

Using a Service Oriented Architecture Approach to Clinical Decision Support: Performance Results from Two CDS Consortium Demonstrations

Marilyn D. Paterno MBI^{1,2,3}, Howard S. Goldberg MD^{1,2,3}, Linas Simonaitis MD^{4,5}, Brian E. Dixon MPA, PhD^{4,6,7}, Adam Wright PhD^{1,2,3}, Beatriz H. Rocha MD, PhD^{1,2,3}, Harley Z. Ramelson MD, MPH^{1,2,3}, Blackford Middleton MD, MPH, MSc^{1,2,3}

¹Clinical Informatics R&D, Partners HealthCare System, Inc., ²Department of Medicine, Division of General Medicine, Brigham & Women's Hospital, ³Harvard Medical School, Boston, MA, ⁴Regenstrief Institute, Inc., ⁵Indiana University School of Medicine, ⁶Indiana University School of Informatics, ⁷Center of Excellence on Implementing Evidence-Based Practice, Department of Veterans Affairs, Veterans Health Administration, Health Services Research and Development Service HFP 04-148, Indianapolis, IN

Abstract

The Clinical Decision Support Consortium has completed two demonstration trials involving a web service for the execution of clinical decision support (CDS) rules in one or more electronic health record (EHR) systems. The initial trial ran in a local EHR at Partners HealthCare. A second EHR site, associated with Wishard Memorial Hospital, Indianapolis, IN, was added in the second trial. Data were gathered during each 6 month period and analyzed to assess performance, reliability, and response time in the form of means and standard deviations for all technical components of the service, including assembling and preparation of input data. The mean service call time for each period was just over 2 seconds. In this paper we report on the findings and analysis to date while describing the areas for further analysis and optimization as we continue to expand our use of a Services Oriented Architecture approach for CDS across multiple institutions.

Introduction

The Clinical Decision Support Consortium (CDSC)^[1], funded by the Agency for Healthcare Research and Quality (AHRQ), seeks “to assess, define, demonstrate, and evaluate best practices for knowledge management and clinical decision support in healthcare information technology at scale – across multiple ambulatory care settings and EHR technology platforms.”^[2] The CDSC Services team hypothesized that a service-oriented architecture (SOA) approach to decision support is feasible and will provide benefits in interoperability, reliability, and reusability of knowledge content used in clinical decision support across multiple sites. To test that hypothesis, we created a remotely callable web service^[3], populated it with a set of guidelines designed to be shared among all CDSC members, and implemented it at Partners HealthCare System (PHS) and the Regenstrief Institute (RI). The service ran first in Partners’ ambulatory EMR, the Longitudinal Medical Record (LMR), from early March through mid-November, 2010. In July, 2011, Regenstrief implemented the service at a Wishard Health Services (WHS) community health center in Indianapolis, IN, in a second demonstration that ran simultaneously with the one in Partners’ LMR, through December, 2011. During these trials, we measured performance and response by instrumenting various points in the process, including retrieval and preparation of input data, population of the patient object for the rule engine and guideline execution. Performance metrics from the Indiana site were provided to us for inclusion in our research database; these data measured time needed to fetch data from local databases, assemble a HITSP-C32-compliant HL7 Continuity of Care Document (CCD)^[4] as required input to the service, and complete a “round-trip” call, i.e., send the request and receive the response. In this paper we report our initial findings in the areas of volume, performance and response times during these demonstrations.

Background

Partners HealthCare has produced a variety of clinical decision support (CDS) interventions for many years including research and quality improvement projects for specific departments or institutions^[5-8], and with an

