The State of the Art in the Reduction of Medical Errors

Danny KOPEC [a,b], Suzanne TAMANG [a] , Karen LEVY [b,] , Ronald ECKHARDT [b] , Gene SHAGAS [a]
[a] *The Graduate Center, The City University of New York*
[b] *Brooklyn College, The City University of New York*

Abstract. The IOM report, *To Err is Human, Building a Better Health System*, galvanized public and political attention to the prevalence of medical errors in the Unites States. The IOM set a clear goal, "given the current knowledge about the magnitude of the problem, the committee believes it would be irresponsible to expect anything less than a 50 percent reduction over five years." As part of the IOM's four-part strategy was a recommendation that error reporting systems be established. No one denies that errors that occur in medicine can not be reduced if they cannot be defined. To achieve this goal of reducing errors, we have established a definition of a "medical error", described the current taxonomies that have been created over the last five years for their classification, and suggested a conceptual model for designing and testing a medical error reporting system. A system that facilitates identification, relies on health professionals and electronic repositories of clinical information to report events, and tracks and monitors medical errors, reliably, efficiently, and accurately is the objective of our design. Our next step is to implement, test, and evaluate this system based on our research.

Keywords. Medical error, error reporting systems, systems analysis, expert systems

Introduction

The term "error," especially if attributed to a human, tends to connote blame or responsibility. Many theories have been espoused, but experts have yet to reach a consensus on the definition of an error. Hence, a task that is important to researchers is the development of a more refined and universally accepted terminology for discussing human medical errors.

Senders and Moray [1] maintain that the definition of an error depends on the point of view of the person who judges that an error has occurred. "The actor who commits an error recognizes it only after the fact, with the perspective provided by hindsight, and either an actor or an external judge needs a model of task performance in order decide whether an action has been correctly executed." They specify that an "error occurs when a planned series of actions fails to achieve its desired outcome, and when this failure cannot be attributed to the intervention of some chance occurrence." For instance, a chess expert sees things that the novice cannot see. The novice, therefore, does not do things that an expert would do. Since the novice cannot even *conceive* of the expert's move, it may not be entirely correct to say that he or she made an error. It seems more reasonable to say that where there is no possibility of correct performance there can be no error, even though the performance may be imperfect [1].

Senders and Moray also distinguish between *errors* and *human errors*. "An *error* is any significant deviation from expectation, depending on statistical criteria or experience of normal performance standards whereas *human error* is a deviation from expected human performance." This distinction concerns whether one examines the actor's behavior in isolation, or the performance of the human-machine system as a whole [1]. Senders and Moray also suggest that there are important philosophical issues that should not be overlooked. Some individuals believe that errors do not exist, others consider them as effects and not causes. Still others believe that only certain kinds of human acts can ever be in error, since for many acts, a good reason can be given, even if to the observer the action may appear to be incorrect. What is agreed is that all errors imply a deviation from intention, expectation or desirability. Errors can be perceived as psychological mechanisms, as sensory or perceptual events, cognitive events, motor events, actions in well-defined system (expression of error) or unacceptable consequences in the output of the controlled system (consequence of error) [1, 2].

Reason's definition of human error is one of the most widely accepted: "an error is a failure of a planned sequence of mental or physical activities to achieve its intended outcome when these failures cannot be attributed to chance" [2]. Reason proposed a framework, which classifies human performance into three basic levels: Skill-based, Rule-based and Knowledge-based. Figure 1 illustrates these three performance levels and their link to human perception.

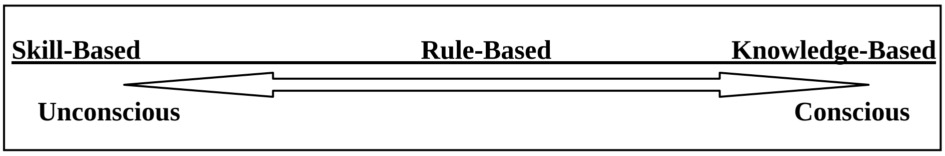

Figure 1. Reason's Three Human Error Performance Levels and their Link to Human Perception

