

Design and Use of a Joint Order Vocabulary Knowledge Representation Tier in a Multi-tier CPOE Architecture

Donald W. Rucker, MD, MS, MBA¹; Andrew W. Steele, MD, MPH, MSc²; Ivor S. Douglas, MD²; Carmela A. Couderc¹; Gary G. Hardel, PhD¹

1. Siemens Medical Solutions USA, Malvern, PA
2. Denver Health Medical Center, Denver, CO

Abstract:

Two major barriers to adoption of computerized physician order entry (CPOE) systems are the initial physician effort to learn the system and ongoing time costs to use the system. These barriers stem from the CPOE system's need to reformulate physician orders into services that can be electronically communicated to ancillary clinical systems such as pharmacy, nursing, lab or radiology as well as to billing systems. Typical CPOE systems use significant custom user interface programming to match the terms used by physicians to order services as well as the aggregation of those orders into order sets with the underlying orderable services. We describe the design and implementation of a commercial CPOE system that has a formal separate intermediate mapping layer to match physician screen vocabulary and ordering behaviors to underlying services, both individually and in groups, supported by powerful search tools.

Introduction:

Despite heavily publicized data about improvements in patient safety that result from use of computerized physician order entry (CPOE) systems, adoption has been limited. Barriers to use of CPOE are the physician effort to learn the system and the ongoing real or perceived time costs in comparison to writing orders on a paper.¹ When writing orders in a paper chart, physicians use a highly compressed writing style that allows rapid expression of the physician's order. Each individual order then needs to be communicated to a person such as a nurse or respiratory therapist or to a service such as lab or radiology or pharmacy which then performs that order. While orders directly communicated to a person may be written arbitrarily, most orders eventually need to be electronically formatted in order to be electronically sent to a lab, radiology, pharmacy or other ancillary system.

The challenge of CPOE system design is to bridge this gap between the terse paper order style and the detailed order formalism needed to communicate to ancillary clinical systems or to the hospital's billing system where orders have to be matched to individual

billing codes. The software table that holds these billing codes and that orders need to be matched to is often called a "service master" or "service catalog".

We describe a unified knowledge base for both order sets and system vocabulary that simultaneously addresses the need for physician expressivity and the need for a detailed order representation to directly communicate with target clinical and billing systems.

Background:

Each commercial and academic CPOE system has its own programming tools. These often proprietary programming constructs are not well described in the scientific literature making comparative architectural analysis difficult. Most CPOE research focuses on end-user evaluation.² There are few publications on CPOE architecture design.^{3,4} However from industry experience, academic demonstrations, and specific descriptions of screens and functionality it is possible to infer some general design strategies.

There appear to be two relatively common CPOE knowledge management architecture strategies that are typically blended to build the ordering screens.

The first strategy is to simply populate the orderable services with the list of all services that can be ordered working backwards from the service master and other ancillary system files. Here, a physician looking to order Tylenol™ 650 mg might be presented, directly from the pharmacy system, with dozens of brand and generic versions of various acetaminophen tablets, caplets, spansules, and gel tabs each of which is seemingly identically labeled for the first twenty characters as: "acetaminophen 650 mg ...". Any use of "generic" manufacturers' corporate names to specifically identify the generic products further adds to the confusion for physicians. The multiple clinically equivalent formulations frustrate the physician who simply wants to order "Tylenol 650 mg po q4h prn".

The second architectural strategy is to program a physician oriented custom display of possible orders. The top level screens often start with departmental menus for lab, radiology, and pharmacy as well as for nursing, consultations and specialty services such as

dietary. The resulting sequence of screen views can be seen as a tree data structure with the leaves being individual orders. Here, connecting the displayed order name with the orderable item or service stored in the ancillary systems is done with custom programming in source code.

