

Infrastructure for a clinical- decision-intelligence system

X. S. Wang
L. Nayda
R. Dettinger

Clinical decision intelligence (CDI) is an emerging area in health care, covering a broad range of subjects, from clinical data integration and data analysis to knowledge management and application development. The goal of CDI systems is to improve health-care quality and reduce costs through the discovery, management, and application of clinical intelligence from heterogeneous and rapidly expanding data sources. These sources include data from clinical practice, nursing, health-care management, health-care administration, and medical research. In this paper, we discuss the functional requirements and reference architecture for CDI systems and their clinical applications. This architecture includes an integrated framework for managing the entire CDI process, a standardized enterprise ontology management system, and a clinical knowledge representation platform. The CDI approach has the potential to transform the health-care management process through the integration of business intelligence, business rule management, and business process management in a clinical setting, and to help health-care organizations move closer to an on demand model.

INTRODUCTION

In the mid-1990s, several initiatives that raised awareness of a crisis in the quality of health care in America were launched. Among these initiatives were those presented by the Institute of Medicine^{1,2} (IOM), the Agency for Healthcare Research and Quality³ (AHRQ), the National Association for Healthcare Quality⁴ (NAHQ), the Leapfrog Group,⁵ and the Institute for Healthcare Improvement⁶ (IHI). Among the problems on which these initiatives focused were poor outcomes and inconsistency in the delivery of health care, medical errors, and rapidly increasing health-care costs.

Some initial recommendations resulting from these initiatives emphasized the effective utilization of information technology (IT) as an essential element in improving health-care quality and reducing costs. This health-care IT transformation has many components, including:

- The implementation of electronic health records,
- The secure exchange of medical information,

©Copyright 2007 by International Business Machines Corporation. Copying in printed form for private use is permitted without payment of royalty provided that (1) each reproduction is done without alteration and (2) the Journal reference and IBM copyright notice are included on the first page. The title and abstract, but no other portions, of this paper may be copied or distributed royalty free without further permission by computer-based and other information-service systems. Permission to republish any other portion of the paper must be obtained from the Editor. 0018-8670/07/\$5.00 © 2007 IBM

- Evidence-based decision making that integrates the best available external clinical evidence from systematic research with individual clinical expertise, thus aiding the delivery of logical and consistent medical care from clinician to clinician and from hospital to hospital,
- Aligning payment policy with quality improvements, which provides financial incentives for practitioners to follow quality improvement guidelines and demonstrate improved performance.

Implementing evidence-based medicine⁷ and pay-for-performance programs brings new challenges to health-care IT. To meet these challenges, the field of clinical decision intelligence (CDI) has emerged, covering a broad range of subjects, from clinical data integration and data analysis to knowledge management and application development. In this paper, we describe the infrastructure for a CDI solution that enables the practice of personalized, evidence-based medicine, supports clinical quality improvement and pay-for-performance programs, and facilitates knowledge sharing and practice standardization. The infrastructure provides a standardized platform for knowledge acquisition, management, and application as well as ontology management for the entire enterprise.

The knowledge discovery and acquisition facilities of the CDI solution collect and manage the latest biomedical research and clinical discoveries and provide the foundation for practicing evidence-based medicine. The knowledge management facilities of the CDI system support formal enterprise knowledge-management policies and processes. These facilities allow organizations to manage clinical policies, quality improvement, and pay-for-performance programs at the enterprise level. The decision support facilities of the CDI solution make its knowledge components, rules, and guidelines available as services to applications through a rule engine in a service-oriented architecture (SOA) paradigm. The solution separates policy management and application logic in order to allow an enterprise to quickly adapt to rapidly changing application development requirements. Service-based decision support also facilitates the integration of legacy systems.

The CDI infrastructure supports standardized enterprise ontology management and integration with

existing electronic medical records (EMRs) and clinical information systems (CISes). Integration with EMRs enables the delivery of personalized medical care based on an individual patient's conditions. Standardization helps improve interoperability throughout an entire health-care organization and with external collaborators, including government agencies and community clinics. Standardized, evidence-based, and personalized medical care means safe, effective, patient-centered, timely, efficient, and equitable care for each patient, and improved performance and cost-effective operation for the business. In addition to these benefits, the integration of clinical analytics, knowledge management, and decision support inherent in the CDI solution helps transform health-care organizations into on demand businesses, which can rapidly adapt to exponential data growth and rapidly changing business requirements.

In the remainder of this section, we present an overview of CDI and describe the basic requirements for CDI systems. We then present our proposed CDI solution (as designed and implemented) and its application in two clinical scenarios.

Overview of CDI and related work

The CDI process is illustrated in *Figure 1*. As shown in the figure, CDI systems are comprised of the following three subprocesses: (1) knowledge discovery and acquisition, (2) knowledge management, and (3) decision support.

No system predating CDI integrates these three subprocesses in a single system. Traditionally, researchers in health-care organizations perform research and discovery functions and publish the results, but the systems are not automated, and the results are rarely collected in a centralized repository to be used to support clinical applications. This results in a disconnection between discovery and decision support. Knowledge management implementations vary, but very few systematic knowledge management systems exist in health care, and they are rarely linked to discovery or to decision support. Decision support is embedded in applications and not integrated with the other subprocesses.

The three subprocesses are performed by people with different expertise, interests, and objectives, and each carries its own unique challenges. The subprocesses are inherently interconnected: each

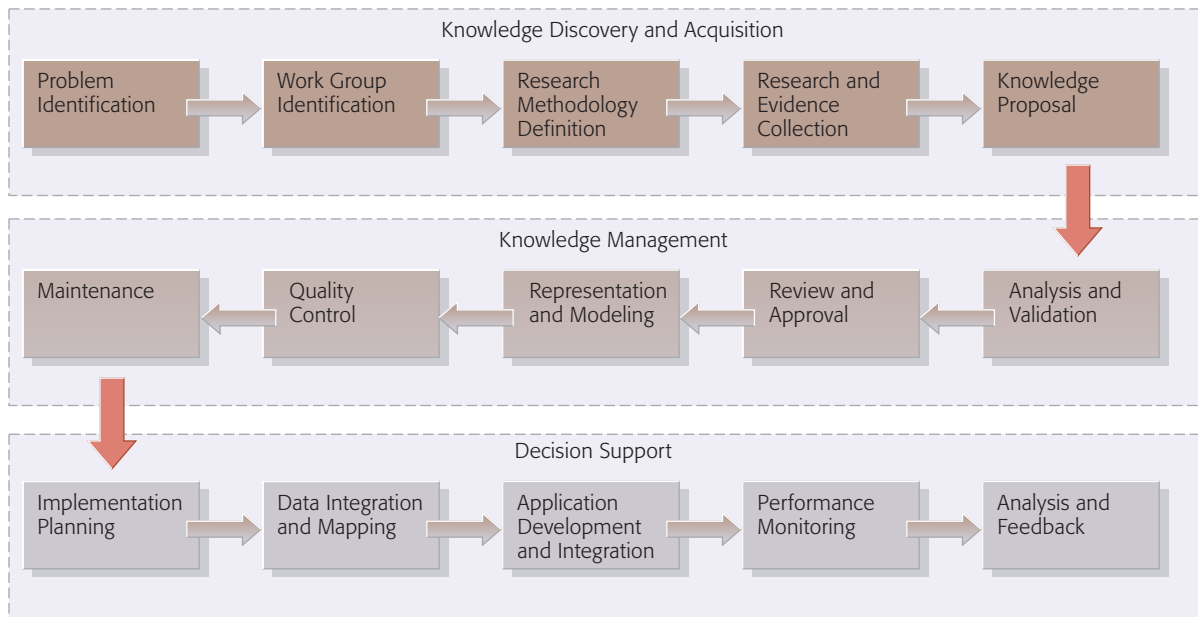


Figure 1
CDI process

cannot be well-managed without consideration of the others. These subprocesses are described in detail in the following subsections in the context of CDI and work preceding and related to CDI.

Knowledge discovery and acquisition

Knowledge discovery and acquisition is performed by many organizations, with or without an automated system. However, the goal of most such projects is to discover and acquire knowledge only, not to support an application. There is a gap between the knowledge acquired and how it will be used later. In CDI, we allow researchers to import knowledge into a system that can be used directly to support applications.

Clinical knowledge consists of statements, descriptions, policies, and guidelines, all of which may be used to provide decision support. The process of clinical knowledge discovery is carried out by a team of researchers and medical experts. During the knowledge acquisition or evidence-based research stage, the relevant knowledge that defines a clinical practice, rule, or policy is assembled from literature review, clinical data analysis and data mining, analytical modeling, and expert consensus. Information is extracted from research papers, reports, evidence tables, flow charts, and guidelines that

include evidence contents, sources, and quality scores.

In the context of this paper, we consider three main sources for knowledge discovery: primary research through analysis of longitudinal clinical data, secondary research through search, review and analysis of existing publications, and information from commercial, government, or academic content providers.

The emergence and evolution of longitudinal EMRs and clinical data warehouses allows researchers access to large amounts of deidentified longitudinal clinical data for performing primary research. Mining this clinical data provides additional insights through advanced analytics such as association analysis, clustering analysis, classification, and predictive modeling. These newly discovered insights can be used as evidence for supporting new organizational policies and guidelines, and new or existing clinical applications. The knowledge might be in the form of description summaries, diagrams, charts, or models. The evidence might be in the form of raw data or original analytical outputs.

