Analysis of clinical decision support system malfunctions: a case series and survey

RECEIVED 6 November 2015
REVISED 7 January 2016
ACCEPTED 12 January 2016
PUBLISHED ONLINE FIRST 28 March 2016

Adam Wright,[1,2,3,*] Thu-Trang T Hickman,[1] Dustin McEvoy,[3] Skye Aaron,[1] Angela Ai,[1] Jan Marie Andersen,[1] Salman Hussain,[1,4] Rachel Ramoni,[2,5] Julie Fiskio,[1] Dean F Sittig,[6] and David W Bates[1,2,3]

AMIA INFORMATICS PROFESSIONALS. LEADING THE WAY. OXFORD UNIVERSITY PRESS

ABSTRACT

Objective To illustrate ways in which clinical decision support systems (CDSSs) malfunction and identify patterns of such malfunctions.
Materials and Methods We identified and investigated several CDSS malfunctions at Brigham and Women's Hospital and present them as a case series. We also conducted a preliminary survey of Chief Medical Information Officers to assess the frequency of such malfunctions.
Results We identified four CDSS malfunctions at Brigham and Women's Hospital: (1) an alert for monitoring thyroid function in patients receiving amiodarone stopped working when an internal identifier for amiodarone was changed in another system; (2) an alert for lead screening for children stopped working when the rule was inadvertently edited; (3) a software upgrade of the electronic health record software caused numerous spurious alerts to fire; and (4) a malfunction in an external drug classification system caused an alert to inappropriately suggest antiplatelet drugs, such as aspirin, for patients already taking one. We found that 93% of the Chief Medical Information Officers who responded to our survey had experienced at least one CDSS malfunction, and two-thirds experienced malfunctions at least annually.
Discussion CDSS malfunctions are widespread and often persist for long periods. The failure of alerts to fire is particularly difficult to detect. A range of causes, including changes in codes and fields, software upgrades, inadvertent disabling or editing of rules, and malfunctions of external systems commonly contribute to CDSS malfunctions, and current approaches for preventing and detecting such malfunctions are inadequate.
Conclusion CDSS malfunctions occur commonly and often go undetected. Better methods are needed to prevent and detect these malfunctions.

Keywords: clinical decision support, electronic health records, safety, anomaly detection, machine learning

RESEARCH AND APPLICATIONS

Gregory: "Is there any other point to which you would wish to draw my attention?"
Holmes: "To the curious incident of the dog in the night-time."
Gregory: "The dog did nothing in the night-time."
Holmes: "That was the curious incident."
–Sir Arthur Conan Doyle, in the short story "The Silver Blaze"

BACKGROUND AND SIGNIFICANCE

Significant and mounting evidence suggests that clinical decision support (CDS), when used effectively, can improve healthcare quality, safety, and effectiveness.[1–6] A large part of the electronic health record's (EHR) promise to improve patient care rests upon CDS. Because of this, implementing CDS has been an increasingly substantial part of the Meaningful Use program.[7,8]

Nevertheless, these benefits are accompanied by risks, and significant unintended consequences as well as safety issues surrounding CDS have been reported.[9–17] As we will describe in this paper, these problems are widespread and clinically important, and their importance may be underappreciated. Further, as Sherlock Holmes astutely noted above, it is easy to overlook the failure of something (such as a CDS alert) to occur. In the Holmes story, the detective notes that the dog on the property did not bark on the night a prize-winning horse disappears and its trainer is murdered, and correctly deduces that the dog must have known the intruder who perpetrated these crimes.

Finding ways to improve the safety of these systems is still a work in progress. As stated by the Institute of Medicine, "Health IT [information technology] creates new opportunities to improve patient safety that do not exist in paper-based systems. For example, paper-based systems cannot detect and alert clinicians of drug-drug interactions, whereas electronic clinical decision support systems can. As a result, the expectations for safer care may be higher in a health IT-enabled environment as compared to a paper-based environment. However, implementation of health IT products does not automatically improve patient safety. In fact, health IT can be a contributing factor to adverse events . . . [some of which] have led to serious injuries and death."[18–24]

Given the known value of CDS, a particularly significant type of health IT-related adverse event is the malfunction of clinical decision support systems (CDSSs) – ie, situations in which a CDSS does not function as it was designed or expected to. Examples include when alerts stop firing due to a system upgrade, when alerts fire for the wrong patients after a drug dictionary change, and when alerts are inadvertently disabled. To explore the issue of CDSS malfunctions, we present four case studies of malfunctions that occurred at one academic hospital. These cases are illustrative of the range of CDS problems that can occur. Based on these case studies, we also conducted a preliminary survey of Chief Medical Information Officers (CMIOs) to assess whether similar CDSS malfunctions occurred at other sites.

*Correspondence to Adam Wright, PhD, Brigham and Women's Hospital and Harvard Medical School, 1620 Tremont St., Boston, MA 02115, USA; awright@bwh.harvard.edu; Tel: (617) 525-9811; Fax: (617) 732-7072. For numbered affiliations see end of article.