These two architectures are often blended. There are other modes of presenting order options such as having the user type in a string or partial string and then doing a search against the set of available orders returning those orders that match the string.^{5,6}

These two order representation architectures have fundamental limitations. The approach of simply presenting the underlying orderables described by their service master or ancillary system labels means physicians have to learn additional terms and are frustrated to the extent that these service master terms differ from the words they use when speaking with other clinicians or handwriting on paper order sheets.⁷ The naming conventions in these service master and ancillary vocabularies are often quite different from the names used in everyday practice.

The approach using a custom programmed architecture requires major effort on the part of the enterprise to represent all necessary physician ordering pathways. Each failure to anticipate and pre-program an orderable service may lead to frustration on the part of the ordering clinician if they have to search for the “missing” service at the time they are simultaneously trying to deliver clinical care. The ensuing effort to enter the “missing” order typically draws in other physicians, nurses, unit clerks and ancillary staff and is often escalated to hospital administration before finally being presented to the IT team as a repair task. Such repetitive and often heavily publicized failure modes have a corrosive effect on the entire CPOE implementation.⁸

System Design:

Modern enterprise software uses a layered architecture to optimize run-time performance and maintenance and is referred to as “n-tiered”. Typical layers include a screen display tier, a “business logic” tier, and an underlying persistent storage tier using a relational database.⁹ Options for designing the overall architecture of these systems have been called “design patterns” in the classic description of this by Gamma, Helm, Johnson and Vlissides.¹⁰

Here we describe a hospital information system “design pattern” that combines CPOE order set content, CPOE screen behavior and the system’s vocabulary. This design pattern separates the orders knowledge which is typically embedded in a combination of the screen display tier and an

underlying service master tier into a distinctly separate and independent tier. The system represents orders in the same parameterized way that it represents vocabulary.

This design pattern was used to re-engineer a commercial CPOE system. The goal was to increase the usability for multiple CPOE tasks by building and manipulating an independent intermediate representation of the orders knowledge base. Using an independent knowledge representation allows both corporate developers and interested customers (who have a broad range of robust tools to do programming modifications themselves) to isolate screen design from content management and also to isolate the content representation from the underlying service master and ancillary system based representations of the orders. The system integrates the representation of orders, order sets, and order behaviors with the component vocabulary terms using a set of tools originally designed to handle the vocabulary as a term dictionary.

This CPOE design pattern can achieve a number of valuable sub-goals in the construction, use and maintenance of orders and order sets:

Orders and Order Set Construction

- 1) Support an independent partitioning of orderable services based on their frequencies of use and or the users’ mental models of how they place orders.¹¹ One can use this to partition and automatically display tabbed pages for labs, radiology, meds, and nursing orders. Within the lab section one can have a list with the most common labs overall and then individual common lists for the sub-departments within lab such as chemistry, hematology, and microbiology. The most commonly ordered items require the fewest mouse clicks. This follows what is colloquially known as the “80-20 rule”. This minimizes mouse clicks as well as overall reading, thinking and navigating. These lists of commonly ordered items can be made specialty specific so that the specialty of the physician (internist, urologist, pediatrician etc.) logging on to the ordering session determines what the physician sees.
- 2) Support nested sub-department lists such as lists limited to CT scans, MRI studies, ultrasounds or similar groupings within a department.
- 3) Support selective reuse of clinical content such as underlying order detail sets describing frequency, duration, priority, and indication can be individually or collectively reused allowing the users to see the same screen appearance each time.
- 4) Support the incorporation of closely related order set behavior such as whether an order is default “on or off” in an order set display or whether a text

banner is displayed within an order set and what that banner reads. Other supported behaviors include being able to set limits for the minimum and maximum items selected within an order set. A common use of this feature is to mandate the selection of one antibiotic from a list of institutionally preferred antibiotics (such as picking one of three preferred antibiotics listed in a pneumonia order set) without use of custom code.

5) Support the construction of the compound order strings such as “amoxicillin 250 mg po q8h” using the same underlying knowledge representation and incorporating specific order set behaviors rather than through hard coding as text.