Published peer-reviewed research papers and clinical trial reports supply significant supporting

evidence for secondary research, which can be used to build the clinical knowledge base.

Biomedical literature publications have been growing at exponential rates. The MEDLINE** database, provided by the United States National Library of Medicine (NLM), contains nearly 11 million abstracts from over 7,300 different publications from 1965 to the present and adds 7,000 to 8,000 abstracts per week.⁸ Researchers usually follow an established search methodology when collecting evidence for a search project. The knowledge is in the form of a summary report, a diagram, or a flow chart. In this case, the evidence is the collection of citations or original publications.

There are many government agencies, organizations, and businesses that focus on assembling clinical evidence and knowledge from diversified, reputable resources to provide clinical guidance for end users. These types of content may or may not need additional local validation or modification. Almost invariably, knowledge provided through this channel is supported by validated evidence.

One of the most notable enterprise knowledge acquisition projects is Quality Enhancement Research Initiative (QUERI) at the Veterans Health Administration (VHA).^{9,10} QUERI is an innovative integration of health-services research, policy, and clinical-care delivery designed to improve the quality, outcomes, and efficiency of VHA health care. The six-step QUERI process defines the policy and processes for identification and implementation of evidence-based practices in routine care settings. Through these rigorous processes and the overall effectiveness of the program, the VHA has established a national model for health-care quality improvement.

Many existing knowledge acquisition systems were developed based on the requirements of a specific knowledge-management or decision-support project. There is a consequent lack of a standard data model, standard import utilities for capturing knowledge and evidence, and a centralized repository for captured contents. This in turn creates problems for representing knowledge in a standard executable format and using the knowledge for decision support. Therefore, it is critical to create standards for knowledge-capture data models and tools. The knowledge acquisition process requires

tight linkage with the overall knowledge-management process.

Knowledge management

The benefits of clinical knowledge management have become apparent in recent years as more health-care organizations adopt evidence-based practice and develop quality improvement programs. An ineffective knowledge management system results in costly programs, duplicate efforts, and loss of valuable knowledge within the organization, all of which contribute to poor quality of care and increased costs. Without centralized and enterprise-level knowledge management it is difficult to collect knowledge from multiple sources and to search, manage, consolidate, and make this consensual knowledge available at the point of care. This also results in unnecessary complications in developing decision support applications.

In knowledge management, the key is to provide a centralized master repository that can be used to support many applications. Organizations have just started to implement knowledge management systems, but knowledge management systems are being used as a content repository only. People use these systems to store and later to search for information, not for creating rules to support applications, as is the case for CDI systems.

Efforts to bring clinical knowledge to the point of care were recently reviewed by Sittig.¹¹ Whether they are library-type applications, real-time decision support systems, or hybrids of the two, most of these systems focus on the contents and user interface of the application, not on building the IT infrastructure.

Hybrid systems, wherein the same system provides both a content repository and decision support, are still projects at the research prototype stage. One such project was undertaken at Partners HealthCare System.¹² This program has been under development for more than a decade and is still not complete. The system has been very effective in reducing medical errors, enabling guideline adherence, and reducing costs. However, it also demonstrates the challenges organizations face in developing complex systems. A particular challenge is the effort to allow organizations to embed their own knowledge into systems. This calls for the use of a standards-based technology platform to provide

the system infrastructure that supports the goal of implementing evidence-based medicine.

Knowledge management methodologies and technologies have been well-established in many industries and are available for adoption by the health-care industry. Recent studies of Banner Health's experiences in developing clinical knowledge management emphasized the importance and impact of business process changes and the cultural aspects of knowledge management.^{13,14} Banner Health has developed an integrated organization-wide effort called Care Management to simultaneously address quality, safety, cost, and performance issues within a 19-hospital health-care system. Lessons learned from the two-year-old program identified the challenges of knowledge management pertaining to process and policy.

The technical considerations for the success of a clinical knowledge management system are its ability to support the business goals, policies, and processes of the organization to ensure systematic translation of findings and knowledge-engineering work products into action. The knowledge management system needs to support centralized knowledge and evidence repositories with evidence linked to knowledge. The evidence needs to be analyzed, disseminated, summarized, and modified to create locally applicable rules, and these rules need to be converted into executable formats so that the knowledge can be applied at the point of care.¹⁵ The system needs to support a rigorous analysis, review, and validation process. Effective collaboration technologies such as online discussion forums or expert locators are necessary to encourage all stakeholders to participate in knowledge sharing. A comprehensive document life-cycle management function must be included in the knowledge management system to maintain records of change history and govern knowledge expiration and update.

Decision support

Clinical knowledge is discovered, acquired, and managed for the purpose of providing decision support. A clinical application takes a clinical situation as input and produces inference output that can be used to provide assistance in a clinical function.¹⁶ There are many types of clinical applications or decision support systems.¹⁷⁻²⁰ In this paper, we discuss only knowledge-based clinical

applications, which are systems with a knowledge base and an inference mechanism operating on a patient database.¹⁶

CDI may be compared with existing clinical decision support systems (CDSSes). In traditional clinical decision support, people tend to develop a new application based on their needs, with the rules for decision support embedded in the application. These rules often apply only to a single application. There is no systematic way to optimize the application development process and best use existing resources. In CDI, we use the same rules to support many applications. The focus is on application integration and simplifying application development.

Our focus is on large-scale, component-based systems supporting multiple functions including the following applications:

- *Computerized Physician Order Entry (CPOE)* systems are applications that enable physicians to enter orders for medications, laboratory tests, or radiology tests.
- *Clinical Reminders and Alerts* systems are applications that monitor and identify critical clinical conditions or clinical events and send notifications through wired and wireless devices.
- *Adverse Drug Effects (ADE)* is an application that reviews a patient's medication records and generates alerts if adverse drug-drug or drug-food interactions are found.
- *Report generation* systems create knowledge and both evidenced-based and personalized patient reports on laboratory test results, pathology test results, and so forth.
- *Clinical pathway* systems are patient-focused tools which describe the time frame and sequencing of routine, predictable, multidisciplinary interventions and expected patient outcomes, based on a patient's demographic profile and clinical conditions.
- *Disease management* systems coordinate health-care interventions and communications for populations with conditions in which patient self-care efforts are significant. Disease management systems emphasize prevention of exacerbations and complications by using evidence-based practice guidelines. These systems evaluate clinical, human, and economic outcomes on an ongoing basis with the goal of improving overall health.

Among the technical challenges of clinical decision-support application development are issues related to application integration and interoperability. These issues can be addressed with business process management and optimization. Integrating multiple applications can be difficult for clinical decision-support systems because most clinical applications are developed to solve a specific problem within the clinical information system, taking input from data within the same system, embedding rules and knowledge in the application, and using terminology specific to the system. Embedding knowledge into applications (known as “hard coding”) makes it difficult to change and update the knowledge without breaking the application and difficult to share the same knowledge with other applications.

Because applications are tightly linked to a particular EMR format, very often, one EMR system supports only applications within that system, and it is difficult to create an application using data from multiple CISEs or EMR systems. This causes serious interoperability problems as the need arises to provide an overall clinical assessment of a patient. This also limits the extensibility of legacy systems and creates system integration issues. Huge integration efforts are required to expand legacy application access to EMR data from other systems or to implement the latest clinical discoveries.

Applications are developed using terminology specific to a particular clinical information system. This makes application integration very difficult, as many point-to-point mappings have to be made between the terminologies of the applications. To help develop flexible, expandable, and interoperable applications, it is important to separate input data, knowledge logic, and terminology from the application. Data, logic, and terminology can be made available as services to any application.

Requirements for CDI systems

Technical and business-process issues call for an improved CDI system that addresses the issues mentioned in the previous section at an enterprise level. A successful enterprise CDI system needs to meet the requirements outlined in this section.

The CDI system should support knowledge acquisition, knowledge management, knowledge representation, and decision support subprocesses under a single framework. This helps improve quality of

care and reduce cost through better integration of clinical analytics, knowledge management, business process management, and performance monitoring. The system should provide knowledge-acquisition and knowledge-management tools suitable for the medical knowledge, IT skills, and organizational responsibilities of target users. The tools should be compatible with standard analytic processes and applications.

In addition, the system should provide a centralized knowledge and evidence repository and support enterprise search management, workflow management, collaboration, and document life-cycle management. Flexible development and integration should be enabled, improving the interoperability of decision support applications. Input data, medical knowledge and logic, and terminology should be separated from the application. This requirement necessitates a rule engine, a virtual data warehouse for patient data, a standard enterprise terminology-management system, and a standard knowledge-representation format. In the following section, we describe a CDI solution which satisfies these requirements.

A PROPOSED CDI SOLUTION

In this section, we present the functional description and reference architecture for a proposed CDI solution that we have designed and implemented.

Functional description

A block diagram of the CDI solution that we have developed is shown in *Figure 2*. The CDI system provides a framework for managing the entire knowledge discovery, knowledge management, and decision support processes and integrates with existing clinical information systems within an SOA framework. The system supports clinical transformation by providing technology enablement for improved clinical knowledge management and clinical application development processes.