event-driven rules engine ^[9] that has been in operation at Brigham and Women's Hospital (BWH) since 1995. The multiplicity of these projects caused difficulties with maintenance of the clinical knowledge and logic for each CDS instance. Further, interventions were generally implemented directly into the source code for particular applications, making it difficult to share intervention knowledge and logic across the various institutions and applications running in our enterprise. Even the BWH Clinical Alerting System, while built with reusable logic templates, thus separating logic from code, runs only at that institution. While the Clinical Alerting System could eventually be shared across the internally developed clinical applications in our academic medical centers and the LMR, it is not shareable with our community and specialty hospitals and care centers, most of which use a different technology and/or vendor-provided applications. To improve our ability to provide CDS that is reliable, consistently applied, and whose knowledge artifacts and technical implementation are easily maintainable, we investigated the use of a commercially available rules engine ^[10] and designed a web service to take patient data as input, prepare it for processing and inference by a commercial or open-source rules engine, and return a response in a standard format. Named the Enterprise Clinical Rules Service (ECRS), Partners' next generation decision support service is the platform being used by the CDSC to test the feasibility of a service-oriented approach for integrating and sharing decision support knowledge, producing a consistent implementation of CDS interventions to accelerate effective use of health IT with CDS, and comparing the result against the traditional direct-integration approach of knowledge embedded in application-specific code.

Two goals drive the evaluation of ECRS for the CDSC: performance and interoperability. Bates et al., in the "Ten Commandments for Effective Clinical Decision Support" ^[11] start their list with, "Speed is Everything" and go on to recommend sub second "screen flips" as a performance goal. Both Partners and Regenstrief, have made sub-second response time the standard for CDS^[12], regardless of whether it is embedded within their application or provided by a service^[13, 14]. A key issue in the use of SOA, therefore, is performance. One concern that we have heard anecdotally is that the gains of shareability that SOA offers may be offset by slower performance when compared to CDS that is embedded within an individual clinical application or system. We were unable to find support for or against this concern in published medical informatics literature. However, literature from the software industry does provide guidance on how to model and test performance, as well as some generally positive results from studies that have looked at SOA performance in other settings.^[15-17] Creating a CDS service that can demonstrate its ability to meet performance requirements is a primary goal of the CDSC Services team.

Meeting the goal of interoperability, i.e., providing a CDS service that is capable of operating against one set of guidelines for multiple consumers is also an important goal. It is not in the scope of this paper to consider the shareability of the guideline knowledge; that has been considered by other CDSC teams^[18], and their reports are forthcoming. However, system interoperability, i.e., the ability to call a web service that is external to one's application, system, or from outside a firewall is considered here in the context of service performance.

Methods

The CDS Guidelines used for this demonstration include guidance for preventive care screening for hypertension in adults, and for the chronic care management of patients with diabetes or coronary artery disease (CAD)^[19]. For the CDSC demonstrations, patient data is input to the ECRS in the form of a HITSP-C32-compliant HL7 CCD.

For the purpose of this report, a "service call" is defined to mean the time from when ECRS receives a request until it returns a response to the calling clinical application. An "ECRS round trip" will be used to specify the time period when a consumer initiates a call to ECRS from outside the Partners firewall until it receives the response. Communication with ECRS differs depending on whether the caller is inside or outside the Partners network. External callers obtain a digital certificate that allows them to send data successfully through our firewall using SOAP web services accessed over the HTTPS transport protocol. All calls are processed by ECRS components. Internal consumers, those calling from inside the PHS firewall, provide only a patient identifier; a CCD will be created for them. Other components handle translations of concepts as needed to required terminologies, classification of certain data types to facilitate rule execution, and creation of a patient object that is sent by ECRS to the rules engine. The engine executes the rules and returns a result that ECRS sends back according to an action-recommendation schema that was developed by the CDSC team.

Calling the Service: LMR (Partners). The guidelines are implemented as clinical reminders (alerts) in LMR, which has a robust set of native reminders that are presented when a clinician opens a patient's record, and updated when clinical data within the patient record changes. For example, signing a new problem of diabetes mellitus into the patient's medical record will lead to the reminders being regenerated and the screen display updated. Since alerts and reminders for hypertension, diabetes, and CAD already existed within the LMR, these alerts and reminders were suppressed for this demonstration in the participating clinics and were only presented if the ECRS call failed. Once ECRS receives a request to execute the rules, it has the capability to request a CCD from an independent service built to create these artifacts, known as the CCD Factory. The CCD Factory assembles patient data through retrieval from various source systems in the Partners clinical systems environment, translates them into the CCD specified standard vocabularies, and creates the input document. ECRS uses this CCD to execute the CDSC rules and returns the results to LMR.