Orders Use and Search

- 6) Support search at both a global level and a department or sub-department level. Thus one can search for an item among all orders, just those in lab, or just those in hematology. The ability to constrain search without asking separate questions of the users such as is done in the advanced search features of Google.com or Pubmed.gov search engines, means users are returned a shorter list. For example, typing the string “thyr” against a global order search will return various thyroid medications and thyroid tests while typing that same string against a labs only search on a labs tab screen will return only thyroid tests and not thyroid medications.
- 7) Allow multiple different search tools to be applied to this integrated representation of the knowledge base. For example one can search by substring anywhere in the order so the user need not distinguish between “CT Head Non Contrast” and “Head CT Non Contrast” when searching by the anatomical term “head”.
- 8) Can allow the simultaneous display on one screen of multiple order search and navigation strategies.
- 9) Support any number of synonyms.

Maintenance

- 10) Increase maintainability of the content – now content is maintainable independently of the code for screen designs and clinical workflows. By separating the development environment from the base hospital information system, order set content can be uncoupled from other systems issues.
- 11) Support enterprise knowledge management by externalizing the order content enterprise-wide as well as locally.
- 12) Support mapping of order names familiar to physicians to items in a service master or other ancillary systems hiding the service master names from the end user.
- 13) Support mapping of orders to external reference terminologies such as SNOMED or LOINC.

To summarize, having an independent middleware layer for the representation of orders and ordering behavior provides a powerful tool to support and maintain multiple strategies which increase physician ease of use with CPOE. Combining the order representation with the vocabulary representation allows the use of the same search tools for both.

This is analogous to an “n-tiered” business architecture with formal separation of business logic.

System Description:

This product uses the core order processing of the Siemens Invision™ hospital information system. This system runs on an IBM mainframe in both an application service provider (ASP) and an on-site mode. Screen displays are via a thin-client browser. The browser screen displays and the underlying clinical architecture each have families of flexible programming tools allowing customers to adapt the system to their own needs.

The intermediate layer modeling the vocabulary, orders, order sets, and associated behaviors is defined and stored in an IBM DB2 relational database. The intermediate layer, which is commercially named the “Common Vocabulary Engine™” (CVE) runs as a separate process linked to the Invision order processing through a server which has precompiled indices of all terms and their relationships. The intermediate layer also serves as a term dictionary and synonym map for the Invision clinical system. Use of data dictionaries to do term mapping in clinical systems is well-described.^{12, 13}

Internally, the system represents orders as a hierarchically nested series of terms and accompanying individual defined parameter sets. The nesting of the terms provides order set behavior. The grouping of terms into sets drives screen displays. Lower level behaviors such as whether orders are defaulted to be on or off in an order set, medication routes and allowed doses are defined by parameters and parameter sets. The system provides users flexibility to arbitrarily nest terms and combine parameters. Sets of parameters supporting orders are grouped in tables and make reference to a base table of all available terms. The system therefore supports but does not enforce ontologies.

Six relational tables form the core of the database structure: Term, Term Type, Term Synonym, External References, Set Definition, and Support Term. A given Term may have one or more associated Term Type tables with associated Synonym and External Reference tables. Terms themselves may be grouped into sets using Set Definition tables that support various rules for term

uniqueness and explosion characteristics. The Support Term provides a mechanism that allows users to establish relationships between terms (both individual and sets).

Parameter are simply terms that defines a piece of data that may be collected for an order and associated information about that data (allowable values, suggested values, defaults, etc.). Typical parameters that might be defined for an orderable drug are Frequency, Dose, and Priority, but the system allows complete flexibility for users to define parameters for any data they wish to collect. Individual parameters are then grouped into sets that are applied to a specific orderable service or a class of services with appropriate defaults to complete the definition required to enter an orderable service.