The functions of the CDI system support a well-defined clinical management process. The key components of the process and functionalities are enterprise terminology management, evidence collection management, knowledge representation and management, and knowledge execution. Details of these components are discussed in this section. *Table 1* lists the roles and responsibilities of key users of the system.

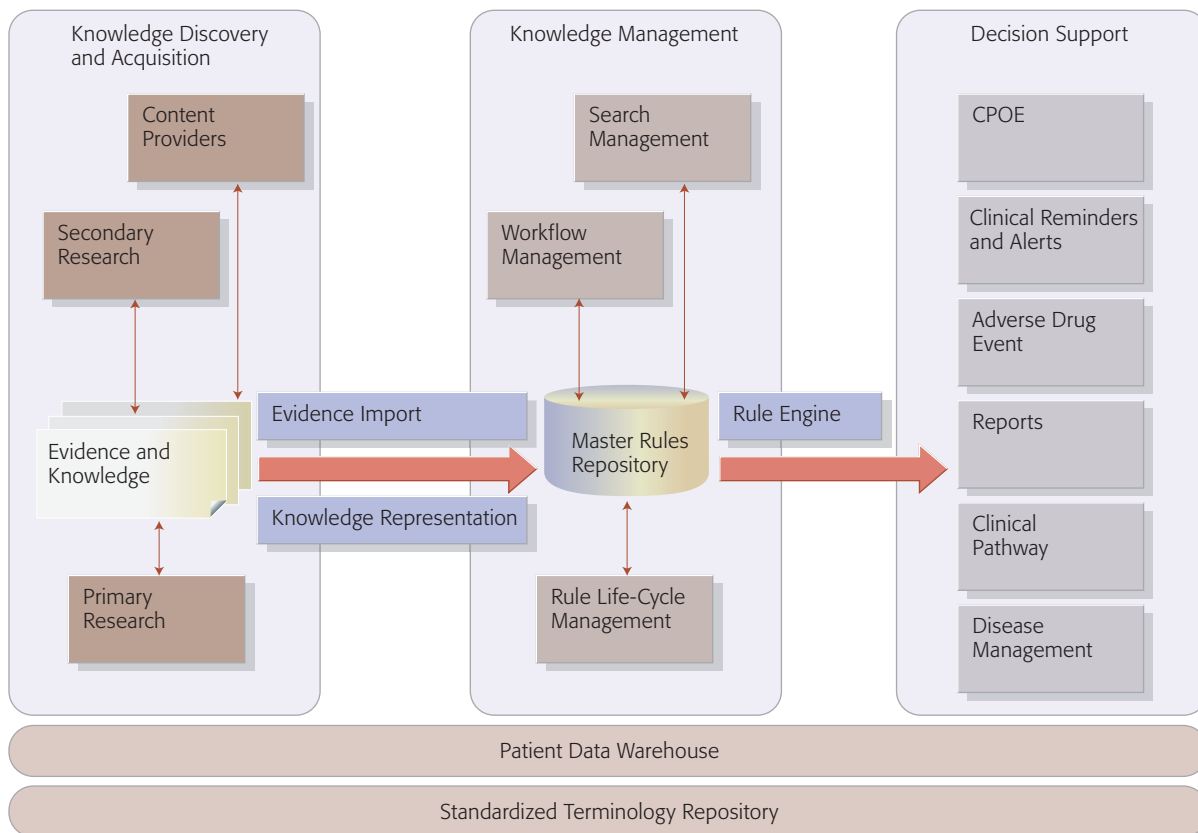


Figure 2
CDI solution

Table 1 Roles and responsibilities in the CDI process

Name of Role	Description and Responsibilities
Chief Rule Officer	Person or group of persons who are responsible for managing the entire CDI business process. Responsibilities include initiating new evidence collection and knowledge discovery projects for managing knowledge and for providing knowledge for applications. This person or group may also play the roles of chief medical officer, chief quality officer, or knowledge-management project manager.
Medical Researcher	Physician or medical research-team member who conducts research, gathers evidence, and creates, edits, and updates the knowledge statement.
Knowledge Engineer and Rule Quality Control Engineer	A medical informatician (a practitioner of informatics) who has medical knowledge and understands medical logic and who is responsible for creating logic and executable rules from the knowledge statement.
Evidence/Rule Reviewers	Internal and external domain experts assigned to review and validate the rules and supporting evidence. These are users with medical knowledge and with understanding of rule language who are responsible for reviewing and validating that text and boolean rules entered by the knowledge engineer are consistent and correct.
Medical Review Board	A formal committee responsible for overall quality of knowledge and its application in decision support.
Application Developers	Responsible for developing or modifying clinical applications using rules.

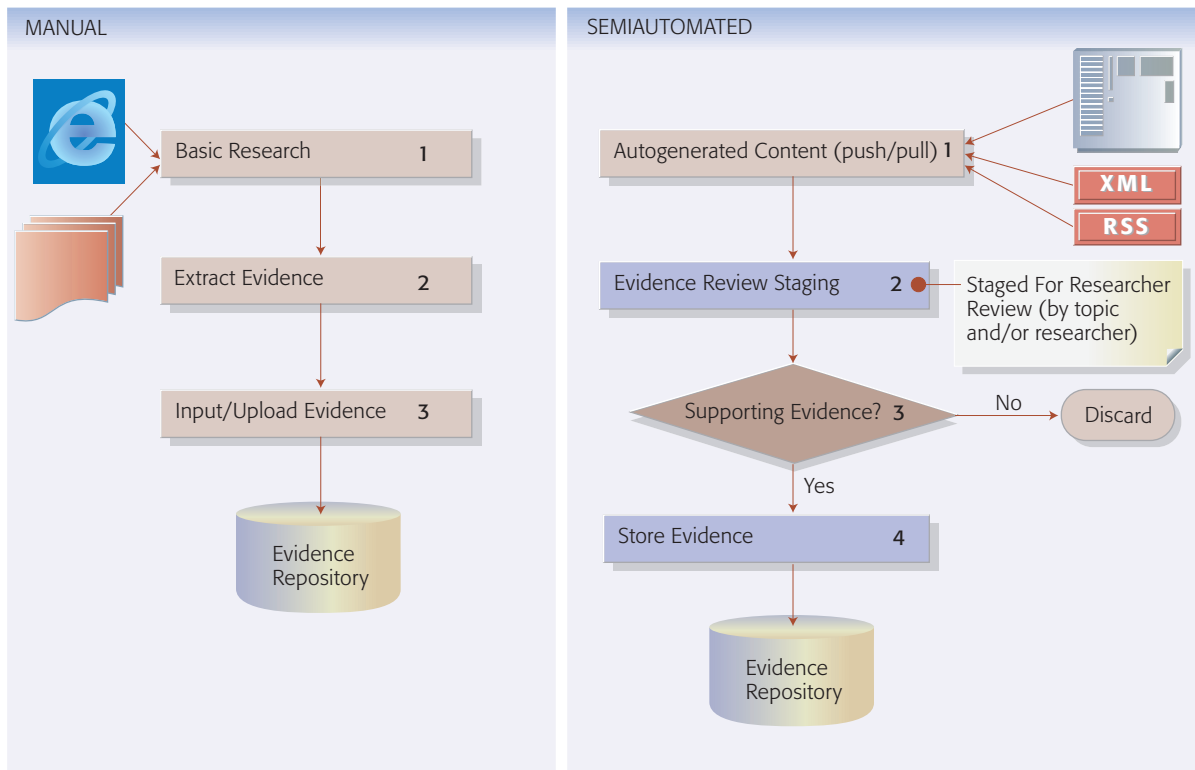


Figure 3
CDI evidence collection process

Enterprise terminology management

Each organization needs to create and maintain a standardized terminology repository, preferably based on industry standards. The CDI system supports a terminology data model containing definitions and structures for an enterprise-level standardized vocabulary. A graphical user interface (GUI) is provided to create, edit, and update system terminology.

Changes made to the system terminology dictionary must be controlled according to a defined business process, with only authorized personnel enabled to make changes. All changes must be reviewed and approved before taking effect. The terminology-update workflow-management function enforces these rules. A mapping service (the “terminology mapping process”) is provided to map nonstandard terminology from existing applications to standard terminology in a bidirectional process to support existing and new standard-compliant applications.

Evidence collection management

The evidence collection process is shown in *Figure 3*. A knowledge discovery project may be

initiated by researchers as part of standard research efforts. The researchers may submit the research results to the clinical knowledge base to be validated and approved for implementation in an application. More often, the project might be initiated by a “chief rule officer” or “chief quality officer” who oversees the collection of evidence and the support for a new rule or application.

In either case, the researchers are responsible for conducting primary and secondary research by collecting evidence and formulating a knowledge statement to be later converted into an executable format. The *evidence data model* is used to store evidence object data and metadata in the system, linked to the knowledge they support. The data model allows multiple items of evidence from multiple sources to support one knowledge set and a single item of evidence to support multiple knowledge sets.