A *slip* or "an unconscious glitch in automatic activity" is a skill-based error. A slip results in an incorrect execution of a correct action sequence. An example of a slip would be a physician who chooses an appropriate medication but writes 10 mg when the intention was to write 1 mg. The intended dose was correct, but the action sequence (writing the correct dose) was incorrect. A *lapse* is a rule-based error and like a slip, is an error of execution, with the difference being that a slip is observable, and a lapse (for example, not being able to recall something from memory) is not. A *mistake* is a knowledge-based error whereby an action proceeds as planned but fails to achieve the intended outcome because the planned action was incorrect. Mistakes typically occur from a lack of or misapplication of knowledge [1-3].

As seen from the above discussion, there is no universally recognized terminology for *error*, *human error* or even *medical error*. However, for the purposes of our research, we have adopted the definition outlined in the IOM report, which defines a *medical error* as "the failure of a planned action to be complete as intended or the use of the wrong plan to achieve an aim" [3]. It is also important to note that an error is not defined by an adverse or serious event. An adverse event may occur with no error if the intention was the proper one, the action was properly executed, and the outcome was probabilistic in nature (as in administering a medication or performing an operation known to be risky). Thus, an adverse event is defined as “an injury caused by medical management rather than the underlying condition of the patient” [3].

2. Error Classification Schemes

Reason differentiates between active errors and latent errors. Active errors are errors that are generally readily apparent (for example, pushing an incorrect button or ignoring a warning light) and almost always involve an operator at the frontline. Latent errors (or latent conditions), in contrast, refer to less apparent failures of organization or design that contributed to the occurrence of errors, or allowed them to cause harm to patients [2].

Active errors are sometimes referred to as errors at the *sharp end*; errors at the sharp end are noticed first because the person closest to the patient commits them. Examples would be an orthopedist who operates on the wrong leg, or a nurse programming an intravenous pump incorrectly or when a health care professional performs any aspect of direct care with some error. Latent errors are referred to as *blunt end* errors; this term refers to the many layers of the health care system removed from the direct control of the health care professional. These layers include poor system design, inadequate protocols/procedures, incorrect installation of software or equipment, bad management decisions and poorly structured organizations [2].

3. Existing Medical Error Taxonomies

One of the problems with classifying medical errors is that there are many ways of accomplishing this task. For example, one can focus on processes such as diagnosis or on some underlying system failures. Also, one can classify errors in terms of the types of disease, drug or procedure that is most commonly associated with the error, or in terms of outcome.

Another issue is that the nomenclature for defining medical errors in health care is not standardized, meaning that there is no single internationally recognized taxonomy for defining and classifying medical errors. The IOM made the first attempt to clarify definitions [3]. Current taxonomies are very diverse and vary widely in scope and aim. Of the available taxonomies used to classify medical errors, some include lists of descriptive words organized into several domains, while others use hierarchies based on the developers’ approach to the understanding of errors [4].

Furthermore, the approach to codification varies widely among different groups studying medical errors. For example, the New York Patient Occurrence and Tracking System (NYPORTS) and early users of the American Academy of Family Physicians (AAFP)-Linnaeus Primary Care Patient Safety Taxonomy, primarily applied a single global code to an error report; the Applied Strategies for Improving Patient Safety (ASIPS) and Medical Errors Reporting System (MERS) coding schemes label each event with multiple relevant codes. The resulting description of medical errors may vary, based on the coding approach as well as the codification schema. Additionally, most taxonomies are conceptually based and have not been evaluated to determine their utility in furthering the understanding of the processes involved in errors [4].