METHODS

The case studies describe CDSS malfunctions that occurred at the Brigham and Women's Hospital (BWH). Until very recently, BWH used the longitudinal medical record (LMR) in the outpatient setting. The LMR was developed locally at BWH and certified by an Office of the National Coordinator for Health Information Technology Authorized Testing and Certification Body. The first case was identified through happenstance, and the remaining three were found by carefully examining alert firing data. In each case, our team conducted an extensive investigation of the CDSS malfunction in order to identify key factors that contributed to the issue. These investigations included a review of alert firing logs, alert rule logic, alert system configuration, audit data, interviews with system developers and users, and audits of source code and system design specifications.

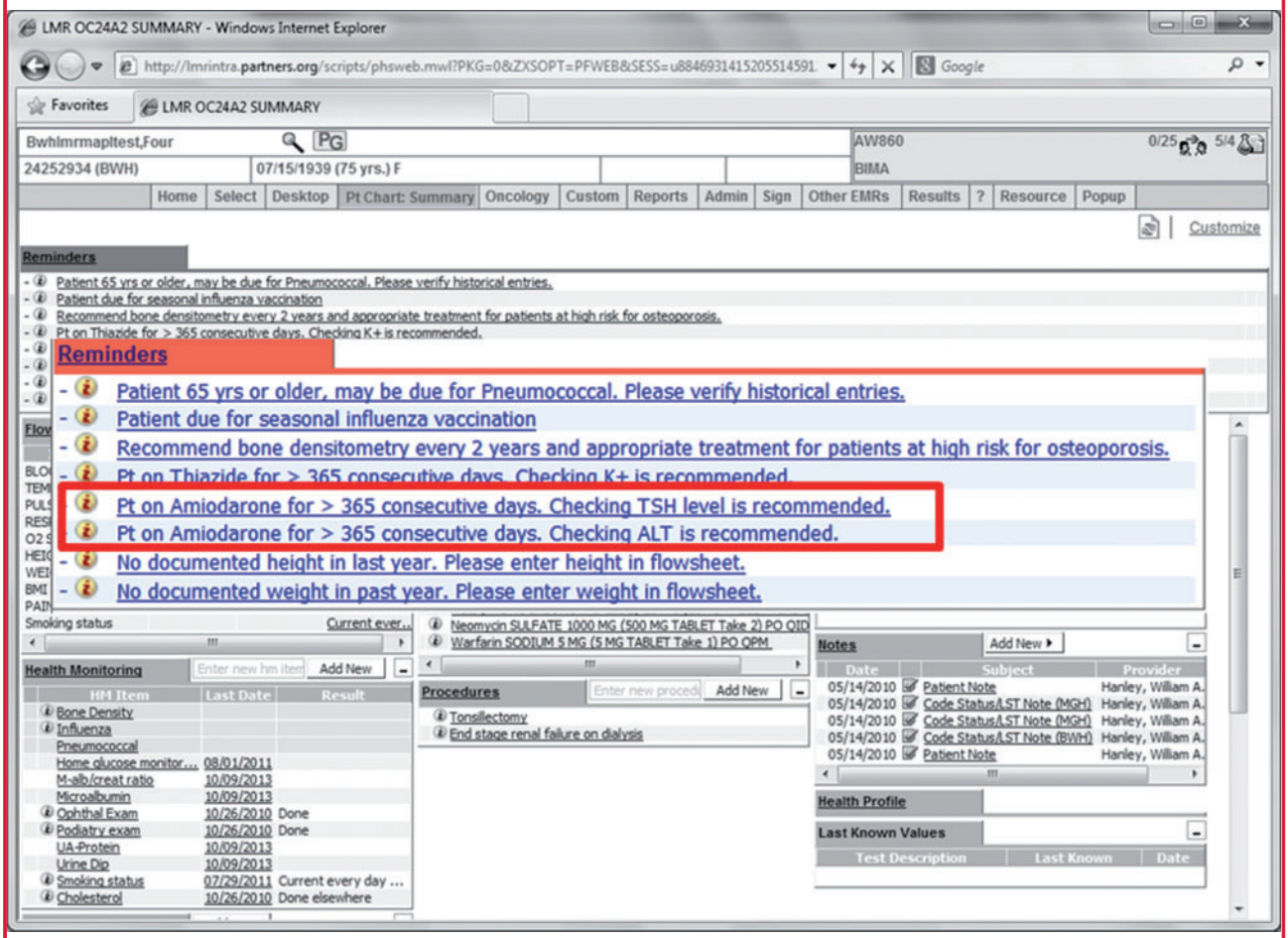
To estimate the size of each anomaly, we extracted the alert firing data for each of the four alerts that exhibited an anomaly, and divided them into data from the period when the malfunction occurred and data from before and after the malfunction occurred. We fit a linear model to the nonanomalous data, adjusting for the date (to account for typical increases in alert volume over time) and whether each date was a weekday or weekend day (because, typically, many fewer alerts fire on weekend days). We then used this model to estimate the expected number of alert firings per day during the period when the malfunction occurred, assuming the alert was working correctly, and subtracted the actual number of firings during the period when the malfunction occurred to estimate the number of excess or missed alert firings while the malfunction was happening.