There are two ways to enter evidence into the CDI system, as shown in *Figure 3*. In the manual import process, researchers perform these steps: primary or

secondary basic research (step 1), extracting evidence (step 2), and using the CDI evidence-importing utility to load the data into the evidence repository (step 3). In the semiautomatic import process, evidence is generated from other analytics applications (step 1) and pushed or pulled into a staging area for review (step 2). The reviewers examine the evidence and approve or reject the evidence (step 3), and approved evidence is stored in the evidence repository (step 4).

Knowledge representation and management

Clinical knowledge is represented in descriptive formats and in executable formats. In the CDI system, the *knowledge representation model* contains three formats: the knowledge statement, clinical logic, and the executable rule.

In this paper, CDI *knowledge statements* are defined as clinical knowledge in any format, such as text statements, decision tables, and flow charts that contain critical medical information. Knowledge statements are often created by a team of medical experts with deep domain knowledge and are intended to be interpreted by medical practitioners. The knowledge statement represents the consensus of the team describing what problem is to be solved, how it is involved in the clinical setting, who might be affected, what the outcome is, and so forth. Clinical knowledge statements are sometimes summaries and extracts of comprehensive clinical guidelines or summary reports.

Creating a clinical knowledge statement is the first step of disassembling medical knowledge into a computer-interpretable format. These statements might be in the format of a set of modules consisting of simple “if-then” rules or equivalent rule statements. For example, a drug usage guideline for patients undergoing percutaneous coronary intervention (PCI) might be stated as: if a patient is to undergo PCI, then a thienopyridine (e.g., clopidogrel, ticlopidine) with a salicylate (e.g., aspirin) should be given before and after the procedure, unless contraindicated.²¹ The knowledge statement might also be in the format of decision trees, order sets, and so forth.²²

Clinical logic is translated from the knowledge statement into a format compliant with a data model by knowledge engineers with broad medical knowledge and understanding of knowledge-mod-

eling technology. Great progress has been made in knowledge representation data models in the past 10 years, but the field is still challenged by the lack of widely adopted standards, lack of interoperability, and inaccuracy in knowledge representation.²³

The Arden Syntax is an HL7** (Health Level Seven**) standard for representing computable clinical knowledge.^{24,25} In the Arden Syntax, each decision rule is called a medical logic model (MLM). An MLM is a hybrid between a production rule (such as an if-then rule) and a procedural formalism. Each MLM contains sufficient logic to make a single clinical decision. Sequencing tasks can be modeled by chaining a sequence of MLMs as instructions, including queries, calculations, logic statements, and write statements. MLMs have been successfully implemented to generate clinical alerts and reminders, interpretations, diagnoses, screening for clinical research studies, quality assurance functions, and administrative support.

Another category of model, the task network model (TNM) has been developed for representing complex, multistep clinical guidelines and for more effectively describing temporal and other relationships between component tasks. These include the Asbru, EON, GLIF, GUIDE, PRODIGY, and PROforma models.²⁶ TNMs can explicitly model alternative pathways or sequences of tasks (i.e., control flow) and can provide tools for visual representation of plans and the organization of tasks within them. Common features from these models are evolving into a new HL7 standard called GELLO, a language for representing clinical guidelines. The group developing the object-oriented GELLO query and expression language aims to create a standard data model that is compatible with the HL7 Reference Information Model^{27,28} (RIM).

Executable rules are rules in a machine-readable format that can be interpreted by an execution engine. Executable rules are created by programmers who understand the language used by an execution engine, by knowledge engineers using a GUI that generates code for the engine, or by machine-generated code.

The CDI knowledge representation model supports all three formats of knowledge representation: knowledge statements, clinical logic, and executable rules. The data model also supports the quality control workflow which ensures that executable

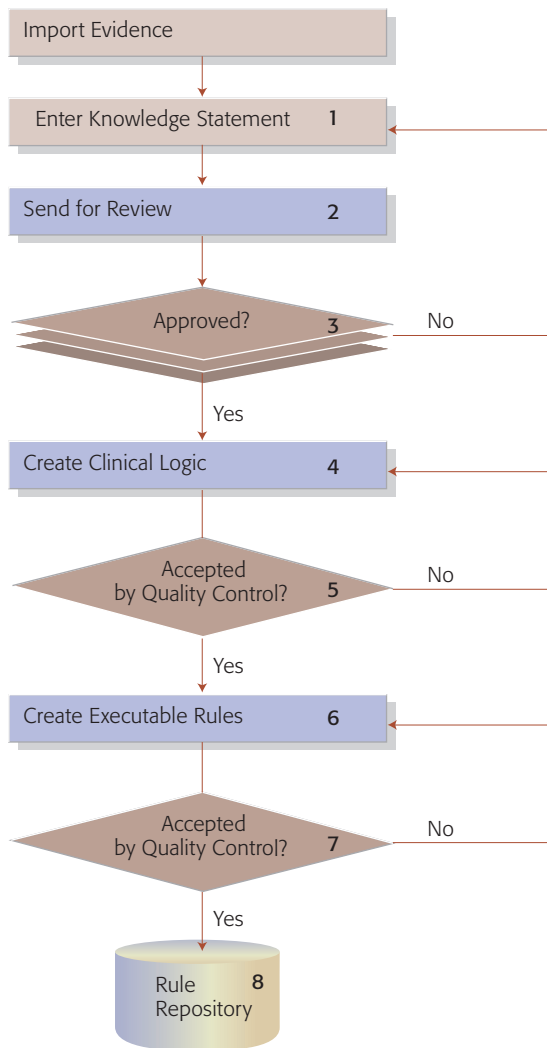


Figure 4
Knowledge management workflow

rules are consistent with clinical logic and knowledge statements. CDI allows free-form knowledge statements. CDI supports the Arden Syntax for describing medical logic and ABLE (Agent Building and Learning Environment) rule logic for the executable rules.

Vocabularies used in clinical logic and executable rules need to comply with the standard, user-defined terminology in the system vocabulary. Only terminology existing in the system vocabulary is allowed. This is controlled and managed from the system vocabulary control features of the CDI user interface.

The knowledge management workflow is shown in **Figure 4**. Because the system consists of a set of

preconfigured building blocks, the workflow sequence can be rebuilt easily, based on changing business processes. A knowledge statement is entered into the system during the knowledge discovery process. Typically, knowledge statements created by the knowledge discovery project team are sent to a team of internal or external experts for peer review. The peer review often focuses on validating the scientific credibility of the evidence, confirming that the evidence is appropriate to support the stated conclusions, and providing their own level of confidence in the review.

After several iterations, the knowledge statement is updated and submitted to a review board for approval. This is often a formal business process, requiring participation from affected departments or personnel. A knowledge statement may be rejected for many reasons, including questions on the value of the scientific evidence, the statement's applicability to organizational environments, duplication or conflict with existing organization policy, and so forth. A rejected knowledge statement might be modified, processed, and approved in another workflow cycle.

An approved knowledge statement is then translated into clinical logic by a knowledge engineer. A quality control process is in place to ensure that no medical context is lost or modified during this translation. The clinical logic is then translated into executable rules through a manual, semiautomatic, or automatic process. Again, a quality control process ensures that the contents are consistent in all three formats. A life-cycle management system ensures that a rule cannot be activated for production until it has gone through the complete quality control process. This system also manages versioning of rules. It ensures that any changes made to the rules are recorded and that rules have an expiration date so that all rules can be periodically reviewed. Rules need to be properly annotated and indexed so that they can be found by search systems and used for decision-support application development as well as business management.

Knowledge execution

The knowledge execution process involves three components—a rule engine, its inputs, and its outputs. Inputs for the rule engine are composed of rules and patient data. Rules are stored in the executable rule format created during the knowledge management process. Patient data is aggre-

gated from multiple systems and passed on to the rule engine. The output of the knowledge execution process is provided to clinical applications as rule services. These rule services can be in various forms and formats.

The CDI *rule engine* is an inference engine capable of interpreting simple if-then rules, as well as complex forward-chaining and backward-chaining inference rules. Data in legacy systems is mapped or converted to standard terminology before being passed to the rule engine.

Before being passed to the rule engine, patient data is aggregated. It is likely that for every patient in the system, data is stored in multiple CISEs and EMRs. Although it is not necessary to consolidate all patient data into a single repository, it is desirable to create a *virtual patient data warehouse* to establish a single view of all information related to a patient. It also helps facilitate a mapping of the data to the master vocabulary repository. Having this virtual patient data profile means that decisions about the patient can be made based on more complete knowledge of the patient over a period of time, not just a subset of patient data stored in a single system. Evaluating the whole patient means accurate, timely diagnosis and treatment and better outcomes.

Reference architecture

Before the details of the CDI system and the subsystems that compose it are described in this section, key technologies and standards used in CDI are reviewed. The CDI implementation platform is reviewed at the end of this section.

Key technologies and standards utilized in the CDI solution

The technologies and standards reviewed here include ABLE, the Arden Syntax MLM, service-oriented architecture (SOA), and Web Services.

Agent Building and Learning Environment. Although the clinical data intelligence domain has matured, it is still evolving, and its standards are still under development. Standardization among rule engine languages (i.e., languages that are actually executed by a rule engine) has not occurred and probably will not happen in the foreseeable future because of the specific functionality and algorithms of the respective rule engines.

