Clinical Decision Support Systems
Enabling Medication Reconciliation

Lisa Beckman, Gordon Bleil, Hugo Fernandes, Lisa Nelson and Jeremiah Rothschild
11/21/2010
## Contents

Introduction ................................................................................................................................................... 1  
Approach ....................................................................................................................................................... 1  
System Design .............................................................................................................................................. 1  
  Use Case and Workflows .......................................................................................................................... 1  
  Functional Design ..................................................................................................................................... 3  
  Data, Knowledge and Logic Modeling ..................................................................................................... 3  
  Prototype: User Interface Modeling .......................................................................................................... 4  
System Architecture ...................................................................................................................................... 5  
  Make Versus Buy ...................................................................................................................................... 6  
  Knowledge Acquisition and Maintenance ................................................................................................. 6  
  Physical Design .......................................................................................................................................... 7  
  Scalability and Sustainability .................................................................................................................... 7  
Evaluation and Results .................................................................................................................................. 8  
Recommendations and Conclusions ................................................................................................................ 9  
Works Cited ................................................................................................................................................ 10  
Glossary ...................................................................................................................................................... 12  
Appendix A. Figures ................................................................................................................................... 13  
Appendix B. Prototype demo ...................................................................................................................... 19
Introduction

The number of successful implementations of electronic health records (EHR) is increasing and focus is shifting from data storage and retrieval to reliable data flow for coordination of care. Accordingly, a new class of application called a clinical decision support system (CDSS) is emerging. The CDSS includes a broad set of systems, applications and processes that enable health professionals to make decisions regarding care (HIMSS). A CDSS focusing on the transfer between care venues (that is, transitioning from inpatient care to ambulatory or post-acute care settings) is needed to reduce errors. Specifically, the area of focus is medication reconciliation which is often a manual process that exposes patients to significant risk. In fact, while the Joint Commission established the “goal for hospitals to perform medication reconciliations at all transitions of care in 2005, many organizations still struggle to implement effective processes” (The Joint Commission, 2010).

Currently providers have to glean the necessary information for decision making from dispersed sources. Further, complicating matters is the predominantly manual process which is highly susceptible to errors. Ideally, the provider would be presented with key information, on-demand, optimally organized, filtered for the users’ activity and context sensitive to improve medication reconciliation. Moreover, the system should provide improvements in patient care to satisfy the “Meaningful Use” requirements (Liaw & Pradhan, 2010). The objective of this envisioned CDSS is to provide a reconciled medication list in electronic form to bridge the information gap with a focus here on facilitating medication reconciliation at discharge from an inpatient facility to a nursing home. The system is intended to support decision making that optimizes medication selection of the reconciled list using knowledge of indications for use, drug therapeutic classifications, available medication formulary information and other outcomes data. Our project should demonstrate the impact a CDSS can have on the exchange and interoperability of medication information, while simultaneously showing how narrowly focused and deep a CDSS must be to address the decision support requirements of the medication reconciliation function.

Approach

To prove CDSS technology can attain goals established for medication reconciliation, a concept for a CDSS was designed to meet the Joint Commission (TJC) requirements and evaluation criteria were drafted to determine if the system could adequately and appropriately support medication reconciliation. Additionally, the impact of the design was logically assessed to understand what reasonable expectations might be for reducing Adverse Drug Events (ADE’s), while increasing the rate of performing medication reconciliation at discharge, and also reducing the rate of errors made during the medication reconciliation process.

System Design

Designing the CDSS began with analysis of needs and goals for the system, which considered the use case and workflows for both current state and the envisioned future state. This work product established an initial functional system design. A multidisciplinary collaboration to flesh out information and logic required to make decisions was performed. Using the data and logic requirements, a comprehensive knowledge schema was developed to guide implementation and plan for the acquisition and maintenance of the knowledge assets that were to comprise the system. Finally, completeness of the design was logically validated through a series of prototype models of the user interface. This served the purpose of graphically depicting workflow and envisioned functionality and permitted designers to clarify expectations.