We conducted these case studies at BWH because we had complete access to the information systems in place there and their source code, the ability to access all data-related to the alerts, and the opportunity to interview the developers and implementers of the systems, which allowed us to conduct a thorough analysis of the issues. We hypothesize, however, that the CDSS malfunctions that occurred at BWH are representative of the types of CDSS malfunctions that occur in self-developed and commercial EHR systems around the world.

We explored this hypothesis by conducting a preliminary survey of a sample of CMIOs at hospitals across the United States to assess common patterns in CDSS malfunctions. The survey was informed by the results of the four BWH case studies and asked respondents about the types of CDS in use in their organizations, the frequency with which CDSS malfunctions occur at their organizations, contributing factors, modes of detection, and the respondent's confidence in the ability of their processes and procedures to prevent or detect CDSS malfunctions.

Both the case studies and the survey were reviewed and approved by the Partners HealthCare Human Subjects Committee.

Figure 1: Laboratory monitoring reminders for amiodarone in the Partners Healthcare longitudinal medical record (LMR). The main screen of the LMR is shown in the background, with the reminders enlarged and the amiodarone reminders highlighted in a box.



RESULTS

Case Study 1: Monitoring Thyroid Function in Patients Receiving Amiodarone

While participating in a cross-vendor evaluation of user-defined CDS,[25] one investigator (A.W.) was giving a demonstration of the LMR and attempted to show its drug-lab monitoring alert capabilities,[26] using the system's suggestion of thyroid-stimulating hormone (TSH) testing for patients who have been on amiodarone for at least 1 year as an example. Amiodarone is an antiarrhythmic drug with many side effects, including, commonly, hypothyroidism and, less commonly, hyperthyroidism. As a result, our hospital considers it important to monitor the thyroid function of patients taking amiodarone by performing a TSH test annually while the patient is receiving the drug. The LMR team created an alert (shown in Figure 1) that suggested ordering a TSH test if the patient has been on amiodarone for at least 1 year and has not had a TSH test within the last year. The pseudocode for the alert is shown in Figure 2.

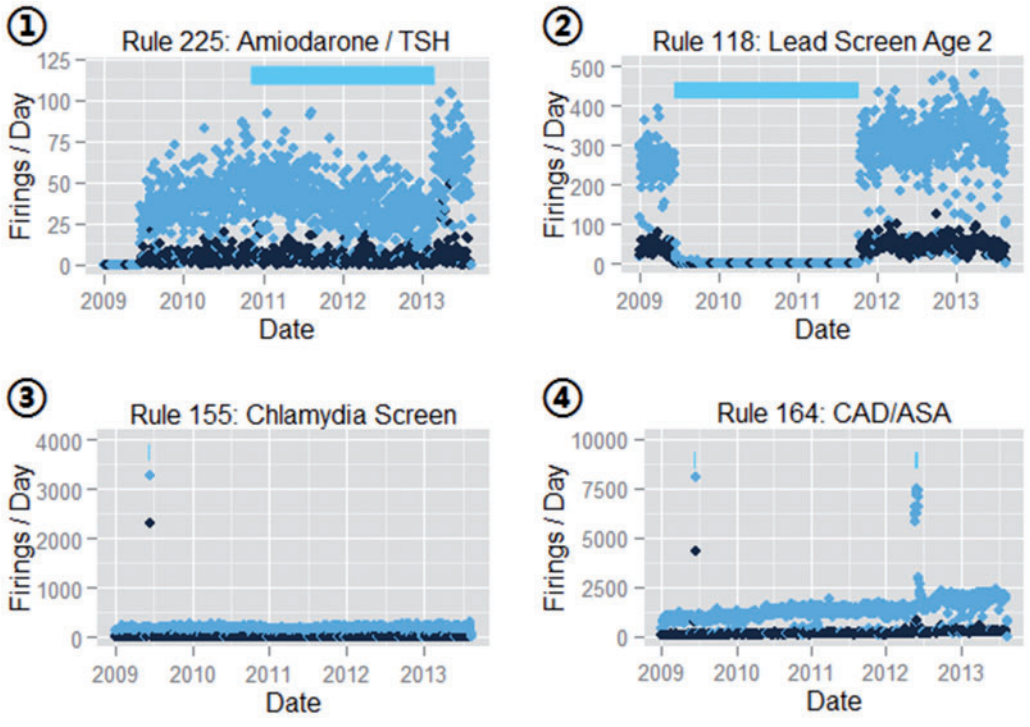
Figure 2: Pseudocode representation of the amiodarone/thyroid-stimulating hormone (TSH) test reminder.

```
if patient is on drug [40 (amiodarone)]
   with start date > [365] days ago
and
most recent lab [tsh] not found
   or > [365] days ago
then
suggest lab [tsh]
```

During the demonstration, the alert unexpectedly failed to fire for several test patients that had been on amiodarone for more than a year and had never had a TSH test. Working with the Partners HealthCare knowledge management team, we discovered that, in November 2009, the LMR's internal code for amiodarone had been changed from 40 to 7099, but the rule logic in the system was never updated to reflect this change. The issue was discovered during the demonstration described above, which took place in February 2013, and fixed the next day.