ABLE consists of a software architecture and framework, component library, development tooling, and an agent platform for constructing autonomous intelligent agents and multiagent systems.²⁹ At the core of ABLE is the ABLE rule engine. The ABLE toolkit and rule engine were chosen for our CDI solution for several reasons, including the ability to discretely embed ABLE as a subsystem of the overall CDI system, the breadth of algorithms supplied, and its extensibility and incorporation of new algorithms. Additionally, ABLE runs on a J2EE** (Java** 2 Enterprise Edition) platform and, while utilizing standard J2EE facilities such as Java Message Service (JMS), Java Management Extensions (JMX), and Web Services, enables seamless integration with the rest of the enterprise.

The Arden Syntax MLM. The Arden Syntax for Medical Logic Systems,^{17,18} first introduced in 1989, has been one of the leading representation models for clinical knowledge. Our CDI solution uses the Arden Syntax as a base schema for clinical logic representation and supplements this base schema with additional CDI metadata. This design allows the CDI system to import, export, and fully utilize Arden Syntax MLMs. The logic portion of the Arden Syntax MLM contains pseudo-rule execution language. Although not directly executable by the CDI, this logic is translated by the CDI into the ABLE rule language and executed by the ABLE rule engine.

SOA and Web Services. SOA is a paradigm that has recently emerged as the de facto standard for integrating enterprise data and applications. By decoupling functionality and technology, an SOA can provide the foundation for an agile and stable infrastructure to support ever-changing business demands and services.

The business services provided and consumed in the enterprise using an SOA are realized as Web Services. Web Services are a standard method for providing IT services to the enterprise and do not depend on any particular underlying platform, operating system, or programming language. The Web Service is formally described in a specification or contract describing the inputs and outputs of the service. The specification is defined in a WSDL (Web Services Description Language) document.

By adopting the SOA/Web Services approach, the CDI system can be integrated more easily with the

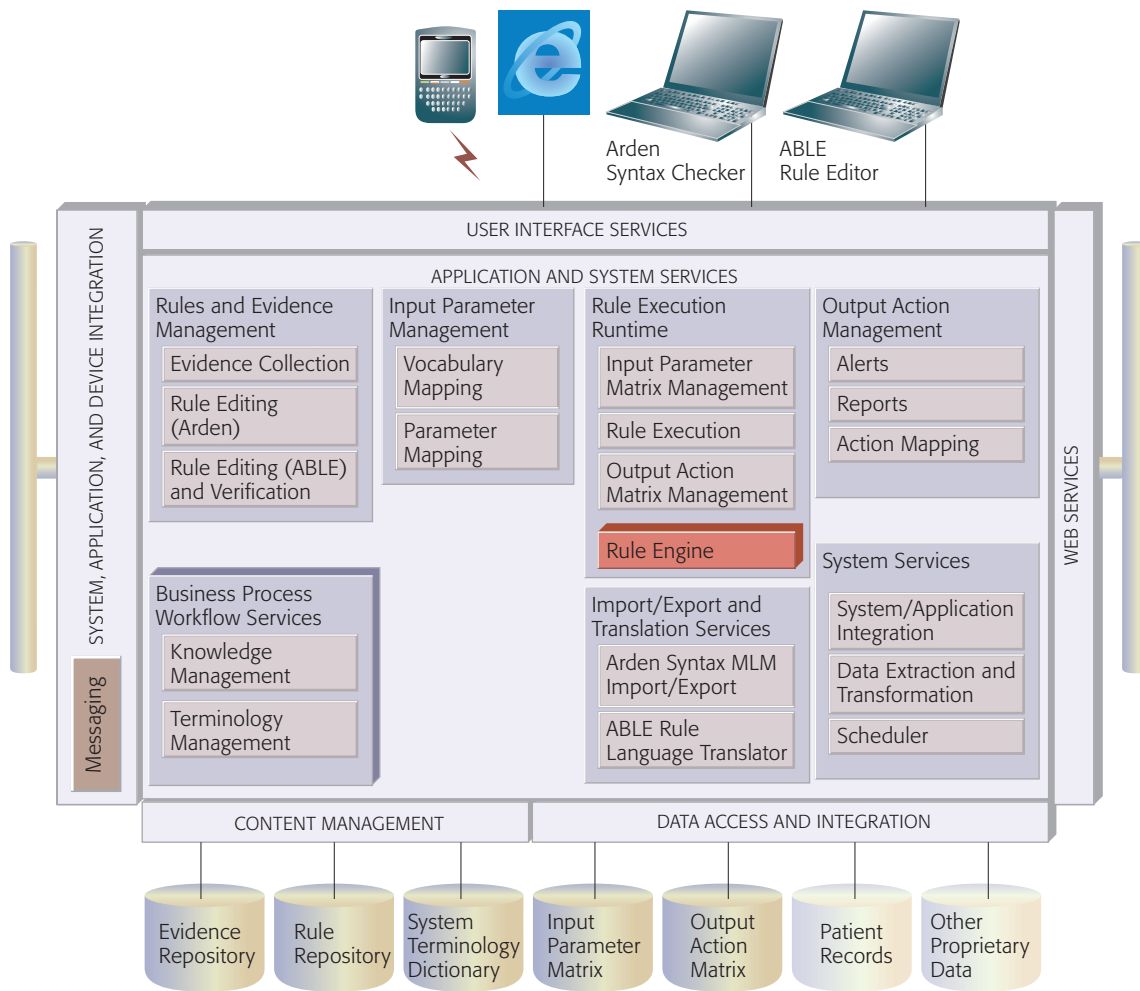


Figure 5
CDI architecture

rest of the enterprise. The CDI infrastructure can be plugged directly into an existing enterprise SOA or can provide the foundation for an SOA and enables standardization and improved integration with systems external to the CDI.

CDI system and component subsystems

The CDI architecture is comprised of six subsystems, as shown in *Figure 5*. These are: User Interface Services; Application and System Services; Web Services; System, Application, and Device Integration; Content Management; and Data Access and Integration. The abstraction of services into discrete subsystems assigns definitive responsibilities to the respective subsystems and minimizes interdependencies between areas. This allows for

extension and growth while maintaining acute localization of impact on the system. The business logic of the system is encapsulated entirely in the Application and System Services subsystem. For clarity of design, this subsystem has been further divided into service areas, each representing a functional area of the CDI system. Each subsystem typically uses services provided in other subsystems to achieve an overall task. The CDI subsystems and functional service areas are explored in the following sections.

User Interface Services subsystem. User Interface Services provide presentation facilities and support user interaction with the CDI system. The CDI application is a Web-based application with most of

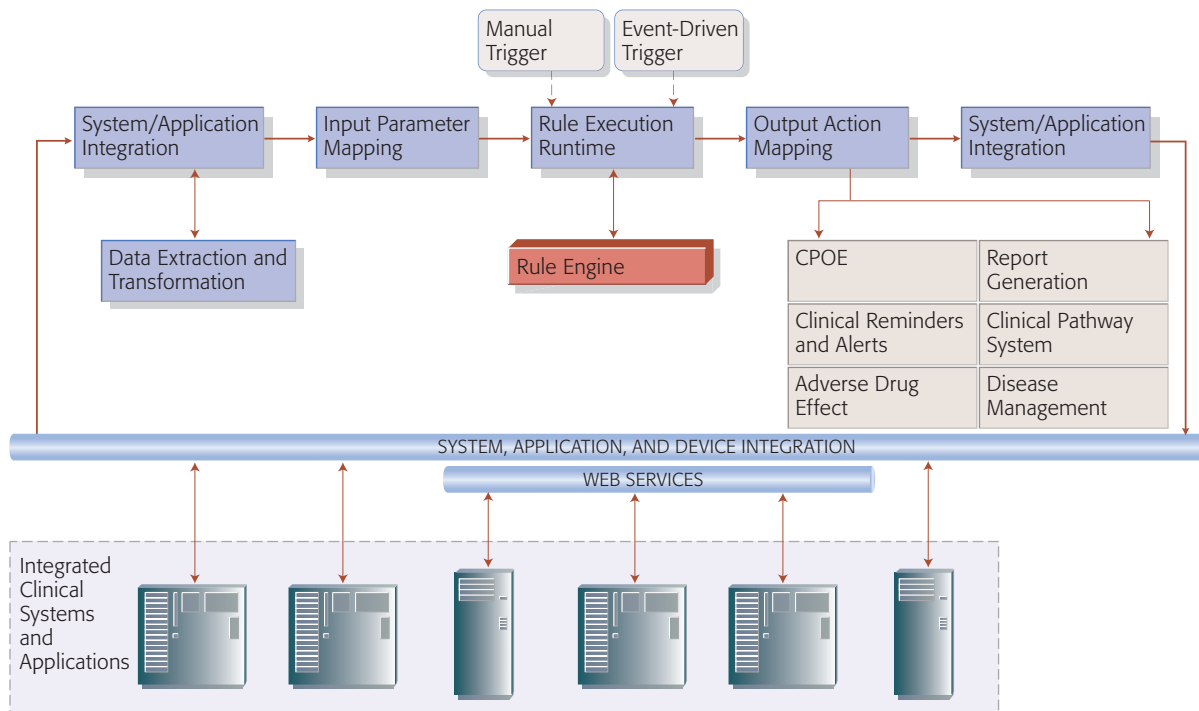


Figure 6
CDI rule execution process

the user interaction performed by using a standard Web browser. Additionally, the services provided by this subsystem support other types of interaction with wireless handheld devices, allowing freedom of movement away from a desktop, as in hospital scenarios. This subsystem also provides services and data for existing third-party authoring applications, such as the Arden Syntax Checker, the ABLE rule editor, or the IBM Data Discovery and Query Builder (DDQB) rule editor.³⁰⁻³²

Application and System Services subsystem. The Application and System Services subsystem encapsulates the logic of the CDI application and is realized in the form of Enterprise JavaBeans** (EJB) and Java classes. This subsystem is the largest of the subsystems and encapsulates all the higher-level functionality of the system. It uses the lower-level functionality enabled by other subsystems as support for enabling the higher levels of functionality. This subsystem is categorized further into seven key service areas and facilities, which are described in the following subsections.