Use Case and Workflows

The use case and workflow maps clarified the current process and future process. It highlighted the importance of medication reconciliation as a patient safety measure, especially for elderly patients with chronic illnesses who often take medications prescribed by multiple providers. For example, having to
reconcile medication lists is a critical step in coordinating care since these patients are typically seen by multiple providers who independently write prescriptions and do not adequately communicate. Moreover, it is also an important step when a patient is transitioning from one level of care to another. Medication reconciliation requires the discharging physician to perform these five tasks: 1. Develop a list of current medications; 2. Develop a list of medications to be prescribed; 3. Compare the two lists; 4. Make clinical decisions based on the comparison; and 5. Communicate the results (The Joint Commission, 2010).

**Figure 1. Current State Workflow for Medication Reconciliation**

The workflow analysis illustrates that the current medication reconciliation process is often manual, paper-based and the needed information is not readily available (shown in Figure 1). To be performed properly, medication reconciliation requires physician interaction with a broad set of knowledge bases. Not surprisingly, the process is frequently incomplete. Because of the clear inefficiencies and risks for error in the process, an opportunity exists to leverage clinical best practices to improve performance.

**Figure 2. Future State Workflow**

The envisioned process includes data-driven decision support and time-driven triggers related to maintenance and system function. Inference processing flags duplicates and other potential issues. Comparisons are made with the available formulary (based on insurance and facility) as well as the most significant issues reported by the nursing home. This information is based on review of the nursing home’s clinical repository analysis and standard safety references (e.g., Beer’s list and Canadian list).
The EMR’s CPOE is invoked to check drug interactions as the physician makes selections during the reconciliation of medications. The system provides physicians with information about problem indications and medication effects which supports the decision making process during the medication reconciliation.

The envisioned system supporting the future workflow highlights physician options when complex sets of medications need to be compared. The system helps to reduce errors by specifying and clarifying the available options. The output promotes interoperability and better communication, a key element of meeting “Meaningful Use” (Federal Register, 2010).

Figure 2 depicts the enhanced workflow, designed to be more efficient and effective than the current process. The flexible entry methods and multiple options limit perceived threats to professional autonomy. Doctor-patient interactions are unaffected. The implementation plan assures adequate availability of hardware, technical support and training. As users experience the system they should hopefully find it integrates well into their workflow and that clinical messages are relevant and the information timely. It is expected that a system meeting these criteria should be endorsed by the user community and acceptance should be accelerated (Moxey, Robertson, & Newby, 2010).

**Functional Design**

The functional design forms the cornerstone for the CDSS because all subsequently identified system requirements trace back to these high-level capabilities. A requirements traceability matrix was used to describe and follow the life of a requirement from its origins, through its development and specification, to its subsequent deployment and use (Palo, 2003). Applying requirements traceability practices allowed us to create a design that anticipated the need for ongoing change and product evolution.

### Figure 3. Functional Design for the Medication Reconciliation CDSS

- **Communicate with the EMR system**
  - Query Patient Data
  - Admission Medication List
  - Active Medication List
  - Return Discharge Medication List
- **Optimize Medication List**
  - Combine Medication Lists (Admission + Active)
  - Present info efficiently for Physician assessment
  - Identify additional optimization issues: therapeutic duplication, adverse reaction, therapeutic indication
- **Interact with Discharging Physician (GUI)**
  - Present suggestions
  - Permit suggestion over-ride, select different decision criteria
  - Allow manual additions, changes, eliminations of medication at time of patient review
- **Manage Knowledge Base (GUI)**
  - Maintain Medication Terminology – External Knowledge Sources
  - Maintain Inference Rules
- **Consume System-Generated Performance Info**
  - Collect feedback to drive continuous improvement

### Data, Knowledge and Logic Modeling

A collaborative approach between end users and developers was used to ensure the design met both functional and clinical expectations. The analysis involved a detailed and iterative process where users described and demonstrated needs, and designers capture them. This type of communication required tools to organize the gathered information and to validate what was heard.

To examine the process of medication reconciliation, we built Excel worksheets to facilitate collaboration. We used the tool to organize and shape the information into an early model of the problem. It enabled us to understand the function of the Medication Reconciliation List and clarified the types of decisions required to transform that list into a Reconciled Medication List for a patient’s discharge. More
importantly, it provided an active representation of the best practice steps documented for medication reconciliation by TJC (The Joint Commission, 2010).