An important exception to the conceptually derived taxonomies is the AAFP-Linnaeus taxonomy developed by Dovey *et al.* The AAFP-Linnaeus taxonomy uses an iterative, qualitative analysis of medical error reports to develop a hierarchical taxonomy that describes error processes in primary care. The qualitative approach identifies themes found in the available data, instead of imposing on the data concepts and constraints based on preconceived ideas. The AAFP-Linnaeus taxonomy is grouped into two major sections: process errors and knowledge/skills errors. The codes provide detailed descriptions of actual events and appear to be easy to use for anyone with clinical knowledge. However, potential issues with the current version include its inability to easily separate the event processes from the participants, the lack of information on who discovered the event, and the mixing of process and causation codes. The inability to clearly delineate causation may limit the ability of the AAFP-Linnaeus system to promote the development of interventions designed to improve care and decrease errors. It is also unclear if the taxonomy can highlight similar process errors across different clinical activities or among the array of error processes within a given clinical area. The qualitative approach used to develop the AAFP-Linnaeus taxonomy has strengths, and the themes and constructs identified should be included as ambulatory patient safety taxonomies are further developed [4].

Other taxonomies, although primarily developed for inpatient use, have been applied in the ambulatory setting. Given the major differences in the care processes between inpatient and outpatient settings, it is unclear whether descriptions of the breakdowns in care in one setting can be applied to another. Taxonomies designed to evaluate specific domains of errors, such as the National Coordinating Council for Medication Error Reporting and Prevention (NCC MERP), may apply across care settings, but are designed to specifically describe one aspect of the medical error universe, that of medication errors. Moreover, not all error taxonomies have been made available for public scrutiny, thereby hindering continued advancement of the discipline [4].

3.1 Dimensions of Medical Outcomes (DMO)

Dr. Michael Victoroff began working on a conceptual based taxonomy to code medical errors and malpractice reports in the mid-1990s. This system, called Dimensions of Medical Outcomes (DMO), had undergone several revisions and refinements when it was used under agreement with Dr. Victoroff and COPIC, Inc. to code medical error reports for a large AHRQ sponsored research project focused on primary care offices, called ASIPS. The ASIPS research group found that they needed to add some codes to the existing taxonomy to better describe the reports they received, and they also felt that the DMO taxonomy did not demonstrate parallel construction in some areas. This group developed a modified version of the DMO that addressed some but not all of the issues discovered during this research project [4].

3.2 American Academy of Family Physicians (AAFP)-Linnaeus Primary Care Patient Safety Taxonomy

The American Academy of Family Physicians and others have, over several years, been developing a taxonomy, now called the Linnaeus taxonomy based on a qualitative analysis of several hundred error reports collected from the United States initially, and then from six other countries. AAFP/Linnaeus is derived from (mainly) physicians' reports of mistakes they observe in daily practice [4].

3.3 Joint Commission on Accreditation of Hospitals and Healthcare Organizations (JCAHO): Patient Safety Event Taxonomy

The JCAHO Patient Safety Event Taxonomy has four primary classification domains: impact, type, domain, cause, and prevention and mitigation. This taxonomy is based collectively on mechanisms, processes, and outcomes that underlie the failures in structure, process and human behavior. This taxonomy is an effort to develop a codification framework for the description of medical errors that is not domain specific. The taxonomy has been developed with input from a number of medical error "experts," and with consideration of a number of existing taxonomic systems. This taxonomy starts with an outcome and works backwards to the reason(s) for the outcome, whereas the other taxonomies presented start from a mistake and work in both directions - forwards to an outcome and backwards to a cause [4].

3.4 National Coordinating Council for Medication Error Reporting and Prevention (NCC MERP)

Central to the development of medical error reporting is a need for a controlled vocabulary and taxonomy. In response to reducing medication errors, the NCC MERP released a medication taxonomy in 1999. The NCC MERP Error Reporting System is a publicly available taxonomy focused on drug errors and adverse events. Development included input from an international group interested in medication safety. The NCC MERP does not charge for use of the taxonomy which is available for download from the World Wide Web. The NCC MERP medication error taxonomy is organized into eight major categories [4, 5]:

1. Patient Information
2. Medication Error Event
3. Patient Outcome
4. Product Information
5. Personnel Involved
6. Type of Medication Error
7. Causes
8. Contributing Factors

Our research suggests that this taxonomy is particularly strong in the area of physical harm associated with a medical error, and it is one of the bases for the extension of the taxonomy recommended by Kopec *et al.* [6] (see Figures 2 and 3).