Calling the Service: WHS (Regenstrief). When a patient registers at the front desk of a WHS clinic, an electronic HL7 Admission/Discharge/Transfer (ADT) message is generated. This ADT message is sent from the WHS clinic site to Regenstrief, which triggers the assembly of a CCD containing a limited data set for that patient. Regenstrief then transmits the CCD via secure mechanisms to the ECRS decision support engine operated by Partners. The ECRS processes the CCD and evaluates selected CDS rules. If any of these rules is evaluated true, a CDS reminder is generated, sent back to Regenstrief, and stored in the Regenstrief data repository.

Asynchronously (10 to 30 minutes later), the clinician treating the patient logs into the CareWeb order entry system and selects that patient's record. At that time, CareWeb obtains preventive care reminders from the data repository, and displays them to the clinician on a dashboard page.

Executing the Service: ECRS. Upon receipt of a CCD, ECRS executes the requested set of rules and generates a recommendation based on the results. The CDSC recommendation includes an assessment, with recommended actions to be performed (depending on rule), a target recipient for the recommendation, and an explanation message for the targeted recipient. The recommendation is returned to the calling application, where it is parsed and the results are presented as appropriate in the context of the receiving application.

As stated above, we measured the performance not only of the service call, but of the process of preparing data for the SOA-based CDS by all consumers, and of the ECRS round trip for external consumers. This requires capturing timing metrics for the rules service, for the creation of the CCD, and for retrieving, translating, classifying, and formatting of patient data. To enable this evaluation, both ECRS and CCD Factory capture and record start and stop times for the methods within their service, and for each downstream call made. The CCD Factory makes calls to a variety of databases to fetch patient data on allergies, lab results, problems, procedures, vital signs, medications, and demographics. In addition, it calls terminology services to translate many of these data from local code systems to standard ones such as SNOMED CT, LOINC, and RxNorm. Code systems specified were selected by knowledge, services, and demonstration team members of the CDSC, from among those included in the CCD specification.

CCDs sent from entities outside the Partners firewall, on patients not in the Partners system, must be prepared locally. To enable the evaluation of the use of CCD as input to a CDS service, the CDSC requests each consumer of these guidelines to provide performance data as described in the Introduction. These metrics are sent not as part of the service call, but in a regularly scheduled batch file, and are linked to the service logs as needed to analyze performance. Regenstrief transmits performance data once a week.

ECRS uses the CCD data to populate a patient object, which includes classifying medications, allergies, problems, and procedures into appropriate groups (such as a drug class). Each of these is done by calling a classification service. In addition to capturing time metrics for these calls, ECRS also captures times for its own methods, which include populating the patient object and calling the rules engine.

Unlike the Regenstrief call, which is asynchronous and occurs before the clinician sees his or her patient, LMR calls the service synchronously, while the patient is in the room with the clinician. The CDS Service and LMR teams agreed on a performance rate of less than 5 seconds, for the purpose of these demonstrations. ECRS set a timeout in its configuration for the maximum time it would wait for the CCD

Factory or the classification services to return, and when that threshold was reached, it returned an error message to LMR. When that occurred, LMR executed its native version of the reminders.

Performance monitoring data from ECRS and the CCD Factory are loaded into a SQL Server 2008 database once a day; this research database is also the repository for Regenstrief’s weekly transmissions. In addition to the time metrics, data sent includes identification of which rules fired, success status, failure messages, and several key values needed to match the records from the ECRS and CCD Factory together. Once the first trial period ended, it was necessary to set up filters on the evaluation database, removing test patients, duplicate records that inadvertently had been added during an upgrade from SQL Server 2005 to SQL Server 2008, records from the CCD Factory logs that either originated from other consumers of the factory service, and those did not occur during the study period.