Figure 4. Excel model used to communicate detailed functionality for medication reconciliation

The Excel model was then developed into an information model that considered the processing logic required by the system. It reflected the information objects, and how they were related, classified and categorized. The system logic identified the logic criteria and processing rules needed to drive the inference engine of the system.

Figure 5. Information schema and system logic for the CDSS

Prototype: User Interface Modeling

Modeling the User Interface allowed deeper assessment of the system’s design. It also identified subtle but significant aspects of functionality that required additional system knowledge elements to be included in the design and others to be explicitly excluded. Due to time constraints for the project, dose and route considerations were deferred. Addressing these requirements in the medication reconciliation holds
potential to substantially benefit patient safety, but adding them to the design would have represented substantial additional complexity.

**Figure 6. Prototype user interface screens.**

The user interface prototype demonstrated the planned functionality of the system and made the improvements for the medication reconciliation process more tangible. It showed how the tool presented physicians with the complete set of medications for patient discharge. It provided a visualization of how flags would identify potential issues. It enabled simulation of what it would be like to see the possible problem indications along with medication effects while selecting among multiple medications for a patient. It also illustrated how providers have the opportunity to document their thinking and explain the decisions.

The prototype served to validate with users that the system, as designed, would improve the way physicians performed medication reconciliation compared to how it is done today. Further, it was designed to capture user feedback to gain input for future use.

Developing a software prototype allowed us to verify our design and it provided valuable planning information for designing the software, hardware and “liveware” components of the system. Liveware is a term Ponedal coined to describe the people who develop, maintain and use the system as a third but integral component of a system working in conjunction with the software and hardware (Ponedal, 2002).

**System Architecture**

The system design phase clarified the requirements of the medication reconciliation CDSS in terms of the logical components of the solution.

The user interface (UI) supports physicians reconciling patient medication lists at the time of discharge. It also gathers performance information while the system is being used so that administrators can review and process that information back into the knowledgebase. The inference engine processes the medication information. It utilizes information gathered from the clinical data repository about the patient and decision rules and criteria available from the knowledge base. It serves the UI with the information to be presented to the physician (Cho & Kim, 2010). The interface engine serves as a data traffic router and translator, providing a seamless interface between the knowledge base, the clinical repository and the EMR. Medication lists communicated between the CDSS and EMR utilize HL7 messaging protocols and are formatted as CCD compliant documents. External information sources are processed with a UI devoted to interacting with the knowledgebase. The knowledgebase stores the information sources, the inference rules and decision criteria attributes.
Make Versus Buy

After surveying available system capabilities to support our design, it was determined an add-on module to an EHR that would work with CPOE was optimal. If the CPOE provided drug-drug interaction testing that a customer preferred, we would supply an interface that would allow the CPOE’s drug interaction alert decision criteria to be utilized by our CDSS. If the CPOE didn’t offer that capability, then our system’s drug interaction checks would be used.

A service-oriented architecture (SOA) was selected for the design of the software. SOA offered several advantages over other system architectures. It provided greater modularity which enabled the processing to be evenly distributed among the various logical components of the system. Working in harmony with a hospital’s existing electronic medical record was a consideration. The SOA architecture would integrate more easily with other EMR environments. Generalized services would make it possible to take advantage of information stored in an EMR’s clinical repository (Wright, 2008).

Knowledge Acquisition and Maintenance

To minimize cost and development effort, an open source tool was used to provide the knowledgebase and administrator’s UI. The tool provided functionality to import terminology standards and other information sources into its knowledge representation structure. The software developers kit (SDK) provided an application interface (API) which allows the CDSS to access information in the knowledgebase (Apelon, 2010). This approach was chosen to provide the capabilities needed to implement and augment the information schema while keeping costs down.

Figure 8. Knowledge representation model in the open source KB tool
A subscription service for the needed standard terminologies is used, enabling disparate standards to be delivered in a single format that supports uniform treatment within the knowledgebase (Apelon, 2010). The information acquired includes RxNorm and SNOMED CT which are the standards designated by the HITECH Act under Meaningful Use for medications and problems, respectively (Carter, Brown, Bauer, Elkin, Erlbaum, Froehling, Lincoln, Rosenbloom, Walmer-Roedler, Tuttle, 2006). RxNorm meets the needs of medication reconciliation, e-prescribing, formulary checking and transmission of medication information (Carter et al.). RxNorm is organized around normalized names for clinical drugs. The names contain information on ingredients, strengths, and dose forms (RxNorm, 2010).