The Rules and Evidence Management facility provides functionality to enable the collection of

evidence, the creation of rules, and the mapping of rules to supporting evidence (which validates the knowledge discovered or created). Additionally, services are provided to enable the creation, editing, and verification of the executable form of the rule engine scripts.

The Input Parameter Management facility provides the services for mapping input data to the rule runtime parameters. Details of the CDI rule execution process are shown in *Figure 6*. Input parameter mapping defines the relationship between a rule's input parameters and the source data for each respective parameter. Part of this task involves the vocabulary mapping that translates the terminology of the source data into the CDI terminology required by the rule. A browser-based interface is provided for the user to create and manage the input mappings. By eliminating hard-coded input mappings through this additional mapping layer, the CDI system is better able to concurrently support a multitude of external systems and input mapping scenarios. The Input Parameter Management facility delegates the responsibility of interfacing with the external systems to the System, Application, and Device Integration subsystem.

The Rule Execution Runtime facility provides low-level services that are transparent to the user. Its sole purpose is to execute rules in the rule engine based on the input parameters and provide the rule outcome to the Output Action Management facility (see Figure 6). This facility is responsible for executing the appropriate action based on the rule outcome. The CDI system presently operates with the ABLE rule engine executing rule logic translated from the Arden Syntax, from the DDQB rule editor, or from manually edited ABLE rule scripts.

Similar to input mapping, the Output Action Management facility provides services which allow for the abstraction of rule outcome from the action triggered by the rules (see Figure 6).

The Business Process Workflow Services facility is responsible for business-process and document workflows. Document workflows can be executed in isolation or may be contained in larger business process workflows. The CDI system has several predefined workflows for knowledge management and terminology management and can be extended with additional workflows to support enterprise demands.

The provided *rule management workflow* controls the review, validation, and acceptance of a rule into an approved, executable rule by the CDI system. The full contents of the rules, including supporting evidence, become the documented and validated knowledge of an organization.

The Import/Export and Translation Services facility is an extensible set of low-level services that provides capabilities to import and export data and translate between different formats of clinical logic and executable rules. The import/export services allow users to import and export evidence, knowledge statements, clinical logic (e.g., Arden MLMs) or executable rules (e.g., ABLE scripts), into and out of the CDI system. The translation services enable the automated translation of clinical logic into engine-specific rule execution scripts.

The provided *system terminology management workflow* controls the review and approval process for system terminology. The resulting approved system terminology is presented as a data service through the Data Access and Integration subsystem and used by the Input Parameter Management, Rule Execution Runtime, and Output Action Management

facilities. The system terminology workflow also controls the vocabulary used by the Data Extraction and Transformation service.

The System Services facility provides functionality that manages the logic of the System/Application Integration module and the Data Extraction and Transformation module. This logic specifies how the CDI system integrates, extracts, and transforms data while relying on services of the System, Application, and Device Integration subsystem and the Web Services subsystem to support low-level physical-system integration.

Also contained in the System Services facility is a scheduler to provide scheduled services (i.e., execution of rules, output action triggering, alerts and notifications, data extraction, etc.) to other facilities.

Web Services subsystem. The Web Services subsystem is responsible for composing and making available services to the enterprise as well as managing the interface for consumption of Web Services provided by other systems. This subsystem is the primary mechanism for integration of the CDI system with other systems in the enterprise.

The following is a typical example of a flow integrating the CDI system with enterprise systems by using the Web Services subsystem.

1. A CPOE system calls the rule execution Web service exposed by the CDI system and passes either patient metric data or a patient identifier (to obtain the patient metric data).
2. The CDI Web service responds with an urgent alert to the CPOE system, notifying the physician of the risk or providing recommendations specific to the patient.
3. The rule executed might additionally send a message to an external alert logging facility or e-mail a third party (this is optional, based on output action mappings).

System, Application, and Device Integration subsystem. The System, Application, and Device Integration subsystem is responsible for the integration of the individual facilities and components that, when used together, comprise the CDI solution.

A key component of this subsystem is store-and-forward messaging. Although Web Services provide

integration at the enterprise level, certain systems, applications, and devices do not expose their interfaces as Web Services. Another issue is performance. Web Services carry a substantial amount of overhead and are mainly targeted at exposing enterprise business services. Certain devices and applications in the enterprise require higher performance and a finer granularity of communication to viably integrate with the CDI system. The System, Application, and Device Integration subsystem wraps and masks the complexity of integrating these systems and devices and exposes them internally for use by the CDI system in a standard way. The logic for how they are wrapped and internally exposed is controlled in the System Services facility of the Application and System Services subsystem.

Content Management subsystem. The Content Management subsystem is required for storing, indexing, searching, and retrieving all forms of multimedia used by the CDI system. An example of this is the storage of evidence, which may be in various forms including textual, graphical, and audio data.

Document workflows provide services for routing and life-cycle management of documents used within the CDI system. The low-level services provided by the Content Management subsystem are used by the Rules and Evidence Management facility and the Business Process Workflow Services facility to deliver the high-level functionality provided by the respective facilities.

Data Access and Integration subsystem. The Data Access and Integration subsystem provides centralized and uniform access to all relational and nonrelational data used by CDI. The data accessed through this subsystem includes management data stored locally, supporting the operation of the CDI system (e.g., the output action matrix), and also the integration of enterprise data (e.g., patient records). The integration performed by this subsystem is done by using a federated database, which masks the location, format, access protocol, and general heterogeneity of the underlying source data. Data may be stored locally and outside the CDI system on geographically dispersed systems. This enables the creation of a virtual data warehouse that coexists with and federates existing data warehouses in the enterprise. This federated database provides unified and uniform access to all data used by the CDI

system and can be used by other applications and systems in the enterprise as well.

Platform

The CDI solution has been implemented on a J2EE platform, specifically the IBM WebSphere* Application Server. This platform was chosen due to its provision not only of J2EE services but also extensions above and beyond the standard J2EE services. These extensions are typically required in an enterprise-class application. It also provides a suite of out-of-the-box integration facilities and connectors for many third-party applications. Additionally, the IBM WebSphere platform is tightly integrated with other products and services required to support the CDI system, such as Web services, messaging, data integration, content management, and document workflow. The messaging, data integration, and content management and document workflows are provided by IBM Websphere Process Server, IBM Websphere Information Integrator (II), and DB2* Content Manager respectively.

APPLICATIONS OF THE CDI SOLUTION

In this section, two CDI system-application use cases are discussed. The first use case describes a new clinical application based on the CDI infrastructure. The second use case describes how CDI can be used as the foundation for enterprise application integration on an SOA.

Personalized oncology-treatment recommendation system

An application has been developed to create a personalized oncology-treatment recommendation system. This solution will be used in an advanced diagnostics laboratory where immunohistochemistry, clinical genomics, and other esoteric tests are performed on tissue samples from cancer patients.

The knowledge discovery and acquisition subprocess applies advanced text-mining technology on biomedical literature to find associations between phenotype, genotype, and therapeutic treatments. These findings are captured in the knowledge base as knowledge statements and executable rules.

The knowledge management subprocess enables all knowledge workers, such as researchers, knowledge engineers, content reviewers, review committee members, and quality control engineers, to access the system through a single Web-based interface.

The knowledge workers collaboratively add, modify, review, and update the contents in the knowledge base based on their roles in the predefined workflow process.

The decision support subprocess generates personalized-treatment recommendation reports based on a patient's demographic, phenotypic, and genotypic profiles and rules in the knowledge base. The reports can be generated upon user request or upon a Web Services request from the laboratory systems, and can be retrieved, viewed, and edited through a secure Web interface. Any modifications to the report are tracked by the system, and a modification history is available. The reports become part of the patient's EMR. The solution supports HIPAA (Health Insurance Portability and Accountability Act) patient privacy regulations.

Data services provided by the system include an enterprise vocabulary management service and a patient data integration service. The vocabulary management service provides a vocabulary repository and mapping services for all applications in the enterprise. The data integration service aggregates patient information from CISes, which contain a patient's demographic and phenotypic information, and multiple laboratory information management systems (LIMSes). The integrated patient data creates a virtual patient data warehouse with data from different commercial systems and homegrown applications. The complete patient profile is used when mining the rule repository, and those rules applicable to the patient along with supporting evidence are included in the final report. The reports also include patient information obtained from multiple data sources.