Results

The tasks that are performed during the process of executing the CDSC-selected guidelines may be grouped into two categories. The first category is that of "Preparation Tasks": (1) fetching data from the repository; (2) normalizing (includes the translation of local codes to standard structured vocabularies), and (3) CCD generation. Preparation tasks were initiated by the ECRS service in the case of the LMR demonstration and were handled by the consumer in the Regenstrief demonstration. Table 1 illustrates CCD Preparation metrics for each consumer demonstration, summarizing the time needed to complete all preparation tasks and generate this required input to the service. Data Fetching and normalization performances are shown separately for each consumer in Tables 3, 4, and 5,

<i>CCD Preparation</i>	<i>Mean</i>	<i>Median</i>	<i>StDev</i>
LMR (Demo # 1 Mar – Nov 2010)	1645	1432	1263
LMR (Demo #2 Jul - Dec 2011)	1590	1595	557
WHS (Demo # 2)	9510	7291	6695

Table 1. CCD Preparation times in milliseconds (ms)

The second category is that of "Execution Tasks" that are handled by the ECRS. These include: (1) the total times for each service call, counted from receipt by ECRS to return of its response; (2) organizing the data for the rules engine by populating a patient object; (3) classifying certain types of patient data (problems, procedures, medications, and allergies) into appropriate groups, which takes place during the process of populating the patient object; and finally (4) rule engine execution. Table 2 presents performance data for each of these sets of tasks for both trial demonstrations. In the table, "Service Call" includes time used by all other tasks, and "Populate Patient" includes Classification time.

<i>Task</i>	LMR: Demonstration # 1				LMR & WHS: Demonstration # 2			
	<i>Mean</i>	<i>Median</i>	<i>StDev</i>	<i>95th Percentile</i>	<i>Mean</i>	<i>Median</i>	<i>StDev</i>	<i>95th Percentile</i>
Service Call Total	2331	2154	3174	5009	2314	2297	6225	3187
Populate Patient	1646	1585	1246	3799	2127	2156	745	2976
Classification	396	408	209	661	107	33	169	459
Rule Execution	34	32	32	60	44	28	1351	96

Table 2. Service execution task times (ms)

Demonstration 1. The demonstration in LMR ran from March 1, 2010 through November 16, 2010. Four LMR clinics agreed to participate in the research project; during this time period 680,062 calls were made to ECRS from these locations. Total time per call is represented by the Service Call row in Table 2. The mean time for all service calls was 2.3 seconds with a standard deviation of 3.2 seconds. The majority of

this time involved gathering the patient data by the CCD Factory service (mean time of 1.6 seconds). The expectation of the CDSC was that the consumer would prepare and submit a HITSP-C32-compliant CCD as input to the ECRS. However, given that both the ECRS and LMR exist inside the Partners firewall, we agreed that the best course would be to create a new, independent service for generating CCDs for Partners' patients ("CCD Factory"). To further streamline the implementation, LMR would pass to ECRS only a patient identifier; ECRS would use this identifier to call the new CCD Factory service and obtain a CCD. The CCD Factory would fetch, translate, and normalize the data required for building the CCD. Data retrieval and Translation times for this first demonstration are provided in Tables 3 and 4.

<i>Data Fetch</i>	<i>Mean</i>	<i>StDev</i>	<i>Median</i>	<i>95th Percentile</i>
Lab Last Known Values	361	287	333	789
Demographic Details	198	151	176	281
Medications	167	189	140	311
Problems	142	153	120	242
Procedures	311	258	225	731
Vital Signs	164	172	136	305
Allergies	153	158	133	258

Table 3. Demo # 1: Data Retrieval time by data type / service (ms)

<i>Translation</i>	<i>Mean</i>	<i>StDev</i>	<i>Median</i>	<i>95th Percentile</i>
Allergies to RxNorm	52	152	30	126
Meds to RxNorm	186	834	128	384
Problems/Procedures to SNOMED	66	121	54	383
Other terminologies to Standard Vocabulary	205	154	168	242

Table 4. Demo # 1: Translation Services execution time (ms)