National Drug File – Reference Terminology (NDF-RT) is used because early adopters of RxNorm, such as the FDA and VA are utilizing linkages to NDF-RT to classify drugs from multiple perspectives (Carter et al.). A study of the categorical information in pharmaceutical terminologies reported that NDF-RT’s categorical reference model accommodates more than 76% of the information identified in drug class names (Carter et al.). One of the major challenges facing the use of NDF-RT in this CDSS remains disagreement regarding the classifications of physiological effect and therapeutic intent. The WHODRUG dictionary should additionally be acquired as another medication classification source based on Anatomic-Therapeutic-Chemical (ATC) classification (Dark).

Other external lists, such as the Beers Criteria from the Duke Research Institute, provide information on drugs that are potentially inappropriate for the elderly (Beers Criteria, 2010) and would need to be manually created. Drug formulary information would be solicited in an electronic feed from insurance carriers or entered manually.

Because of the complexity associated with maintaining the many specialized information assets in the knowledgebase, the partnership between user and developers really never ends. Ongoing interaction should be needed over the full life of the system.

**Physical Design**

The physical design of the system takes requirements of both the developed and purchased aspects of the solution. Interfaces to external information sources and the EMR system ensure the CDSS functions as a modular solution that can be added to any hospital IT environment.

**Figure 9. Physical design for the medication reconciliation CDSS**

**Scalability and Sustainability**

A performance study captures and quantifies volume information to determine the impact of the number and type of transactions to be processed (e.g., to support 50 discharges in one day and the timing associated with those transactions as well as other system administration functions) (Wright, 2008). Growth in the number of decision criteria, inference rules and information sources was anticipated so the
system would be versatile and scalable. Affordability, from a cost of ownership perspective, was also a key design consideration.

**Evaluation and Results**

Have we built the right system? Did we build the system right? Validation confirms that “a model is a representation of the phenomena being modeled and that it is adequate for the purpose of the study of which it is a part”, i.e. “Building the right system.” Verification confirms that “the actual model that has been constructed is indeed the one intended to be built.” in other words “Building the system right” (Miser & Quade, 1988). Issues of validation and verification must be addressed in order to confirm that the design of our CDSS is sound and meets the goals we defined for it.

To validate the product and prove “the right system” was built, the Medication Reconciliation Review tool from the Institute for Healthcare Improvement should be used. The tool “provides step-by-step instructions for conducting a review of closed patient records to identify errors (ADE’s) related to unreconciled medications” (IHI, 2004) and allows baseline data to be gathered prior to implementation. Post implementation data should be collected in a standardized and objective fashion that can be used to determine whether the system meets the established goals.

A retrospective evaluation including the following steps should be performed:

- Identify a multidisciplinary team consisting of, at a minimum, a nurse, a pharmacist, and a physician.
- Obtain a set of 30 closed patient records (each with a minimum stay of 3 days), using as random a selection process as possible.
- Have each team member review one-third of the patient records, counting errors due to unreconciled medications. (For consistency, the criteria for determining errors due to unreconciled medications are spelled out in the tool- see page 3, sections 3.a-f). (IHI, 2004)

The data collection tool should include: the patient record number, the review date, the number of reconciliation errors at discharge, and whether or not a readmission occurred related to an ADE. Furthermore, in order to evaluate improvements in communication, information gathered it should include whether or not a discharge medication list was delivered to any of the following: the patient and/or guardian, the nursing home, other involved providers. The post-implementation data also should note whether or not the medication reconciliation tool was used. Metrics should be collected and tallied prior to implementation, then again post implementation at monthly intervals for 6 months, then quarterly for 2 years.