Panel 1 of Figure 3 shows the pattern of alert firing at BWH. Light blue dots show the number of firings, per day, on weekdays, and dark blue dots show the number of firings on weekend days. The superimposed horizontal blue bar shows the period of malfunction. The malfunction of the amiodarone-TSH alert was initially subtle, because patients who were already receiving amiodarone retained drug ID code 40 in their record, and, thus, the alert continued to fire for these patients; the alert only failed to fire for patients who were started on amiodarone since the November 2009 change in amiodarone's drug ID code in the LMR. Because the alert does not fire until a patient has been on amiodarone for at least a year, there was no observable effect for the first year, and then the rate of alerting subtly fell as some patients were taken off amiodarone (with the old code 40) and others were started on amiodarone (with the new internal LMR code 7099). The abrupt increase in the alert firing rate for the amiodarone/TSH test alert at the end of the blue bar in Figure 3 represents when the alert logic was corrected to include amiodarone's new drug ID code, and the alert began firing again for all patients who were receiving amiodarone (codes 40 *or* 7099). During the

Figure 3: Firing rate of four alerts at Brigham and Women's Hospital over a 5-year period (weekend days are represented by darker dots, and weekdays are represented by lighter dots), with anomalies indicated (superimposed horizontal bars show anomalous periods).



period when this malfunction was occurring, the amiodarone-TSH test alert fired 23 519 times, but the linear model projected that it would fire 33 293 times, yielding an estimated deficit of 9774 missed firings. The same alert can fire multiple times for a single patient, so the number of patients impacted is likely smaller, although still a large number.

Case Study 2: Lead Screenings for 2-Year-Olds

Based on the first case study, our team generated graphs showing the alert firing rate of all 201 active alerts in the LMR and manually reviewed them to look for anomalous patterns. The firing pattern of an alert that suggests that clinicians order lead screenings for 2-year-olds immediately stood out as unusual (see Panel 2 of Figure 3). This alert fires for children between 23 and 29 months of age and suggests that a lead screening test be ordered if no blood lead test result is available from within the previous 6 months. Regular lead testing of children in Massachusetts is required under the Massachusetts Lead Law (M.G.L ch. 111 §194). As seen in Figure 3, the alert firing rate slowed abruptly in 2009, stopped, and then abruptly resumed in 2011. Similar alerts for 1-year-olds, 3-year-olds, and 4-year-olds continued to fire during this period without apparent issue.

We began our investigation by reviewing the rule change logs that are manually maintained by the Partners HealthCare knowledge management team in a database. Each time a knowledge engineer edits a rule in the LMR, he or she is required to document the specific change and the reason for it in this database, which is separated from the actual rule editing environment in the LMR. This database contained no documented record of changes to the rule for the lead screening test alert since 2006. Further investigation revealed no changes to the way that lead tests or ages were recorded (eg, change in the internal LMR code) during the period of interest. By reviewing the LMR's source code, we were able to identify a little-known audit log of changes to its rule logic. Entries in this audit log are automatically generated when rule logic or configuration is changed in the LMR. The audit log suggested that several changes to the lead screening test alert rule were made around the times when the alert stopped firing and then restarted; however, because of a software issue in the audit logging routine, it was not possible to reconstruct the sequence of rule changes or the specific dates when individual changes occurred.

Continuing our investigation, we located system backup tapes in storage from the period when the malfunction was occurring and requested that tapes from 1 month before and 1 month after the malfunction occurred be restored. By reviewing the restored backups, we were able to determine that two additional clauses were added to the lead screening test alert rule, apparently by accident. The first clause checks the patient's gender, and the second evaluates the patient's smoking status. As clauses are added, the values the clause will match must be specified. For example, the gender primitive can be configured to match only male patients, only female patients, or patients with no gender specified, and the smoking status clause can be configured to identify former smokers, current smokers, never-smokers, or patients with an unknown smoking status. When the inadvertent smoking and gender clauses were added to the lead screening test alert rule, no values to match were specified. The result of omitting these required clauses is unspecified in the rule, but, in practice, omitting these values causes the clauses to always evaluate as "false," which prevents the rule from firing. Several years later, the inadvertent clauses in this rule were removed (also without any documentation in the change log), and the rule resumed firing. The linear model estimates that 176 708 lead screening test alerts were not generated during the 850-day period of the alert's malfunction – approximately 208 missed alerts per day. The number of patients this malfunction affected is likely much lower, because alerts are regenerated whenever a patient's chart is opened by a new user or modified, so the same alert can fire several times for a patient during a single visit.