Demonstration 2. The second demonstration covered the period July 1, 2011 through December 31, 2011, and included clinics from LMR and the WHS. In the interim period between demonstrations, two of the four LMR clinics had opted to remove themselves from the study, which reduced the total number of calls made to ECRS. The total for this trial was 316,685, of which 315,420 were made from LMR and 1,265 came from WHS. At Wishard, use of the CDSC was limited to a pilot group of only three clinicians who do not practice at the clinic every day. The result is a much smaller dataset in contrast to that of LMR. When we extracted the mean times as presented in Table 2, we did the same for each site to see how the two sites differed (see Table 6). Mean service call time for Regenstrief does not include data fetching or CCD preparation tasks, therefore was considerably less than for LMR. Performance data for Regenstrief's preparation tasks was aggregated and provided weekly to the research database. Table 5 shows the times reported for fetching and normalization of data. These are included in the total time needed to prepare and generate a CCD, as seen in Table 1.

Data Fetching and Normalization - RI				
<i>Task</i>	<i>Mean</i>	<i>Median</i>	<i>StDev</i>	<i>95th Percentile</i>
Fetching	6210	5012	3826	13653
Normalization	1682	561	3346	11827

Table 5. Data Preparation Tasks for Regenstrief (ms)

Demonstration # 2	Partners HealthCare				Regenstrief Institute			
Task	Mean	Median	StDev	95th Percentile	Mean	Median	StDev	95th Percentile
Service Call	2322	2303	6247	3188	1174	973	1157	2379
Populate Patient	2136	2163	734	2976	884	728	1073	1742
Classification	104	33	161	454	541	472	452	1156
Rule Execution	43	23	1356	95	61	46	92	138

Table 6. Service Execution times by Consumer Site (ms)

Discussion

The CDSC has successfully implemented ECRS, a decision support rules service which delivers preventive care reminders to electronic medical record (EMR) applications. As we describe in this paper, this rules service is being used by two very different EMR applications. The LMR is operational at Partners Health Care clinical sites in the Boston area; the RI CareWeb system is operational at Wishard Health Services centers in Indianapolis. Both systems are able to provide the preventive care reminders for real patients in a production environment. Despite the marked differences between these two systems, both are able to interact with the same rules service by adhering to a common set of specifications.

Although participation in the CDSC demonstration by the LMR clinics was voluntary, the service nonetheless was implemented in a production system, i.e., used in real clinical care settings while clinicians were seeing patients, and needed to meet adequate performance requirements. During the first demonstration there were times when the complaint of delay caused LMR to shut down the service while an investigation into the cause was pursued. Although the source of the problem was generally found not to be due to the ECRS service itself, nonetheless ECRS did not function for a few brief periods of time while the causes of slowness were being investigated. It was primarily due to this problem that two of the clinics chose to opt out of the study.

There are some limitations to the analysis that we are unable to remedy at this time, due to the complexities of using SOA architecture. Few published studies are available that have tested the performance of SOA-based services for decision support, though there is growing interest in its feasibility^[20-22], and proposed architectures to support it^[23-25]. The importance of appropriate testing of SOA has been gaining attention, and a variety of methodologies are being proposed as best practices or solutions offered^[26-29]. We capture times from multiple points in the process, and as we analyzed each, we have identified gaps of unreported time. We assume that some are due to network latency, and are in the process of identifying where each occurs and how to measure it. As one example, at each point where another network or technology is needed, there is the potential for additional time to be lost and performance to degrade. In another example, a service may perform multi-threaded tasks, yet the total time it records for the overall task is greater than the time needed for the longest task. For instance, ECRS starts up classification threads simultaneously for problems, procedures, medications, and allergies, and waits to move to the next task until the last classifier completes. Yet the measured time for “classification” is always greater than the length of time used by whichever thread (e.g., problems) took longest. While we do not yet have answers to why these occur, we think that identifying that they occur, and where, informs the SOA testing process and contributes to our understanding of what needs to be considered when implementing SOA-based CDS.