These metrics should also be used to show that the CDSS meets Meaningful Use requirements. In other words, it should contribute greatly in the effort to “improve care coordination” (as outlined in the Meaningful Use Matrix, 2009), by providing “medication reconciliation at each transition of care from one health care setting to another” as well as “clinical decision support at the point of care”. It should also contribute greatly to “improving quality, safety, and efficiency” by helping to maintain a fully reconciled active medication list that is shared with all providers as well as the patient.

To assess user satisfaction, stakeholders (e.g., clinicians, pharmacists, patients, nursing homes, and other providers) should be surveyed monthly for the first 6 months, then quarterly for 2 years. These surveys should elicit qualitative feedback regarding ease of use, quality of information, quality of presentation, and perceived benefit to professional practice and to patient care and outcomes.

Furthermore, a time study for a 1 month period prior to go-live, quantifying the time spent on medication reconciliation upon discharge from the hospital, as well as transition to the receiving facility should be conducted to determine value of the solution to the enterprise. The time study should also be repeated after “go-live”. The previously described metrics (that is, number of ADE’s and number of readmissions related to medication reconciliation at transition of care) should be used to quantify savings.
In order to verify the product and prove the system was “built right”, a software development quality system should be used to test the system’s functionality at all stages of development, implementation and maintenance. Toward that end, stringent requirements traceability should be implemented at all stages of development, as previously described. Multiple system checks should be performed prior to “go-live” to ensure that the CDSS is fully functional and that there are no obvious glitches. On-going error tracking, with swift response from our “liveware” team to apply “fixes” and updates to the system in a timely manner should also be performed. The “liveware” team should oversee subscription maintenance and contracts with entities that support the knowledgebase and intercept any notices of glitches, errors, or quality issues that may be reported.

Additionally, the system has been designed to include a fix-it button that allows end users to report problems or complaints in real-time. The function automatically “grabs” a screen shot and transmits internal buffers, tracking system activity at the point of the report. The “liveware” team receives the information to augment the evaluation of the reported problem, enabling a faster response that appropriately matches the level of severity for the issue.

The quality system governs software development and bug fixes as well as the entire change management cycle for investigating, fixing, testing deploying new versions of the application and training users on all changes.

**Recommendations and Conclusions**

Clinical decision support tools, including the one proposed herein, are viable options and offer promise for realizing value from the EHR investment. The design validated that it is possible to streamline and automate medication reconciliation similar to how financial institutions reconcile various statements. However, there are limitations to CDSS tools that must be considered in order to support safe, effective, patient-centered care.

The CDSS proposed solution could be put into operation in short order with a modest level of effort and support from a skilled hospital IT department in partnership with trusted clinicians. It is reasonable to expect implementation in six months, with evaluation at one year, and a useful life of 3 to 7 years.

Communication of the utility of the solution is just as important as the design. Providers must be aware that the solution would have limitations and could provide incomplete or misleading recommendations, due to the presence or absence of data and the quality of the clinical data training set. Finally, there is no dominant design on medication classifications or universally accepted method of describing indications for medication use. This increases the risk of confusion or miscommunication between information systems.

In the near term, this system would provide process consistency and the ability to meet the criteria for National Patient Safety Goal #8 -“to completely reconcile medications across the continuum of care” (The Joint Commission, 2010). In the future, the capabilities of the system could expand to permit recommendations and error checking related to dosing based on stage of illness (e.g. higher doses for critically ill patients or other factors). The decision support capabilities of the tool also could be expanded to consider factors related to the patient’s environment of care and stage of illness, not just the diagnosis and drug interactions. Ultimately, the most significant potential for this system would come from extending its use to all transition points in the care process.
Works Cited


http://himssclinicaldecisionsupportwiki.pbworks.com/Use+of+high-risk+medications+in+the+elderly+%28cf+Beers+criteria%29


http://en.wikipedia.org/wiki/Clinical_decision_support_system

**Glossary**

**Effects (of medication)** - the set of clinical or biological consequences that result from taking a medication.

**Indications** - the reasons for which a treatment (e.g. medication) is commonly used. The indication may not be the same as the Problem for which the treatment is prescribed. *Atrial fibrillation causes blood clots to form in the atrial chambers of the heart. This causes strokes in 25% of untreated patients. Coumadin is used to prevent blood clots and therefore is used to prevent the clots that cause the strokes. The Problem list should usually include Atrial Fibrillation, but not ‘prevent blood clots’ or ‘stroke prevention’. It has been acceptable practice to use the Diagnosis that causes the risk factor as the indication for the (medication) treatment.*