We were unable to determine why the erroneous clauses were added to the lead screening test alert rule, but it may be that the knowledge engineer intended to modify another rule and selected the lead screening test alert rule by mistake. If that is indeed what happened, the testing process at the time would have focused on the rule intended to be modified. There was no regression testing process to make sure that other rules that were not supposed to be modified continued to work as designed.

Case Study 3: System-wide Spike in Alerts

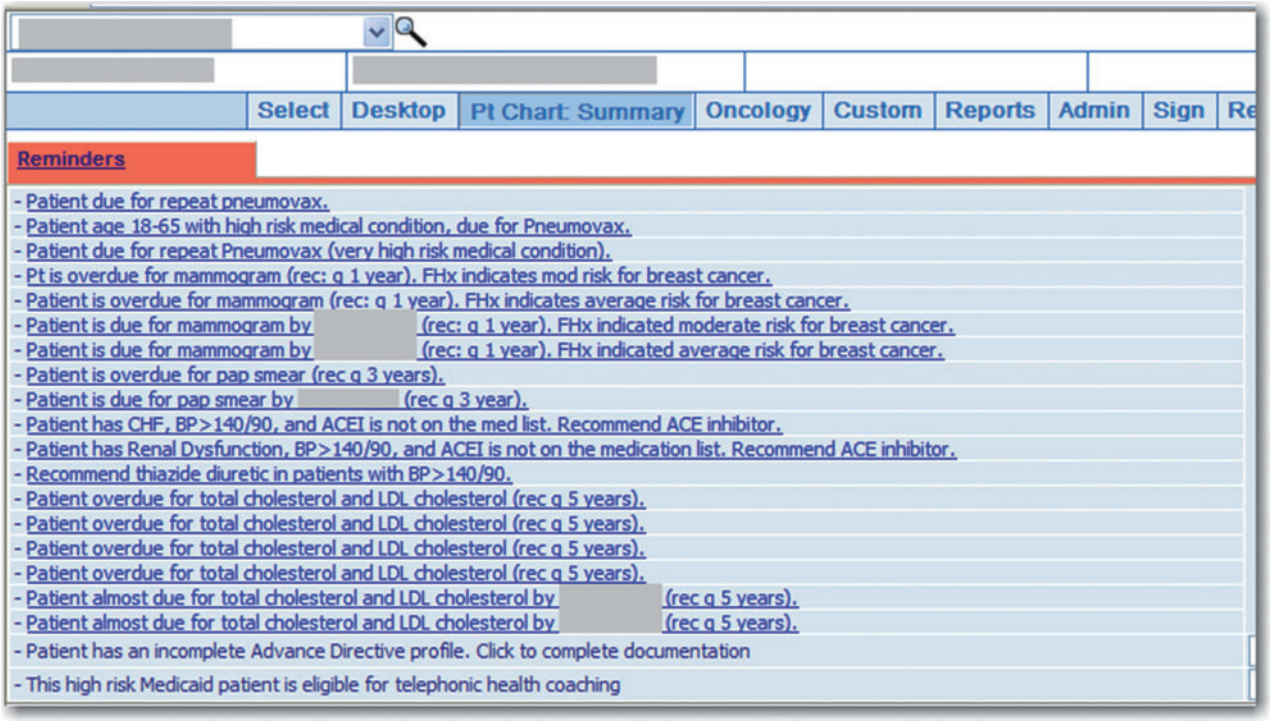
The third case study involved a much less subtle alert malfunction, which was immediately reported by many users and is shown in Panel 3 of Figure 3. The graph shows a representative alert for chlamydia screening, which typically fired a few hundred times a day, then peaked at over 3000 firings on a Sunday and Monday (there was also a less visible increase on the day before, Saturday). The increase in chlamydia screening alert firings coincided with an upgrade to the LMR software. The upgrade included some cleanup to the code that processes reminders – specifically, a command was added to ensure that commonly named variables used in a subroutine did not conflict with variables in the calling function (more technical details are provided in Supplementary Appendix 1). The purpose of this change was to increase the code base's maintainability. Although this change was not intended to have any effect on how the reminder code worked, it inadvertently caused numerous reminders to fire incorrectly for many patients, often multiple times. Figure 4 shows a sample record of a healthy 2-month-old boy that contains numerous duplicate reminders, including suggestions that the physician order mammograms, Pap smears, pneumococcal vaccination, and cholesterol screening, and suggestions that the patient be started on several medications, all of which should not apply to this young, healthy, male patient.

Looking only at the chlamydia screening alert rule, the alert fired 5950 times during the period that the malfunction occurred compared with the 332 times it was expected to fire, yielding an excess of 5618 alerts. All the alerts in the system fired 1 240 017 times during the period that the malfunction occurred compared with the 380 835 times the alerts were expected to fire during this period, yielding an excess of 859 182 alerts that can likely be attributed to the malfunction.