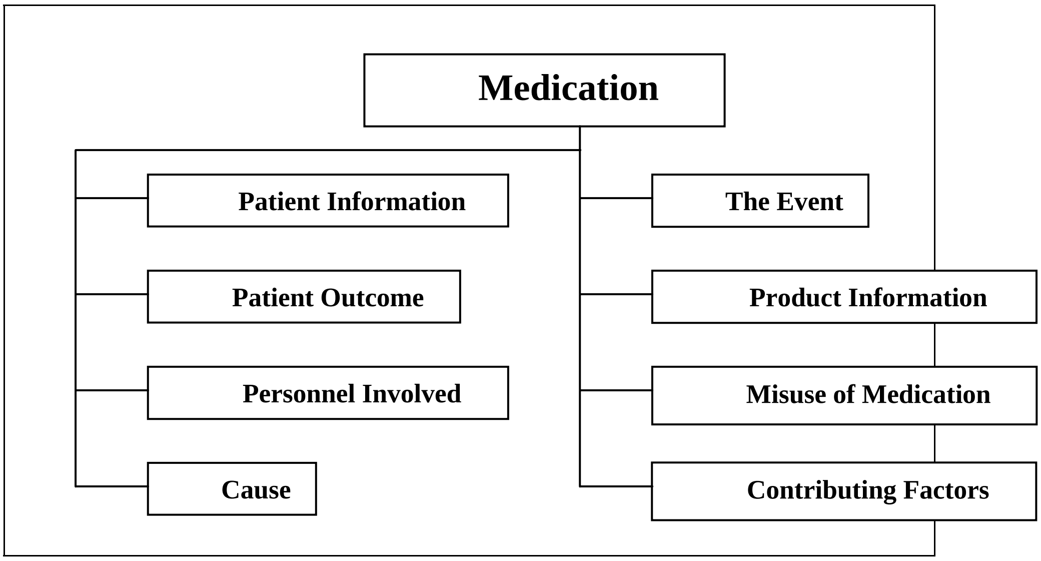

Figure 2. Classification of Medication Errors based on NCC MERP Taxonomy [5].

The final taxonomy recommended by Kopec *et al.*, [6] presented below in Figure 3, is a combination of the IOM, JCAHO and NCC MERP classifications.

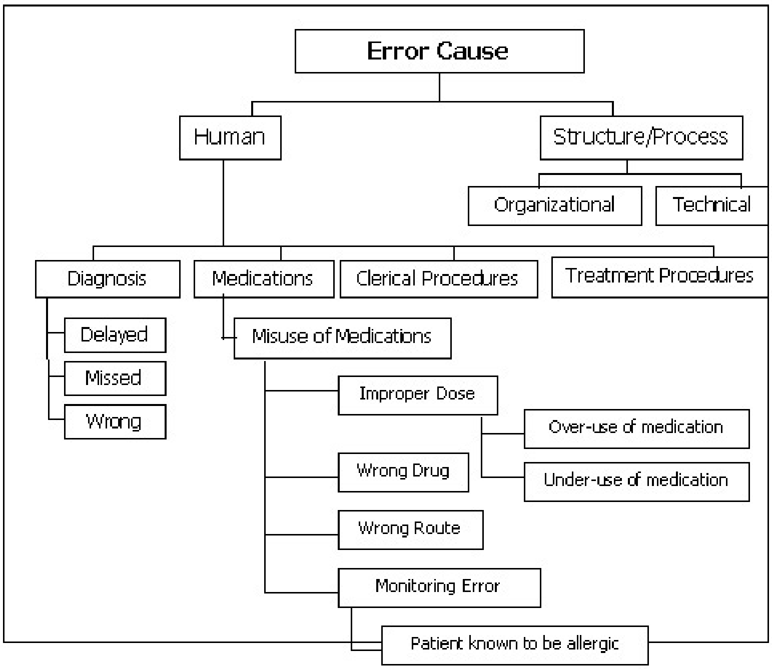

Figure 3. Taxonomy of Medical Errors Recommended by Kopec *et al.* [6,7]

3.5 Harvard Risk Management Program

This taxonomy is more a set of domains with lists of second level elements, utilized to analyze events reported to hospital risk management organizations. It is used by a number of private hospitals in the New England area, as well by a number of Academic Health Centers across the United States [4].