The first demonstration took place within our firewall at Partners, whereas the second and all future CDSC demonstrations will be across the firewall to remote consumers of the ECRS. This will make it all the more important to identify and isolate the places where latency occurs. We need to be able to test the feasibility of such services in a world where patients move about and receive care in multiple institutions within and outside a single delivery network, and where having high-reliability, standards-based decision support available is ever more desirable. As part of the evaluation of each demonstration in the CDSC, we must investigate gaps in time thoroughly so that we can identify every potential area of latency, isolate inefficiencies, and resolve problems early. A careful review and assessment of the proposals for SOA testing that are available in the software industry with respect to the feasibility of making use of them in our clinical systems should prove to be a useful exercise as the CDSC continues to implement and evaluate these CDS demonstrations.

ECRS makes use of services external to itself to accomplish much of its work. Included among these is the call to a CCD creation “factory”, which itself calls downstream services that fetch data from the appropriate databases. A Patient “factory” instantiates and populates a patient object, during which process it also calls services external to itself; these services translate and classify concepts into supported terminologies and classes prior to rule execution. The CCD Factory, the services it calls, and the services that the Patient Factory calls are all managed by teams other than ECRS.

We conclude that the ECRS performed well; the rules execution was accomplished consistently and in a short (sub-second) time. Measured service call times for the two sites (Table 5) differ between the two demonstration sites, which necessitates explanation. At Partners, ECRS requests the generation of a CCD, therefore that time is included in the service call. At Regenstrief, the CCD preparation is done prior to calling ECRS, therefore it is not part of the ECRS service call, nor of the ECRS round trip. As a result, the mean time for an ECRS service call from Regenstrief, just under 1.2 seconds, is a better measure of overall ECRS performance. Interestingly, the average time for classification, which occurs during the process of populating the patient object, was much higher for WHS data than for LMR data. We have not yet identified the reason for this, and will continue to analyze it as part of the next demonstration. Bottlenecks that occurred for Partners were in data retrieval and translation, which occur as part of the CCD generation process in the CCD Factory. Some of the services called by ECRS or the CCD Factory were prepared as API calls for local use within the legacy system on which our systems have been built, and not for SOA web service use, therefore have not been tuned for optimal performance. Examined in that context, they appear to perform well; it is in moving to the future of web services that many will need to be reviewed, optimized, and perhaps re-architected and rewritten. The CCD Factory service has been updated; new methods expected to improve its performance will be in use prior to the next CDSC demonstration.

While we draw positive conclusions with respect to the ECRS itself, we noted exceptions in the aggregate time for external services that caused some variances in performance. Examination of the individual services did not reveal a single source of delay; exceptions are noted in many, if not all, the services used. As noted previously, these services are not under the control of the ECRS team to improve, nor in fact are we certain that these exceptions result from issues related to the performance of downstream services themselves. It is at least as likely that exceptional delays may be caused to some degree by network load or other infrastructure issues. We did not remove exceptions from the descriptive statistics presented here; we consider it important to note their occurrence, and to discuss the impact that they have on our consumers.

Use of the CCD as input to a CDS service was determined at the start of the CDSC project in 2008; the primary reason for it was the plan to use national standards wherever possible, and on a practical level, to select something that was currently in use and held the greatest possibility of vendor support. At the time, the two best options were CCD and vMR; however, although vMR was an attractive model, it was not in actual use anywhere and was not identified as a standard. Given that the CCD was already an approved standard, recommended by HITSP and Integrating the Healthcare Enterprise® (IHE), with support for it required by the Certification Commission for Health Information Technology (CCHIT) in its ambulatory electronic health record certification criteria roadmap, we considered that there existed a strong likelihood of vendor support for CCD use. Preparing a CCD, as has been noted above, involves fetching, translating, and normalizing data in addition to creating the required sections and organizing the CCD. Table 1 shows the mean times used for this task from each consumer during each demonstration. Note that there is a large variation between the time needed to generate a CCD at Partners compared with Regenstrief. During the demonstration, Regenstrief found that higher times were associated with patients whose electronic medical records contained large numbers of observations (e.g., vital signs, lab results) over the more than 30 years WHS had been using the Regenstrief Medical Record System. Regenstrief subsequently has refined its CCD generation processes to limit the timeframe used for retrieving patient data. Although the time needed to prepare the CCD averaged 9.5 seconds, this did not have an effect on either clinician or patient, as the process occurred asynchronously between clinic check-in and physician login.