Case Study 4: Antiplatelet Therapy for Patients with Coronary Artery Disease

The final case study involved a spike in one alert (ie, an increased number of firings) reported by LMR users. The alert in question suggests that antiplatelet therapy be ordered for patients with known coronary artery disease, no active antiplatelet therapy, and no contraindications to antiplatelet therapy (a separate rule makes a similar suggestion for patients with relative contraindications, such as peptic ulcer disease). To increase maintainability, the rule for this alert uses drug classes to identify antiplatelet agents (such as aspirin, thienopyridines, and P2Y12 inhibitors). Panel 4 in Figure 3 shows the alert's firing pattern. The system-wide spike in antiplatelet therapy alerts from the period when the malfunction occurred is visible, but another spike in the alert can be seen in 2012. An investigation into the spikes revealed that the reminder system depended on an external drug classification service. This service malfunctioned due to a database issue, which occurred

Figure 4: Sample reminder display for a 2-month-old boy, containing numerous inappropriate reminders for this patient.



after the server hosting the service was rebooted, that caused it to begin reporting that no drugs were in the antiplatelet classes. As a result, the antiplatelet therapy alert would fire for all patients with coronary artery disease, regardless of whether they were already taking an antiplatelet drug. Although users began reporting the issue almost immediately, it took several weeks for the teams involved to locate the cause of the issue and resolve it. The linear model estimates that there were 69 592 excess firings of the antiplatelet therapy alert during the 19-day period when the malfunction was occurring (94 343 actual firings compared to 24 751 expected).

Survey

Based on the case studies and similar experiences with other EHRs, we attempted to investigate whether other organizations had reported similar CDSS problems. We were only able to locate two reports of similar malfunctions in the literature; one written by investigators at Columbia Presbyterian Medical Center[27] and the other by investigators (including D.F.S.) from the University of Utah[28] – both articles were over 20 years old.

To assess how widespread these issues might be, we surveyed a nonrepresentative sample of CMIOs through the Association of Medical Directors of Information Systems (AMDIS). AMDIS is a professional organization for CMIOs and other physicians responsible for health IT, including CDSSs, at healthcare organizations in the United States. We e-mailed the survey to the AMDIS mailing list and invited members to provide information on their experiences with CDSS malfunctions. The survey instrument is provided in Supplementary Appendix 2. We received 29 responses.

The results of the survey are shown in Table 1. The 29 CMIOs who responded to the survey reported that they had implemented a range of different CDS types, with nearly all having drug-drug interaction checking and drug-allergy alerts and a sizeable number having a range of other types of CDS. Two CMIOs reported that their organization had never experienced a CDSS malfunction, but the other twenty seven reported experiencing at least one, with the greatest number reporting four or more such malfunctions every year.

The most common contributing factors to CDSS malfunctions reported by the CMIOs mirror BWH's experience: 18 of the survey respondents reported CDSS malfunctions occurring at the time of an upgrade of their EHR software, and 18 reported issues in connection with changes in data codes or data fields (both of which are known high-risk events for CDSS malfunctions[29]). Inadvertently disabling rules, upgrades of other systems, database corruption, and other system malfunctions were also commonly reported causes of CDSS malfunctions. By far, the most common mode of detection of these alerts was reports from end users, and the respondents often identified malfunctions during their own use of the EHR. Many fewer CMIOs reported identifying CDSS malfunctions during pre-implementation testing or ongoing monitoring of their CDSSs. No CMIOs were totally confident that their existing processes and procedures were sufficient to prevent or detect all CDSS malfunctions before they reach the user, and the majority of CMIOs were either "not at all confident" or "not very confident."

We did not collect data on whether the CMIOs' systems were commercial or self-developed, but we expect that the vast majority are commercial systems, given the relative rarity of self-developed systems still in use in the United States and by AMDIS members. To validate this assumption, we used the Healthcare Information and Management Systems Society Analytics Database to identify the EHR in use at the organization represented by the most recent 50 posters to the AMDIS mailing list who worked at a healthcare provider

Table 1: Results of Preliminary Survey of CMIOs

Survey Item and Responses	n of 29 (%)
Which types of CDS are currently in use at your site?	
Drug-drug interaction alerts	28 (97)
Allergy alerts	27 (93)
Screening/preventive care reminders	21 (72)
Renal dose adjustments	16 (55)
Alerts about abnormal test results	16 (55)
Drug-pregnancy alerts	11 (38)
Reminders to patients	5 (17)
How often has your site experienced CDSS malfunctions?	
4 or more times a year	11 (38)
1–6 times a year	8 (28)
Less than 1 time a year	8 (28)
Never	2 (7)
Did any of these factors contribute to CDSS malfunctions that you experienced?	
Upgrade of your EHR software	18 (62)
Changes to underlying codes or data fields	18 (62)
Inadvertent disabling or enabling of a rule	12 (41)
Upgrade of another clinical information system	10 (34)
Database corruption or another system malfunction	7 (24)
How did you find the malfunctions?	
Report from users	24 (83)
Noticed in my own use of the system	14 (48)
Ongoing system testing	9 (31)
Reviewing reports of CDSS performance	6 (21)
How confident are you that your existing processes and procedures are sufficient to prevent or detect all CDSS malfunctions before they reach the user?	
Totally confident	0 (0)
Very confident	2 (7)
Somewhat confident	7 (24)
Not very confident	12 (41)
Not at all confident	6 (21)