3.6 The Australian Incident Monitoring System General Occurrence Classification (AIMS GOC)

The AIMS GOC was designed to capture "things that go wrong" throughout the healthcare system after the AIMS researchers found that existing classifications such as the Read Codes or ICD-9 E Codes were insufficient for their purposes. It has three major categories: contributing factors and hazards, descriptors of the incident, and outcomes and consequences. The GOC is not strictly speaking a taxonomy, but rather a database that organizes salient and important elements of an incident report in a way that preserves the narrative description, yet allows complex analyses of relationships among many variables. It is not possible to display this complex database on paper, only segments of the trees. AIMS is perhaps unique in that it is designed to receive reports from a wide variety of sources, including incident monitoring, medical record review, death certificates, hospital discharges, surveys of general practice, patient complaints, medico-legal investigations, coroner investigations, results of other enquires and investigators and even literature searches [4].

3.7 Medical Error Reporting System-Total Healthcare (MERS-TH)

The Medical Event Reporting System – Transfusion Medicine (MERS-TM), uses a technique known as root cause analysis to uncover the underlying factors, circumstances, and decisions that contributed to the event in question. The outcome of the analysis is represented as a causal tree. A tree provides a visual representation or diagram (such as a fishbone diagram) of the event that includes all possible causes (and recoveries) gathered during the investigation of the event. The investigator gathers information from individuals involved in the event by repeatedly asking why. This process elicits data at multiple levels and defines the actions and decisions leading up to the event. The final product of the root cause analysis is a set of root causes that are described, coded, and ultimately entered into the database [4].

The MERS-TH system was initially developed to describe and modify errors involved in transfusion medicine. MERS-TH is a root cause analysis system for adverse medical events and is an extension of the MERS-TM. It has now been expanded to a wider setting, though is still primarily focused on hospital care. The system adds codes driven by the data collected, that is, when the "non-specific" code within a particular area grows unwieldy. The system is directed by a causal analysis tree and includes a number of analytical tools. It is underpinned by empirical data, and has had extensive conceptual development over many years. Grounded in general safety theory, it was specifically developed to capture reports of adverse events, with the goal of assigning causal codes to these events. It focuses on understanding causation as a first step in understanding and avoiding similar mishaps in the future. It uses the Einthoven Classification model for the Medical Domain, which has three main causal categories: latent errors (technical and organizational), active errors (human), and other (patient-related and unclassifiable). MERS is not strictly speaking a taxonomy, but rather a relational database and tool for causal analyses [4].

It is clear that various medical error taxonomies exist because of the wide range of possibilities that exist for the classification of medical errors. What is notable is that no one system is thought to be intrinsically “better” in incorporating or helping to account for the various possibilities of errors that exist in the multitude of areas in the healthcare system. It is felt that standardization of error terminology will help the health care industry to better study errors, and this in turn will help in a better understanding of the issues at hand.

4. Developing Medical Error Reporting Systems

When contemplating the design, coding and implementation of new, comprehensive health care reporting systems, particularly those which will ultimately allow for medical error reporting, we considered it important to survey not only currently used systems in the United States and other industrialized countries, but also to survey health care reporting systems worldwide.

In terms of the usefulness of data collected, it is many times mandatory to understand worldwide patterns of health and disease and the capabilities of individual countries to respond to health crises. A good example of this need for global monitoring is the current worldwide tracking of H5N1 avian flu virus with different outbreaks in the Far East, Middle East, Asia, Africa, Europe and soon predicted to make is way to the United States and the Americas by migrating wild bird species. By having advanced warning of potential pandemic outbreaks, the world’s medical resources can be marshaled and grouped to respond to outbreaks even in countries with very limited medical infrastructure.