For CDSC consumers who call ECRS from outside the Partners firewall, it is the execution services that are called by ECRS, especially classification and patient object population, that need to perform with the most efficiency. We will need to capture time metrics for the transport of data to and from the service and study those results carefully in order to understand how best to provide SOA services for clinical decision support. For the second demonstration, ECRS round trip from WHS to ECRS to WHS took an average of 2.8 seconds. Given the average service execution time of 1.2 seconds for their service calls, there is a gap

of 1.6 seconds that is unaccounted for, and that we assume is the transport time in both directions across the internet. We plan to review the performance testing methodologies being used across the IT internet industry to determine how best to find and track the movement of data and understand what needs to be accounted for in planning for SOA-based CDS services, and to continue tracking these data, both with Regenstrief and with the next set of CDSC consumers.

Both Wishard and LMR continue to call the ECRS, and for the next phase of the project each will look to increase their participation, either by adding clinics or clinicians to the demonstration. In addition, a third CDSC member, this time a clinic using a vendor-provided EHR will, with the support of the vendor, join the demonstration.

Conclusion

Initial performance data from the CDSC demonstration of decision support guidelines in Partners' LMR and at WHS indicates that the web service can and does perform well. Our overall evaluation of the feasibility of using the SOA approach to CDS is positive. However, multiple dependent services that are used by the ECRS need to be optimized, new processes in use across the hardware and software platforms need to be monitored and additional sources of latency between service calls need to be identified and studied. We continue to analyze these performance data, review areas where we need to monitor additional processes, and will add new data points to the performance logs as we move into the next demonstration project.

Acknowledgements

This work was supported under contract #HHSA-290-2008-10010 from the Agency for Healthcare Research and Quality. The authors wish to acknowledge the work of Frank Chang, who created and maintains the evaluation database and provides technical assistance to the team. The authors further acknowledge Andrew Martin, who provided the performance data of the Regenstrief systems to Partners.

References

1. Middleton B. The Clinical Decision Support Consortium. *Stud Health Technol Inform*. 2009;150:26-30.
2. The Clinical Decision Support Consortium Website. 2008 [cited 2011 3/11]; Available from: <http://www.partners.org/cird/cdsc/>
3. Paterno MD, Maviglia SM, Ramelson HZ, et al. Creating Shareable Decision Support Services: An Interdisciplinary Challenge. *Proc AMIA Annu Fall Symp*. 2010;2010:602-6.
4. Hitsp Summary Documents Using HI7 Continuity of Care Document (Ccd) Component. 2009 [cited 2012 3/15]; Available from: <http://wiki.hitsp.org/docs/C32/C32-1.html>
5. Abookire SA, Teich JM, Sandige H, et al. Improving Allergy Alerting in a Computerized Physician Order Entry System. *Proc AMIA Symp*. 2000:2-6.
6. Maviglia SM, Zielstorff RD, Paterno M, et al. Automating Complex Guidelines for Chronic Disease: Lessons Learned. *J Am Med Inform Assoc*. 2003 Mar-Apr;10(2):154-65.
7. Poon EG, Blumenfeld B, Hamann C, et al. Design and Implementation of an Application and Associated Services to Support Interdisciplinary Medication Reconciliation Efforts at an Integrated Healthcare Delivery Network. *J Am Med Inform Assoc*. 2006 Nov-Dec;13(6):581-92.
8. Paterno MD, Maviglia SM, Gorman PN, et al. Tiering Drug-Drug Interaction Alerts by Severity Increases Compliance Rates. *J Am Med Inform Assoc*. 2009 Jan-Feb;16(1):40-6.
9. Kuperman GJ, Teich JM, Bates DW, et al. Detecting Alerts, Notifying the Physician, and Offering Action Items: A Comprehensive Alerting System. *Proc AMIA Annu Fall Symp*. 1996:704-8.
10. Goldberg HS, Vashevko M, Pastilnik A, et al. Evaluation of a Commercial Rule Engine as a Basis for a Clinical Decision Support Service. *AMIA Annu Symp Proc*. 2006:294-8.
11. Bates DW, Kuperman GJ, Wang S, et al. Ten Commandments for Effective Clinical Decision Support: Making the Practice of Evidence-Based Medicine a Reality. *J Am Med Inform Assoc*. 2003 Nov-Dec;10(6):523-30.
12. Doherty W, Thadani A. The Economic Value of Rapid Response Time. 1982 [cited 2011 3/13]; Available from: <http://www.vm.ibm.com/devpages/jelliott/evrrt.html>.