There are separate utility programs to maintain referential integrity. A term load tool helps load data from reference libraries customers may wish to access such as SNOMED, LOINC or a pharmacy database such as from First Data Bank™.

A separate index server allows the system to achieve the fast performance necessary in a CPOE system.

Here, we describe a relational database architecture for achieving this separation. Other formats of data storage and mapping such as XML may be able to achieve a similar result. Other approaches to re-architecting order behavior are possible and Miller and Geissbuhler describe an earlier approach using the same base Invision™ system.³

Implementation Status:

The first implementation of this approach to designing a CPOE system has been in the 25 bed Medical Intensive Care Unit at the 349 bed acute care hospital of Denver Health, a nonprofit integrated public safety net system serving the needs of Denver, Colorado and surrounding areas. Denver Health runs the underlying Siemens Invision system remotely as an ASP customer.

The system using the intermediate layer common vocabulary engine described is currently live in the MICU at Denver Health. Approximately 6,000 orders are entered on a weekly basis, and over 400 physicians and nurses have been trained.

Orders and Order Set Construction

At Denver Health, clinical content related to order detail is reused. For instance, for compliance with JCAHO standards related to abbreviations, frequency details for daily medicines were changed from “QD” to “QDAY” once and this change was automatically propagated to all orders defined with this value. Representation of orders has been closely aligned to

physician desires. The order entry screens have tabs for nursing, diagnosis and therapy, and medications with subcategories such as chemistry, hematology, and microbiology for laboratory.

Order set development has been facilitated by the ability to nest subordinate order sets within larger order sets. For instance, the order set for deep venous thrombosis (DVT) is used both as a stand alone order set and also incorporated into other order sets. Denver Health has constructed approximately 20 orders sets for the MICU using this technology. Many of these are being reused as other departments start CPOE. To help compliance with evidence based care, orders within the order sets can be defaulted to be “on” so that users have to deselect the order to not have it be entered. As an example, “advanced care directives” is defaulted “on” so the physician needs to address this order for all patients during the admission order process.

Orders Use and Search

The initial installation confirmed that when orders were represented in the knowledge base using only one set of local terms, there were a number of instances when users were not able to find the order. For example, when a user searched for Hepatitis B e antigen by typing “hepa” no terms were returned by the search. Investigation of this event revealed that many of the hepatitis assays were only listed in the lab system with an abbreviation starting with “hep”.

Analysis of failed searches revealed that the most of the “failed searches” were within the categories of either nursing services or tests. This is not surprising as individual tests may have multiple names. For example, different hospitals use the terms “chem panel”, “SMA 7”, “lytes”, “basic metabolic panel” or similar to refer to the same lab panel. The need for synonyms is more limited with medications because medications are commonly known by either their generic name or a single brand name.

After analyzing initial user satisfaction data, Denver Health embarked on a one-time process of loading common laboratory, and radiology synonyms into the intermediate layer vocabulary engine.

Approximately 90 laboratory and 90 radiology synonyms were loaded. This represented about 5% of the orders for those departments. Although some of the synonyms were recommended by physicians and nurses it was more valuable to talk to laboratory and radiology department staff since they had more experience answering questions about the names of specific studies and procedures. In discussions with support analysts, it is clear that complaints of “I can’t find....” were significantly reduced.

Utilizing the CVE has also provided enhanced searching with ability to do substring searching. This is useful when looking for groups of orders. For instance searching for "culture" will return all possible types of cultures allowing the user to select multiple related orders from one screen compared to individually searching for each item.

Maintenance

Denver Health has two analysts specifically dedicated to development and maintenance of the CVE. Many large order sets have two to three hundred total orders with 7-10 embedded sub-order sets. These sub order sets address items such as nursing practices, respiratory treatment standards, and prophylactic measures such as deep vein thrombosis, gastrointestinal bleeding, and ventilator associated pneumonia. Changes to a sub-order set can be automatically propagated to any order sets that contain that sub order set.