Help in equalizing available resources and providing worldwide monitoring is largely the job of international organizations especially the World Health Organization (WHO, www.who.int) which coordinates with the health systems of individual countries and acts as a clearing house for data from health care agencies throughout the world. For instance, WHO’s Epidemic and Pandemic Alert and Response (EPR) group has as one of its stated missions coordinating the global response to human cases of H5N1 avian influenza and monitoring the corresponding threat of an influenza pandemic. Their webpage provides universally available information to track the evolving situation and provides access to both technical guidelines and information useful for the general public. This group publishes: (1) an Outbreak Verification List (OVL) which reports current outbreaks thought to have a potential for international implication, (2) Disease Outbreak News which provides public information about officially confirmed disease outbreaks of international importance, and (3) the Weekly Epidemiological Record which publishes epidemiological information on cases and outbreaks of diseases under the International Health Regulations (yellow fever, plague, cholera) and also on other communicable diseases of public health importance. Clearly, any newly designed comprehensive health care reporting system should anticipate the availability of this type of data and make provision for its maximum utilization.

One exceptionally important factor to remember that is quite clear when reviewing the health care infrastructures of diverse countries worldwide is the extreme variability in terms of available local resources for health care. For example, there is only ONE hospital based psychiatrist in the Republic of the Congo, a country of over 2.5 million people. In many economically poor countries, there are few if any governmental health care personnel assigned to large geographical areas where much of the population may live. This gives rise to serious concerns when designing comprehensive health care reporting systems as to how needed data will be collected and entered into computerized databases for use in determining global heath care needs and issues. Even simple factors such as the absence of electricity in remote areas act as serious impediments to the collection of vital data for analysis. Innovative new technologies such as the use of battery powered hand-held devices to collect data are currently being explored.

Even though there is great variability country-by-country, the collection of available data from different countries is largely managed by international organizations, mainly WHO. WHO maintains an on-line searchable database library called WHOLIS amongst other statistical global databases including (1) World Health Survey Results, (2) an online World Health Statistics database, (3) an online HIV/AIDS database, (3) a Death and Disability Adjusted Life Years database (estimates for 2002 by cause for WHO Member States) which contains estimates of numbers, crude rates and age-standardized rates as well as information on data sources and levels of evidence, and (4) the International Statistical Classification of Diseases and Health Related Problems, 10th Revision, 2nd Edition, which has become the international standard of diagnostic classification for all general epidemiological and many health management purposes.

WHO also sponsors a very important Health Metrics Network (HMN) which is an innovative global partnership founded on the premise that better health information means better decision making and that means better health for all. HMN states that enhancing the availability, quality, consistency, and use of health data requires greater harmonization among stakeholders around agreed technical standards. HMN joins governments with donor agencies, health planners with statistical experts, and communities with health providers in a shared mission to strengthen the systems needed to generate sound health information.

Part of the mission of the HMN is to promote consistent data collection, organization and its integration in to universally accepted Health Information Systems (HIS). To help achieve these goals, HMN provides a variety of tools such as (1) the HMN Frameworks and Standards for Country HIS Development, (2) the HMN Assessment and Monitoring Tool, (3) standardized Country Logbooks, and (4) a SAUCE Tool (Synthesis, Analysis and Use of Country Evidence) for the use of health information for country-level health policy formulation and planning. The HMN also co-sponsors periodic worldwide meetings on health information systems and other health issues. They award funding to facilitate the development of useful health information systems with the next deadline for submission of Expressions of Interest directed to trying to achieve international consensus on ways to measure and monitor health care systems scheduled for April 2006 at the time of this writing.

When one considers development of new comprehensive health care reporting systems, it would be wise to ensure that they are compatible with models of health care systems proposed by global institutions. Because of the lack of health care infrastructure in many poor countries, medical error reporting is probably not going to be available for quite some time on a global basis. However, any new comprehensive health care reporting system can be designed to include this feature with data entered for developing countries as they later acquire a capacity for medical error reporting. A key document specifying issues important to the international community can be found online at: http://www.who.int/healthmetrics/library/issue_1_05apr.doc. The *Bulletin of* *the World Health Organization*, Volume 83, Number 8, August 2005, 561-640, was centered around the special central theme of Health Information Systems with numerous editorials, news and research articles discussing the needs of the global community and an online version may be found at: http://www.who.int/entity/bulletin/volumes/83/8/en/index.html.

