could be a problem for time-sensitive, point-of-care CDS applications. In our experience, we have not found that the use of a CDS service introduces any significant delay. For example, when having SEBASTIAN evaluate a patient with 100 data points (e.g., diagnoses) using 50 disease management knowledge modules, the time required for initiating a network connection to a remote server and receiving a response is only about 150 milliseconds. In fact, when using SEBASTIAN to deliver CDS, a CDS application spends most of its time retrieving patient data from the relevant clinical databases, which is a step that would be required regardless of whether a CDS service was being used. Nevertheless, as previously discussed, clients could maintain and access a local instance of a given CDS service to minimize delays due to network latency.

6. CONCLUSION

At present, the great promise of CDS stands in stark contrast to the limited availability of robust CDS capabilities in most clinical care settings. While many factors certainly contribute to this lack of CDS availability, an important factor is the difficulty of sharing CDS capabilities across applications and care settings. The use of system-agnostic CDS services, and in particular the use of such services within a standards-based SOA, holds great potential for enabling the availability of robust CDS on a much wider scale. By providing a practical, experience-based analysis of the various potential benefits and challenges associated with this approach to deploying CDS capabilities, we hope to have facilitated the realization of this vision for scalable CDS enabled by an open and interoperable marketplace for CDS services. Moving forward, we plan to continue to actively explore this promising approach to realizing improved health and health care through CDS.

ACKNOWLEDGMENTS

CONFLICT OF INTEREST

KK and DFL are co-founders of Clinica, Inc., which is developing an open-source Decision Support Service compliant with the HL7 and OMG Decision Support Service standards discussed in this manuscript. Clinica will make any of its intellectual property required for implementing to the HL7 and OMG Decision Support Service standards available to others for free.

REFERENCES

[1] Kohn LT, Corrigan JM, Donaldson MS, Eds. To err is human: building a safer health system. Washington, DC: National Academy Press 1999.

[2] McGlynn EA, Asch SM, Adams J, *et al.* The quality of health care delivered to adults in the United States. N Engl J Med 2003; 348(26): 2635-45.

[3] Osheroff JA, Teich JM, Middleton B, Steen EB, Wright A, Detmer DE. A roadmap for national action on clinical decision support. J Am Med Inform Assoc 2007; 14(2): 141-5.

[4] Kawamoto K, Houlihan CA, Balas EA, Lobach DF. Improving clinical practice using clinical decision support systems: a systematic review of trials to identify features critical to success. BMJ 2005; 330(7494): 765-8.

[5] Kawamoto K, Lobach DF. Design, implementation, use, and preliminary evaluation of SEBASTIAN, a standards-based Web service for clinical decision support. AMIA Annu Symp Proc 2005: 380-4.

[6] Kawamoto K, Lobach DF. Proposal for fulfilling strategic objectives of the U.S. roadmap for national action on decision support through a service-oriented architecture leveraging HL7 services. J Am Med Inform Assoc 2007; 14: 146-55.

[7] Pryor TA, Hripcsak G. The Arden syntax for medical logic modules. Int J Clin Monit Comput 1993; 10(4): 215-24.

[8] Sordo M, Boxwala AA, Ogunyemi O, Greenes RA. Description and status update on GELLO: a proposed standardized object-oriented expression language for clinical decision support. Stud Health Technol Inform 2004; 107(Pt 1): 164-8.

[9] Boxwala AA, Peleg M, Tu S, *et al.* GLIF3: a representation format for sharable computer-interpretable clinical practice guidelines. J Biomed Inform 2004; 37(3): 147-61.

[10] Erl T. SOA: principles of service design. Crawfordsville, IN: Prentice Hall; 2008.

[11] Krafzig D, Banke K, Slama D. Enterprise SOA: service-oriented architecture best practices. Indianapolis, IN: Prentice Hall PTR; 2004.

[12] Heffner R, Leaver S, An M. Insights for CIOs: SOA and beyond [cited 2010 Jun 14]. Available from: http://www.forrester.com/go?docid=55937.

[13] First DataBank. First DataBank Drug Information Framework [cited 2010 Jun 14]. Available from: http://www.firstdatabank.com/products/di_framework/.