Discussion:

We describe a design pattern and tool to extend a commercial hospital information system with pre-existing order processing support to better address CPOE learning and speed. We describe an independent representation of orders and vocabulary and associated screen approaches which present multiple simultaneous ordering strategies. Physicians use multiple strategies when ordering today (pre-printed orders, superbills, quicklists, lab panels, etc.) and this system facilitates electronic analogs.

Using an intermediate order design pattern layer for CPOE provides efficiencies in building complex orders and order sets for the IT staff, displays orders in a more physician friendly manner, provides for enhanced display and searching, and provides a method to facilitate best practice order sets.

There are some limitations. An intermediate layer, may add complexity to the CPOE application. Processes need to be clearly defined for updating tables and databases, ensuring standardization with naming conventions, and ongoing quality assurance. To leverage this approach, application analysts need specific training. There is a need for the application analysts to work closely with the physicians to better understand their mental models of ordering behavior.

This is an early report. The authors plan further research to measure usability and time performance.

While this system is designed to work with one vendor hospital information system, either the system itself or the overall approach used may be extensible to other systems. Each CPOE system has its own knowledge representation strategy. CPOE literature tends to focus on CPOE evaluation rather than the

details of CPOE system architecture making a formal comparison to other systems difficult.

Improved order knowledge representation tools appear to be important for increasing the voluntary adoption of CPOE and its documented safety and healthcare performance benefits.

References:

- [1.] Ash JS, Stavri PZ, Kuperman GJ. A consensus statement on considerations for a successful CPOE implementation. *J Am Med Inform Assoc.* 2003 May-Jun;10(3):229-34.
- [2.] Kuperman GJ, Gibson RF. Computer physician order entry: benefits, costs, and issues. *Ann Intern Med.* 2003 Aug 1;139(1):31-9.
- [3.] Geissbuhler A, Miller RA. A new approach to the implementation of direct care-provider order entry. *Proc AMIA Annu Fall Symp.* 1996:689-93.
- [4.] Payne TH, Hoey PJ, Nichol P, et al. Preparation and use of preconstructed orders, order sets, and order menus in a computerized provider order entry system. *J Am Med Inform Assoc.* 2003 Jul-Aug;10(4):322-9.
- [5.] Lovis C, Payne TH. Extending the VA CPRS electronic patient record order entry system using natural language processing techniques. *Proc AMIA Symp.* 2000:517-21.
- [6.] McDonald CJ, Overhage JM, Tierney WM, et al. The Regenstrief Medical Record System: a quarter century experience. *Int J Med Inform.* 1999 Jul;54(3):225-53.
- [7.] Berger RG, Kichak JP. Computerized physician order entry: helpful or harmful? *J Am Med Inform Assoc.* 2004 Mar-Apr;11(2):100-3.
- [8.] Han YY, Carcillo JA, Venkataraman ST, et al. Unexpected increased mortality after implementation of a commercially sold computerized physician order entry system. *Pediatrics.* 2005 Dec;116(6):1506-12.
- [9.] Fowler M. *Patterns of Enterprise Architecture.* Addison Wesley; 2003.
- [10.] Gamma E, Helm R, Johnson R, et al. *Design Patterns.* Addison Wesley; 1995.
- [11.] Horsky J, Kaufman DR, Oppenheim MI, et al. A framework for analyzing the cognitive complexity of computer-assisted clinical ordering. *J Biomed Inform.* 2003 Feb-Apr;36(1-2):4-22.
- [12.] Linnarsson R, Wigertz O. The data dictionary--a controlled vocabulary for integrating clinical databases and medical knowledge bases. *Methods Inf Med.* 1989 Apr;28(2):78-85.
- [13.] Huff SM, Rocha RA, Coyle JF, et al. Integrating detailed clinical models into application development tools. *Medinfo.* 2004;2004:1058-62.