5. Testing Methodology

Erik Hatcher wrote, "one of the governing principles of Extreme Programming (XP) is that programmers should perform regular unit testing and should be continuously integrating the changes into production-like environments. Furthermore, XP suggests that this process be automated whenever possible. After all, if developers are to create test cases as eagerly as they do production code, the process has to be relatively painless" [10]. In response to this request, he modified the popular Ant 1.3 and the JUnit test framework for complete, customized automation of the build and test process.

JUnit can be described as a simple, open source framework to write and run repeatable tests. It is an instance of the xUnit architecture for unit testing frameworks. JUnit was originally written by Erich Gamma and Kent Beck [8]. JUnit is Open Source Software, released under IBM's Common Public License Version 0.5 and hosted on SourceForge [14].

JUnit features include:

- Assertions for testing expected results
- Test fixtures for sharing common test data
- Test runners for running tests[8]

When developing or deploying an application, important features of JUnit framework are options to perform manual and automatic testing.

The Eclipse Platform subproject provides the core frameworks and services upon which all plug-in extensions are created. It also provides the runtime in which plug-ins are loaded, integrated, and executed [13]. The Eclipse platform can deal with any type of resource (Java files, C files, Word files, HTML files, JSP files, etc) in a generic [13]. This is makes it easier to share information between constituents. We evaluated JUnit tool for our Java application, created using the Eclipse IDE.

To test server-side classes, we considered two viable approaches: *mock objects*, which test classes by simulating the server container, and *in-container testing*, which tests classes running in the actual server container [15]. *StrutsTestCase* for JUnit allows us to use either approach. *StrutsTestCase* for JUnit provides two base classes, both of which are extensions of the standard JUnit *TestCase*. *CactusStrutsTestCase* class uses the Cactus testing framework to test Struts classes in the actual server container. We will utilize class *MockStrutsTestCase*, which uses a set of *HttpServlet* mock objects to simulate the container environment without requiring a running servlet engine [15]. *MockStrutsTestCase* provides methods that set up the request path, request parameters for *ActionForm* subclasses, as well as methods that can verify that the correct *ActionForward* was used, and that the proper *ActionError* messages were supplied [11].

Many books have already been written about automated testing, but very few of them pay attention to the question of how to organize such tests. As more tests are written, it becomes harder to know where to put a test or even what to call it. This has become a “significant issue with the rise of test-driven development (TDD), which has been popularized by Extreme Programming (XP)” [13].

5.1 Automating Unit Testing with JUnit

To automate testing, you need a testing framework. We selected JUnit for several reasons [9]:

- We do not have to write our own framework
- It is open source
- It is widely accepted among developers
- There are examples to model new applications from
- It allows a programmer to separate test code from product code
- It easily integrates into the build process

5.2 Integrating Testing into the Build Process with Ant

Ant tool is becoming the *de facto* standard in the open-source world. The reason is simple: Ant is written in the Java language, which allows the build process to work on multiple platforms. Ant features include:

- Class extensibility Java classes are used to extend build features instead of using shell-based commands.
- Open source Because Ant is open source, class extension examples are plentiful.

XML configurable Ant goes beyond just being Java based. Ant uses an XML file for configuration of the build process. Given that builds are hierarchical in nature, using “XML to describe the make process” is logical [9].

6. Future Directions

Our future research will concentrate of the design and development of a medical error reporting system, acknowledging the current taxonomies described. This framework will be flexible, scalable and interoperable across platforms. The data definition will be based on internationally accepted standards, to ensure that they are compatible with models of health care systems proposed by global institutions.

This medical error reporting system will be designed to incorporate explicit variables that can be found in medical records and are related to adverse events. However, a recent study at Columbia University concluded that a considerable amount of medical errors that occur are underreported [16]. For example, the New York State health department’s mandatory event reporting program [17] estimates that 16% of code 605 events (death within 48 hours of an operating room procedure) are being reported. In the *Journal of Hospital Pharmacy*, a study indicated that detecting adverse drug events form voluntary reporting had a lower sensitivity than manual review of laboratory reports and pharmacist screening of medication orders [18].