[14] Cerner Multum. addVantageRx [cited 2010 Jun 14]. Available from: http://www.multum.com/addVantageRx.htm.

[15] He H. What is service-oriented architecture? O'Reilly [article on the Internet]. 2003 Sep 30 [cited 2010 Jun 14]. Available from: http://www.xml.com/pub/a/ws/2003/09/30/soa.html.

[16] Lobach DF, Kawamoto K, Anstrom KJ, Russel ML, Woods P, Smith D. Development, deployment and usability of a point-of-care decision support system for chronic disease management using the recently-approved HL7 Decision Support Service standard. Medinfo 2007: 861-5.

[17] Lobach DF, Kawamoto K, Anstrom KJ, *et al.* Proactive population health management in the context of a regional health information exchange using standards-based decision support. AMIA Annu Symp Proc 2007: 473-7.

[18] Borbolla D, Otero C, Lobach DF, *et al.* Evaluación de la sensibilidad y especificidad de un sistema de ayuda a la toma de decisiones como modelo de servicio. Infolac 2008. Pilar, Buenos Aires, Argentina: RevistaeSalud.com; 2008.

[19] Health Level 7. HL7 Decision Support Service specification (Draft Standard for Trial Use) [cited 2010 Jun 14]. Available from: http://www.hl7.org/v3ballot2009jan/html/infrastructure/dss/dss.htm.

[20] Object Management Group. OMG Clinical Decision Support Service standard (adopted beta specification) [cited 2010 Jun 14]. Available from: http://hssp-dss.wikispaces.com/omg_specification.

[21] Kawamoto K, Honey A, Rubin K. The HL7-OMG Healthcare Services Specification Project: motivation, methodology, and deliverables for enabling a semantically interoperable service-

oriented architecture for healthcare. J Am Med Inform Assoc 2009; 16(6): 874-81.

[22]   Health Level 7. HL7 Context-aware Information Retrieval (Infobutton) standard [cited 2010 Jun 14]. Available from: http://www.hl7.org/v3ballot2010may/html/domains/uvds/uvds_Context-awareKnowledgeRetrieval(Infobutton).htm.

[23]   Hulse NC, Rocha RA, Del Fiol G, Bradshaw RL, Hanna TP, Roemer LK. KAT: a flexible XML-based knowledge authoring environment. J Am Med Inform Assoc 2005; 12(4): 418-30.

[24]   Kawamoto K. Integration of knowledge resources into applications to enable clinical decision support: architectural considerations. In: Greenes RA, Ed. Clinical Decision Support: the road ahead. Boston: Elsevier Academic Press 2007; pp. 503-38.

[25]   National Library of Medicine. Unified Medical Language System [cited 2010 Jun 14]. Available from: http://www.nlm.nih.gov/research/umls.

[26]   International Health Terminology Standards Development Organisation. Systematized Nomenclature of Medicine-Clinical Terms (SNOMED CT) [cited 2010 Jun 14]. Available from: http://www.ihtsdo.org/snomed-ct/.

[27]   World Health Organization. International Classification of Diseases (ICD) [cited 2010 Jun 14]. Available from: http://www.who.int/classifications/icd/en/.

[28]   Health Level 7. HL7 Common Terminology Services 2 Service Functional Model (Draft Standard for Trial Use) [cited 2010 Jun 14]. Available from: http://www.hl7.org/v3ballot2010sep/html/infrastructure/cts_r2/cts_r2.htm

[29]   Health Level 7. HL7 Retrieve, Locate, and Update Service specification (Draft Standard for Trial Use) [cited 2010 Jun 14]. Available from: http://www.hl7.org/v3ballot2009jan/html/infrastructure/rlus/rlus.htm.

[30]   Object Management Group. Retrieve, Locate, and Update Service: specification for joint final OMG submission, version 1.07 [cited 2010 Jun 14]. Available from: http://www.omg.org/cgi-bin/doc?health/2008-12-3.

[31]   Health Level 7. HL7 Virtual Medical Record (vMR) Project Wiki [cited 2010 Jun 14]. Available from: http://wiki.hl7.org/index.php?title=Virtual_Medical_Record_(vMR).