This personalized oncology-treatment recommendation system enables the health-care organization to create and manage knowledge and apply that knowledge to practice evidence-based and personalized cancer treatment. This application demonstrates how to develop a new knowledge-management and decision-support system by using the CDI framework. The ease and flexibility of decision-support application development means that new scientific discoveries can be quickly applied to patient care. When a newly published clinical trial demonstrates the effectiveness of an experimental drug on cancer patients with a specific gene expression profile, the laboratory can add this

new trial report as evidence for a new rule, while developing a new microarray test for detecting this expression profile. Test data from LIMSes can be added easily to the virtual patient profile. Once the new rule is reviewed, validated, and put into production, reports can be generated for all existing and new patients to identify candidates for receiving this new treatment. This process significantly reduces the time for new scientific and clinical discoveries to be used in patient care, which has been reported as taking up to 17 years.³³

Enterprise clinical data and application integration

Clinical data and application integration is important, demanding, and challenging. Health-care providers have to manage one of the most complex information management systems used today. It is not unusual for providers to use information management systems from over a dozen vendors to manage clinical, administrative, and financial aspects of their operations. The applications they use include those related to patient information management (admission, discharge, and transfer orders, care plans, etc.), pharmacy management, imaging management, laboratory management, materials management, payer and accounting management, and more. Each of the systems often has its own proprietary data repository and API (application programming interface), which makes point-to-point integration difficult and costly. As more vendors move toward providing standard-compliant Web services, it is important to take advantage of this movement and develop an enterprise integration strategy based on SOA.

The following is a hypothetical use case of a CDI application. In this use case, CDI is used as the foundation for integrating applications related to cardiology care. Cardiology-related clinical knowledge, practice clinical alerts, and position statements from professional organizations are stored in the CDI knowledge repository. The knowledge base covers a broad range of information that might be used for decision support in the cardiology department, such as: What is the profile of patients at high risk for developing heart disease? What tests should be ordered for high risk patients? When should a patient be transferred to cardiologists based on screening test results? What tests should be selected from a cardiac catheterization order set? How should presurgery risk factors be calculated based

on test results prior to catheterization? What treatment recommendation should be made based on cardiac catheterization findings? The information in the knowledge base is collected, validated, reviewed, and approved through a predefined business process. Information is routinely reviewed and updated. Each rule in the knowledge base supports one or more applications, using data from multiple systems inside and outside the hospital.

Multiple clinical information systems are used in the cardiology department, including a specialty cardiovascular EMR system, electrocardiogram (ECG) monitoring devices and data repositories, and a cardiology imaging system. In addition, the department shares data with hospital-wide clinical information systems for a general EMR system, LIMSeS, pharmacy management systems, and radiology management systems. Although each of these clinical information systems has embedded decision support functions, the applications only access patient information within the same clinical system and provide decision support within the system. Additional application development is required to provide decision support based on data from more than one system. The CDI system provides the foundation for such system integration. A federated server integrates the disparate clinical information systems in use to provide a virtual patient data profile. The centralized rule repository, rule engine, and execution services enable rapid new application development by using data from other systems.

It is well-known that diabetic patients suffer from much higher re-stenosis rates than non-diabetics after PCI. Recent clinical trials have demonstrated improved re-stenosis rates and clinical outcomes for diabetic patients when sirolimus-eluting stents (SES) are used during PCI compared to bare metal stents.³⁴⁻³⁷ After systematic review of the clinical trial reports and internal review, a new rule is added to the knowledge base that requires SES instead of bare metal stents for diabetic patients during PCI. Because this is a new rule affecting only a small percentage of patients, the implementation committee recommends that an alert be added to automatically inform the cardiologist of this rule. Meanwhile, a new reporting parameter is added to the quality performance-monitoring application to monitor the effectiveness of this alert on physician behavior and the effectiveness of the procedure for diabetic patients.

To develop this new alert and monitoring system, related fields of patient data from hospital clinical information systems are collected and sent to the rule engine through patient data Web services. When diabetes is reported in the patient's medical history along with real-time data collected during presurgery testing, an alert is sent to the cardiologist during presurgery risk assessment requiring SES to be used for this procedure. Meanwhile, a request is sent to a quality reporting application to collect data on the extent to which SES is actually used by each cardiologist during the procedure and to monitor if the patient has developed re-stenosis. Both items (the alert and the request) are sent as the result of Web Services calls. The quality reporting application which collects data for the Joint Commission on Accreditation of Healthcare Organizations (JCAHO), Centers for Medicare and Medicaid Services (CMS), and other agencies by detecting any deviations from normal patterns is based on the same knowledge and rules used for developing the decision support applications. After a certain period of monitoring, if the hospital reports an improved re-stenosis rate, it could be rewarded with better payment for its services in this field by the health-care payers.

This hypothetical use case demonstrates the benefits of using the CDI system to integrate clinical applications within an SOA. Separating rules from business processes and business logic facilitates rapid development cycles for new applications and ensures that new clinical discoveries can be used within a short period of time without disrupting existing systems and business processes. Having multiple applications sharing the same knowledge and rules ensures consistency among applications and reduces development efforts. CDI simplifies the enterprise application integration process while providing up-to-date, evidence-based personalized medicine.

SUMMARY

We have presented the reference architecture, functional components, and applications of a CDI solution. Through two clinical scenarios, we demonstrated how the system can be used to create an integrated framework for knowledge discovery, management, and decision support.

The SOA-based CDI system presented here enables centralized knowledge collection, review, validation, and application; enables rapid monitoring and updating of clinical best practices; enables enter-

prise-wide quality management programs and standardized practice and ensures approved guidelines are followed in different application environments; expedites the process of turning scientific and clinical discoveries into clinical care; and helps organizations maintain compliance with evolving government regulations, professional guidelines, and pay-for-performance measurements.

Overall, the CDI infrastructure helps organizations achieve the goals set by the quality initiatives: applying evidence-based medicine to health-care delivery, improved utilization of information technology, aligning payment policy with quality improvements, and educating the workforce with standards and guidelines.

ACKNOWLEDGMENTS

We wish to thank Margit Chapman, Frank Wang, Michael Boroch, Gail Ovian, and the anonymous reviewers of this paper for carefully reviewing the manuscript and providing valuable inputs. We thank Dr. Robert Penny, Dr. Edward Suh of the Molecular Profiling Institute, and Dr. Lawrence Gimple of the University of Virginia Health System for sharing their knowledge of clinical genomics diagnostic testing and cardiovascular care. We also thank the many clients who shared their experiences and challenges in developing CDI solutions. We wish to acknowledge the following colleagues for their contribution and support to the design and development of the CDI solutions: Mike Koranda, Houtan Aghili, Anne Aldous, Edward Macko, Paul Mattson, Joe Bigus, Michael Boroch, Frank Wang, Ken King, Gail Ovian, Peter Lange, Saleem Hussain, Leonard Lee, Mike Breitbach, Mitch Arends, Satish Gambire, and Chetna Warada.

*Trademark, service mark, or registered trademark of International Business Machines Corporation in the United States, other countries, or both.

**Trademark, service mark, or registered trademark of The National Library of Medicine, Health Level Seven, Inc. or Sun Microsystems, Inc. in the United States, other countries, or both.

CITED REFERENCES

1. L. T. Kohn, J. M. Corrigan, and M. S. Donaldson, *To Err Is Human: Building a Safer Health System*, National Academy Press, Washington, D.C. (2000).
2. M. Edington, *Crossing the Quality Chasm: A New Health System for the 21st Century*, Committee on Quality of Health Care in America, Institute of Medicine, National Academy Press, Washington, D.C. (2001).