To surpass the limitations posed by medical error self-reporting, our system will be based on both voluntary reporting and keyword searching of electronic clinical data. With the current movement to the automated collection of clinical information, this design is an improved approach. The longitudinal EMR is a repository of electronically maintained information about a patients' health status and their interaction with the overall health system. In addition to this EMR unit, this system can analyze emergency room data, laboratory reports and other types of electronic medical information. The power of the keyword search, especially in the narrative component of clinical data, is designed to capture implicit errors that would otherwise be underreported.

References

[1] Senders, J., Moray, N., (1991). *Human Error: Cause, Prediction, and Reduction.* Lawrence Erlbaum Associates, Hillsdale, New Jersey.

[2] Reason, J. (1990). *Human Error.* Cambridge University Press, Cambridge, England.

[3] Kohn, L.T., Corrigan, J.M. & Donaldsen M.S. (Eds.) (2000). *To Err is Human: Building a Safer Health System.* National Academy Press, Washington D.C.

[4] Pace, W.D., Fernald, D.H., Harris, D. M., Dickinson, L.M., Araya-Guerra, R., Staton, E.W., Van Vorst, R., Parnes, B.L. & Main, D.S. *Developing a Taxonomy for Coding Ambulatory Medical Errors: A Report from the ASIPS Collaborative, 2000.*

[5] *NCC MERP (National Coordinating Council for Medication Error Reporting and Prevention.* NCC MERP Taxonomy of Medication Error. Available: www.nccmerp.org/pdf/taxo2001-07-31.pdf

[6] Kopec, D., Kabir, M.H., Reinharth, D., Rothschild, O. & Castiglione, J.A. (2003). Human Errors in Medical Practice: Systematic Classification and Reduction with Automated Information Systems. *Journal of Medical Systems*, 27(4): 297-313.

[7] Shagas, G. (2004). *Information Technology in Health Care Quality Management and Outcome-Based Practices*, Masters Thesis, Brooklyn College, CUNY.

[8] Beck K., Erich Gamma. *Junit Cookbook.* http://junit.sourceforge.net/doc/cookbook/cookbook.htm Site visited Mar 12, 2006.

[9] Davis, M.G., Incremental development with Ant and Junit *IBM Developer Works Java technology*. http://www-128.ibm.com/developerworks/java/library/j-ant/ Site visited Mar 12, 2006.

[10] Hatcher E. Automating the build and test process. *IBM Developer Work: Java technology*. 14 Aug 2001. http://www-128.ibm.com/developerworks/java/library/j-junitmail/ Site visited Mar 12, 2006.

[11] Class MockStrutsTestCase http://strutstestcase.sourceforge.net/api/servletunit/struts/MockStrutsTestCase.html / Site visited Mar 12, 2006

[12] Prohorenko A., Prohorenko O, Using Junit with Eclipse IDE 2/4/2004. http://www.onjava.com/pub/a/onjava/2004/02/04/juie.html Site visited Mar 12, 2006

[13] *Eclipse Platform Overview*. The Eclipse Foundation http://www.eclipse.org/platform/overview.php. Site visited Mar 12, 2006

[14] Clark M., *JUnit FAQ*. Sourceforge.net Last modified on February 20, 2006. http://junit.sourceforge.net/doc/faq/faq.htm Site visited Mar 12, 2006

[15] StrutsTestCase for JUnit v2.1.3. http://strutstestcase.sourceforge.net/ 2004. Site visited Mar 12, 2006

[16] H. Cao, P. Stetson and G. Hripcsak, (2003). Assessing explicit error reporting in the narrative electronic medical record using keyword searching, *Journal of Biomedical Informatics*, Vol.36, pp.99-105.

[17] Novello A.C., *NYPORTS: the New York Patient Occurance and Tracking System annual report* New York State Health Department 1999.

[18] Berry L.L. *et al.* (1998). Sensitivity and specificity of three methods of detecting adverse drug reactions, *American Journal of* Hospital Pharmacy, Vol.45, pp.1534-1539.

[19] Harrington (2005), Revisiting medical error: five years after the IOM report, have reporting systems made a measurable difference?, *Health Matrix*, Vol.15, No.329.