CDS, clinical decision support; CDSS, clinical decision support system; CMIO, Chief Medical Information Officer; EHR, electronic health record.

organization. Of these 50 posters, 49 worked at organizations that used a commercial EHR, and 1 organization was in the process of transitioning from a self-developed system to a commercial system, suggesting that commercial EHRs predominate at the healthcare organizations of AMDIS members.

Given the nature of the mailing list and the survey, we approached the survey as more of a forecasting and consensus exercise than as an exercise that provides a representative sample of CDSSs malfunctions. However, even in the context of this limitation, the results of the survey demonstrate that CDSS malfunctions are more widespread than is currently represented in the scientific literature.[30]

DISCUSSION

Key Findings

Our findings suggest that CDSS malfunctions are widespread, that they likely occur much more frequently than has previously been described, and that existing detection systems (including testing processes, monitoring procedures, and feedback systems) are inadequate to detect CDSS malfunctions before they reach users. Importantly, two of the malfunctions in the BWH case series went unrecognized for over a year and one for several weeks – only the most dramatic malfunction was identified and fixed within a few days.

We observed several patterns in our study. First, many CDSS malfunctions appear to be caused by issues related to changes in data codes or clinical terminology. These changes are often made outside the CDSS by analysts working on different aspects of the EHR infrastructure, but unless CDSS authors are aware of these changes, they can cause malfunctions. A related issue is the effect of malfunctions in related software modules or systems on an otherwise correct CDSS. As seen at BWH, when a separate drug classification system malfunctioned, CDS that depended on that system stopped working correctly. As CDSSs become more complex and modular, the number of points of failure multiplies. If CDSSs are dependent on separate systems and do not have alternative paths, even a single malfunction in these outside systems can cause significant issues.

Another common cause of CDSS malfunctions observed in our study was the inadvertent editing (as in the first case study) or disabling of rules. Changes to CDS rule knowledge bases need to be made carefully and tested thoroughly (preferably by someone other than the person who edited the rule), which should include regression testing of all other alerts to identify potential defects in rules that previously worked correctly. We also found a heightened risk of CDSS malfunctions during EHR software upgrades. For example, an upgrade at BWH that was considered low-risk caused a dramatic system-wide malfunction that affected all reminders in the LMR system. The CMIOs we surveyed identified software upgrades tied to changes to codes or data fields as the most common cause of the CDSS malfunctions they had experienced at their organizations.

In terms of discovering malfunctions, we found that users were most likely to report issues that manifested as incorrect alerts, particularly when alert volumes spiked dramatically. However, users were less likely to report cases of missing CDS alerts, bringing to mind, again, Sherlock Holmes and the dog in the night. Therefore, error detection systems that rely entirely on user reports to identify CDSS malfunctions are unlikely to be robust. Yet, perhaps not surprisingly, "user report" was the most common mode of malfunction identification reported by the respondents to our survey of CMIOs. Although user reports are a critical source of malfunction identification, CDSS implementers must also construct monitoring and testing strategies and tools that proactively identify and prevent CDSS malfunctions.

A key question is whether CDSS malfunctions such as these are unique to BWH. BWH has a large volume of CDS content,[31,32] which may make monitoring CDSSs more challenging. However, BWH also has advanced processes and tools for knowledge management,[33–36] which should make managing the large volume of CDS content more tractable. In our, admittedly, small survey sample, we found that CDSS malfunctions are nearly universal. The fact that few other sites have reported these issues in the literature suggests that they are not

commonly found when they do occur or that those who find them do not regularly report them. Because such CDSS malfunctions may have important implications for patient safety, we suggest that more attention ought to be paid to them.

During our investigation process, two things stood out as surprising. First, although end users often knew about the CDSS malfunctions, patient safety and quality leaders and software developers were mostly unaware of the issues and were surprised by the frequency with which they occurred. Second, it was quite difficult, at least in the LMR, to piece together the history of changes to CDS rule logic. Change logs were maintained manually outside of the EHR systems, but they were often incomplete and did not always match the logic of actual rules running in the EHR. This suggests that, at least sometimes, changes were made to the rule logic in the EHR but not documented in the change log – a deviation from expected practice. Although we identified an audit capability that was designed to log all the edits to CDS rules, it did not work correctly, and we had to resort to retrieving backup tapes to determine how alerts functioned during past periods.