3. *The 2005 National Healthcare Quality Report*, U. S. Department of Health and Human Services, Agency for Healthcare Research and Quality (AHQR) (December 2005).
4. National Association for Healthcare Quality, <http://www.nahq.org>.
5. The Leapfrog Group, <http://www.leapfroggroup.org>.
6. Institute for Healthcare Improvement, <http://www.ihl.org/ihl>.
7. D. L. Sackett, W. M. C. Rosenberg, J. A. Gray, R. B. Haynes, and W. S. Richardson, "Evidence-Based Medicine: What It Is, and What It Isn't," *British Medical Journal* **312**, Issue 7023, 71–72 (1996).
8. United States National Library of Medicine, National Institutes of Health, <http://www.nlm.nih.gov/>.
9. D. M. Hynes, R. A. Perrin, S. Rappaport, J. M. Stevens, and J. G. Demakis, "Informatics Resources to Support Health Care Quality Improvement in the Veterans Health Administration," *Journal of the American Medical Informatics Association* **11**, No. 4, 344–350 (July/August 2004).
10. L. McQueen, B. S. Mittman, and J. G. Demakis, "Overview of the Veterans Health Administration (VHA) Quality Enhancement Research Initiative (QUERI)," *Journal of the American Medical Informatics Association* **11**, No. 5, 339–43 (September/October 2004).
11. D. F. Sittig, "An Overview of Efforts to Bring Clinical Knowledge to the Point of Care," in *Clinical Knowledge Management, Opportunities and Challenges*, R. K. Bali, Editor, IDEA Group Publishing, Hershey, Pennsylvania (2005).
12. T. H. Davenport and J. Glaser, "Just-in-Time Delivery Comes to Knowledge Management," *Harvard Business Review* **80**, No. 7, 107–111 (July 2002).
13. B. Kirkman-Liff, "The Structure, Processes, and Outcomes of Banner Health's Corporate-wide Strategy to Improve Health Care Quality," *Quality Management in Health Care* **13**, No. 4, 264–77 (2004).
14. T. Burdick, J. Hensing, B. Kirkman-Liff, P. Nenaber, H. Silverman, and M. Simington, "Managing Knowledge to Improve Health Care Quality in Banner Health," in *Creating Knowledge-Based Healthcare Organizations*, N. Wickramasinghe, J. N. D. Gupta, and S. K. Sharma, Editors, Idea Group Publishing, Hershey, PA (2005).
15. L. R. Waitman and R. A. Miller, "Pragmatics of Implementing Guidelines on the Front Lines," *Journal of the American Medical Informatics Association* **11**, No. 5, 436–438 (September/October 2004).
16. J. H. Bemmell and M. A. Musen, *Handbook of Medical Informatics*, Springer-Verlag, Heidelberg (1997).
17. M. E. Johnston, K. B. Langton, R. B. Haynes, and A. Mathieu, "Effects of Computer-Based Clinical Decision Support Systems on Clinician Performance and Patient Outcome: A Critical Appraisal of Research," *Annals of Internal Medicine* **120**, No. 2, 135–142 (1994).
18. R. N. Shiffman, Y. Liaw, C. A. Brandt, and G. J. Corb, "Computer-Based Guideline Implementation Systems: A Systematic Review of Functionality and Effectiveness," *Journal of the American Medical Informatics Association* **6**, No. 2, 104–114 (1999).
19. I. Sim, P. Gorman, R. A. Greenes, R. B. Haynes, B. Kaplan, H. Lehmann, and P. C. Tang, "Clinical Decision Support Systems for the Practice of Evidence-Based Medicine," *Journal of the American Medical Informatics Association* **8**, No. 6, 527–534. (2001).

20. Clinical Decision Support Systems, Open Clinical, <http://www.openclinical.org/dss.html>.
21. S. C. Smith Jr., J. T. Dove, A. K. Jacobs, J. W. Kennedy, D. Kereiakes, M. J. Kern, R. E. Kuntz, J. J. Popma, H. V. Schaff, and D. O. Williams, "ACC/AHA Guidelines of Percutaneous Coronary Interventions (Revision of the 1993 PTCA Guidelines), A Report of the American College of Cardiology/American Heart Association Task Force on Practice Guidelines (Committee to Revise the 1993 Guidelines for Percutaneous Transluminal Coronary Angioplasty)," *Journal of the American College of Cardiology* **37**, No. 8, 2239i-2239lxvi.
22. R. N. Shiffman, "Representation of Clinical Practice Guidelines in Conventional and Augmented Decision Tables," *Journal of the American Medical Informatics Association* **4**, No. 5, 382-93 (1997).
23. D. Wang, M. Peleg, S. W. Tu, E. H. Shortliffe, and R. A. Greenes, "Representation of Clinical Practice Guidelines for Computer-Based Implementations," *Proceedings of the 10th World Congress on Medical Informatics (MedInfo 2001)*, IOS Press, Amsterdam (2001), pp. 285-289.
24. G. Hripcsak, "Arden Syntax for Medical Logic Modules," *M. D. Computing* **8**, No. 2, 76-78 (1991).
25. G. Hripcsak, P. Ludemann, T. A. Pryor, O. B. Wigertz, and P. D. Clayton, "Rationale for the Arden Syntax," *Computers and Biomedical Research* **27**, No. 4, 291-324 (1994).
26. M. Peleg, S. Tu, J. Bury, P. Ciccarese, J. Fox, R. A. Greenes, R. Hall, P. D. Johnson, N. Jones, A. Kumar, S. Miksch, S. Quaglini, A. Seyfang, E. H. Shortliffe, and M. Stefanelli, "Comparing Computer-Interpretable Guideline Models: A Case-Study Approach," *Journal of the American Medical Informatics Association* **10**, No. 1, 52-68. (2003).
27. M. Sordo, A. A. Boxwala, O. Ogunyemi, and R. A. Greenes, "Description and Status Update on GELLO: A Proposed Standardized Object-Oriented Expression Language for Clinical Decision Support," *Proceedings of the 11th World Congress on Medical Informatics (MEDINFO 2004)*, IOS Press, Amsterdam (2004), pp. 164-168.
28. M. Sordo, O. Ogunyemi, A. A. Boxwala, R. A. Greenes, and S. Tu, "GELLO: an Object-Oriented Query and Expression Language for Clinical Decision Support," *Proceedings of the AMIA Annual Symposium* (2003), <http://www.openclinical.org/docs/int/docs/gello.pdf>.
29. J. P. Bigus, D. A. Schlosnagle, J. R. Pilgrim, W. N. Mills III, and Y. Diao, "ABLE: A Toolkit for Building Multiagent Autonomic Systems," *IBM Systems Journal* **41**, No. 3, 350-371 (2002).
30. Arden Syntax Utilities, <http://cslxinfmtcs.csmc.edu/hl7/arden/utilities.html>.
31. ABLE Rules Package Documentation, <http://www.research.ibm.com/able/doc/reference/com/ibm/able/rules/doc-files/pkgIndex.html>.
32. J. W. Tenner, "Hungry for Data? Serve Yourself," *eServer Magazine* iSeries Edition, (February 2004).
33. E. A. Balas and S. A. Boren, *Managing Clinical Knowledge for Health Care Improvement. Yearbook of Medical Informatics 2000: Patient-Centered Systems*, Schattauer, Stuttgart, Germany (2000).
34. M. Cohen, "Progress with Diabetic Patients Undergoing Percutaneous Coronary Intervention," *European Heart Journal* **25**, No. 2, 99-100 (2004).
35. A. Abizaid, M. A. Costa, M. Albartal, H. Eltchaninoff, G. Guagliumi, L. Geert-Jan, A. S. Abizaid, A. G. Sousa, E. Wuefelfert, L. Wietze, J. E. Sousa, P. W. J. C. Serruys, and M. C. Morice, "Sirolimus-Eluting Stents Inhibit Neointimal Hyperplasia in Diabetic Patients: Insights from the RAVEL Trial," *European Heart Journal* **25**, No. 2, 107-112 (2004).
36. J. W. Moses, M. B. Leon, J. J. Popma, P. J. Fitzgerald, D. R. Holmes, C. O'Shaughnessy, R. P. Caputo, D. J. Kereiakes, D. O. Williams, P. S. Teirstein, J. L. Jaeger, and R. E. Kuntz, "Sirolimus-Eluting Stents versus Standard Stents in Patients with Stenosis in a Native Coronary Artery," *New England Journal of Medicine* **349**, No. 14, 1315-1323 (2003).
37. J. Schofer, M. Schluter, A. H. Gershlick, W. Wijns, E. Garcia, E. Schampaert, and G. Breithardt, "Sirolimus-Eluting Stents for Treatment of Patients with Long Atherosclerotic Lesions in Small Coronary Arteries: Double-Blind, Randomized Controlled Trial (E-SIRIUS)," *Lancet* **362**, Issue 9390, 1093-1099 (2003).

Accepted for publication August 7, 2006.

Published online December 27, 2006.

Xueyun Sharon Wang

IBM Software Group, 555 Bailey Ave., Santa Teresa Lab, San Jose, California 95141-1003 (swxwang@us.ibm.com). Dr. Wang is a senior scientist in the IBM Healthcare and Life Sciences division. She is the lead architect for the IBM clinical-decision-intelligence (CDI) solution, responsible for leading the design and development of on demand knowledge acquisition, management, and decision support systems within the service-oriented architecture framework. Prior to CDI, she was responsible for developing and implementing data integration strategies for the pharmaceutical and biotechnological industries. Her interests within biomedical informatics are data integration, knowledge management, and process optimization. She joined IBM in 1997 after receiving a Ph.D. degree from the University of California at San Diego. Dr. Wang is also a certified Project Management Professional (PMP).

Leonard Nayda

IBM Software Group, One Mack Drive, Mack Centre II, Paramus, New Jersey 07652-3908 (lnayda@us.ibm.com). Mr. Nayda is an executive IT architect in the IBM Healthcare and Life Sciences division. He has led several IBM initiatives focusing on systems and data integration within the health-care and life-science industries and more recently on service-oriented-architecture design and implementation for the pharmaceutical and biotechnological industry. Before this, he was responsible for developing and implementing Enterprise Resource Planning (ERP) and RFID (radio frequency identification) systems for pharmaceutical distribution organizations. He joined IBM in 2000 and has been a systems integrator and programmer for over 20 years.

Richard Dettinger

IBM Systems and Technology Group, 3605 Highway 52 N, Rochester, Minnesota 55901-1407 (cujo@us.ibm.com). Mr. Dettinger is an advisory software engineer working on IBM's health-care and life-science initiatives. He has led the Data Discovery and Query Builder team for the past several years and was instrumental in IBM's efforts in collaboration with the Mayo Clinic. Before this, he was a developer in the iSeries™ database area, working on Java and JDBC integration with the platform. He joined IBM in 1997 and has more than 12 years of experience in working with databases and object-oriented technology. ■