13. Biondich P, Mamlin B, Tierney W, Overhage J, McDonald C. Regenstrief Medical Informatics: Experiences with Clinical Decision Support Systems. In: Greenes RA, ed. *Clinical Decision Support: The Road Ahead*. Burlington, MA: Elsevier, Inc. 2007:111-26.
14. Friedlin J, Dexter P, Overhage J. Details of a Successful Clinical Decision Support System. *AMIA AnnuSymp Proc* 2007:254-8.
15. Kumar S, Dakshinamoorthy V, Krishnan M. Does Soa Improve the Supply Chain? An Empirical Analysis of the Impact of Soa Adoption on Electronic Supply Chain Performance. *40th Annual Hawaii International Conference on System Sciences (HICSS'07)* 2007:171b.
16. Papazoglou M, Traverso P, Dustdar S, Leymann F. Service-Oriented Computing: State of the Art and Research Challenges. *Computer*. 2007 Nov. 2007;40(11):38-45.
17. Liu Y, Gorton I, Zhu L. Performance Prediction of Service-Oriented Applications Based on an Enterprise Service Bus. *31st Annual International Computer Software and Applications Conference* 2007:327-34.
18. Boxwala A. Multilayered Knowledge Representation as a Means to Disseminating Knowledge for Use in Clinical Decision-Support Systems. *AMIA Spring Congress*. 2009 May 28-30.
19. Cds Consortium Knowledge Management Portal. 2008 [cited 2012 3/13]; Available from: <http://cdsportal.partners.org/cdscsearch.aspx>
20. Borbolla D, Otero C, Lobach D, et al. Implementation of a Clinical Decision Support System Using a Service Model: Results of a Feasibility Study. *Stud Health Technol Inform*.2010(160(Pt 2)):816-20.
21. Jahnke-Weber J, Price M, McCallum G. Making Available Clinical Decision Support in Service-Oriented Architectures.
22. Ongenae F, De Backere F, Steurbaut K, et al. Towards Computerizing Intensive Care Sedation Guidelines: Design of a Rule-Based Architecture for Automated Execution of Clinical Guidelines. *BMC Med Inform Decis Mak*.2010(Jan 18):10:3
23. Wright A, Sittig DF. Sands: A Service-Oriented Architecture for Clinical Decision Support in a National Health Information Network. *J Biomed Inform*. 2008 Dec;41(6):962-81.
24. Kawamoto K, Honey A, Rubin K. The HI7-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 Nov-Dec;16(6):874-81.
25. Paterno MD, Schaeffer M, Van Putten C, et al. Challenges in Creating an Enterprise Clinical Rules Service. *AMIA Annu Symp Proc*. 2008:1086.
26. Rodriguez J, Demsak D. Lightweight Soas: Exploring Patterns and Principles of a New Generation of Soa Solutions. *The Architecture Journal* 2009 [cited 2011 3/13]; Available from: <http://msdn.microsoft.com/en-us/architecture/bb426891>
27. Barber S. Soa Testing Challenges. [Webinar] 2006 5/9/2006 [cited 2011 3/13]; Available from: http://www.perftestplus.com/resources/SOA_challenges_ppt.pdf
28. Pillars of Soa Testing. [cited 2011 3/13]; Available from: http://www.crosschecknet.com/soa_testing_pillars.php
29. Testing Service-Oriented Architecture (Soa) Applications and Services. [white paper] [cited 2011 3/13]; Available from: <http://www.powertest.com/files/testing-service-oriented-architecture-applications-and-services-whitepaper.pdf>