**Liveware** - the people who develop, maintain and use the system as a third but integral component of a system working in conjunction with the software and hardware. (Ponedal, 2002)

**Problem** – a diagnosis assigned to a patient, or a statement regarding a circumstance which may affect the persons health e.g. risk of blood clots, low education level, low income level. The most common use of the Problem list is as a list of Diagnosis.
Appendix A. Figures

Figure 1. Current Workflow

Figure 2. Future Workflow
Functional Design

- Communicate with the EMR system
  - Query Patient Data
    - Admission Medication List
    - Active Medication List
  - Return Discharge Medication List

- Optimize Medication List
  - Combine Medication Lists (Admission + Active)
  - Present info efficiently for Physician assessment
  - Identify additional optimization issues: therapeutic duplication, adverse reaction, therapeutic indication

- Interact with Discharging Physician (GUI)
  - Present suggestions
  - Permit suggestion over-ride, select different decision criteria
  - Allow manual additions, changes, eliminations of medication at time of patient review

- Manage Knowledge Base (GUI)
  - Maintain Medication Terminology – External Knowledge Sources
  - Maintain Inference Rules

- Consume System-Generated Performance Info
  - Collect feedback to drive continuous improvement

Figure 3. Functional Design for the Medication Reconciliation CDSS

Figure 4. Excel model used to communicate detailed functionality for medication reconciliation
Information Model

Figure 5A   Information schema

System Logic

- **Completeness**
  - if medication AdmissionMedList? then mark-x
  - if medication Active? then continue
  - if medication Stopped? then stop

- **Safety**
  - if medication DCIinteract? then flag-o, group(conlict)

- **Duplication**
  - if medications DupTherap? then flag-o, group(class)

- **Other (Personalization)**
  - if medication not((Formulary?)) then flag-o, group(alt)
  - if (age>=75) and (medication BeersCriteria?) then flag-o, group(alt)
  - if (isNewest medication NursHomeissue?) then flag-o, group(alt)

- **Approval**
  - if medication not(Reconciled?) then prompt-missing, repeat
  - if issue not? then return-list, notes
  - if issue Exists? then return-list, flags, notes

Figure 5B   System logic for the CDSS
Figure 6A  The screen that provides detailed selection of the indications and effects.

Figure 6B  The screen that shows the note entry.
Figure 7. Knowledge representation model in the open source KB tool.

Figure 8. Knowledge representation model in the open source KB tool.
Figure 9  Physical design of the medication reconciliation CDSS.
Appendix B. Prototype demo

This prototype shows that our CDSS provides a complete view of the medication reconciliation list in a familiar excel-like way, making it easier to train physicians on its use. The initial sort order makes it easy to identify Admission Medications that may have been stopped or are still active as well as other currently active medications. A simple, efficient view automatically combines the two lists to be reconciled.

If the physician chooses to have the system provide additional evaluation, pressing evaluate reveals potential issues for consideration. Flags indicate the type of problem: drug interaction, something specific to the patient’s condition or environment, or therapeutic duplication. Medications associated with a particular issue are grouped together by a red box making it easy to see the number and type of issues detected.

A red boxed area can be expanded to reveal additional details about the effects associated with each medication and the indications associated with that problem. When there are duplicate therapies, these details help a physician decide which medication to continue and which to stop. The relevant effects and indications for this patient’s particular situation can be easily recorded by checking those that apply.
The physician has the opportunity to add a quick note explaining the assessment. This information can travel forward with the medication data to offer future explanation. It can also be harvested by system developers and analyzed to support decisions about ways to improve and evolve the system.

This example demonstrates that a physician has the ability to approve a reconciled medication list that includes alerts. The alerts do not prevent finalizing of the reconciliation.

When a physician overrides an alert, the system produces a message explaining that the alert message and the physician’s notes about the issue will be included in the Discharge Summary.

This demonstrates a guiding principle of the design. Our CDSS functions to support decision making, but keeps the decision-making control in the hands of the physician.