Recommendations

We provide our key recommendations in Table 2, which also identifies which case studies might have been impacted by these practices (had they been in place when the respective CDSS malfunctions occurred).

Strengths and Limitations

One of our study's key strengths is the depth with which we were able to investigate the CDSS malfunctions we identified. This was enabled by the fact that we had full access to the EHR system's source code, software specifications, documentation, and those responsible for developing and implementing the system. Investigations at sites using commercial EHR systems may be complicated by a lack of access to the system's source code, or its developers, and would likely need to be done in partnership with EHR vendors.[37] Another strength of this study is its mixed methods approach, which combined in-depth case studies with a high-level survey to validate that the problems we observed were not unique to BWH.

Along with these strengths, our study has two important limitations. First, the case studies all come from a single hospital with an internally developed EHR, whereas commercial EHRs predominate nationally. We believe that the types of malfunctions seen at BWH are likely occurring at hospitals that use commercially developed EHRs, and our survey results lend credence to that hypothesis. We encourage other hospitals, particularly those that employ commercial EHRs, to conduct similar reviews of their CDS alert firing logs, to investigate their own CDSSs, and to report their findings. Second, the survey we conducted is preliminary in nature, and the survey sample was designed to be informative but not necessarily representative. The survey is useful for confirming that BWH is not alone in experiencing the CDSS malfunctions we observed, but given the nature of the survey sample, we cannot draw broader general conclusions about the extent and distribution of these malfunctions. Further, we do not have specific data on the systems used by the CMIOs surveyed, nor did they provide details about the CDSS malfunctions they experienced beyond what is given in Table 1, so our understanding of the exact failure modes and their ramifications is limited. And because the malfunctions identified by the survey respondents are self-reported, there may be additional malfunctions that were not discovered or reported by the survey respondents.

Table 2: Recommendations for Increased CDSS Reliability and the Prevention of Malfunctions

Recommendation	Related Case Studies
CDS rules should be tested in the live environment after any CDS-related change and after major EHR software upgrades. This testing should be done for both new rules and existing rules (regression testing).	Cases 2 and 3
Reliable communication strategies should be employed to ensure that changes in clinical terminologies are communicated to all CDSS teams.	Case 1
Tools to support terminology management should have the capability to detect and mitigate the downstream impact of terminology changes. As terms and codes are changed, it should be possible to determine the effects of those changes on order sets, CDS rules, documentation tools, etc.	Case 1
Proactive monitoring tools and strategies should be employed to enable quick detection of malfunctions in the production systems.	Cases 1-4
External services that a CDSS depends on should also be proactively monitored.	Case 4
Critical external systems that support a CDSS, such as classification and terminology systems, should be fault-tolerant and robust.	Case 4
Enhanced software quality assurance testing methods, including unit and integration testing, supported by test scripts, tools, and automated tests, should be employed to ensure that CDSSs function correctly. These tests are particularly important at the time of software upgrades and CDSS content changes.	Cases 1-4 (particularly Case 3)
CDSSs should be tested by a different analyst than the one that built the content.	Case 2

CDS, clinical decision support; CDSS, clinical decision support system; EHR, electronic health record.

Future Work

We have identified several opportunities for future work in this area. First, we believe that proactive detection systems are needed to identify CDSS malfunctions. Statistical and rule-based process control methodologies, as well as anomaly detection methods, are likely to be useful for finding CDSS malfunctions in real time and should be investigated. Second, we believe that similar investigations should be undertaken at a wider range of clinical sites that have diverse types of EHRs and CDS approaches, in order to identify additional modes and patterns of CDSS malfunctions and to assess their clinical implications. If such work is done, it will be possible to create a taxonomy of CDSS malfunction types and causes and to identify best practices for the prevention and early detection of CDSS malfunctions.

Implications

Our findings have implications for several groups. First, implementers of CDSSs should proactively and continually monitor their CDSSs for malfunctions. Second, EHR developers should provide system implementers with enhanced tools for reporting on and monitoring CDSSs, and implementers should be encouraged to use these tools regularly. Ideally, such tools would proactively alert CDSS implementers to potential CDSS malfunctions that are identified using statistical or rule-based approaches.

Third, policymakers should make use of levers to encourage development and adoption of CDSS monitoring capabilities. The 2015 edition of Office of the National Coordinator's certification criteria proposes criteria that would require a certified "Health IT Module to be able to record at least one action taken and by whom when a CDS intervention is provided to a user (e.g., whether the user viewed, accepted, declined, ignored, overrode, provided a rationale or explanation for the action taken, took some other type of action not listed here, or otherwise commented on the CDS intervention)" and that "a Health IT Module be able to generate either a human readable display or human readable report of the responses and actions taken and by whom when a CDS intervention is provided."[38] Although we agree that this is important baseline functionality for CDSSs, reports and monitoring tools that work across patients are also needed to detect CDSS malfunctions proactively. Fourth, as proposals for increasing oversight of health IT safety[39–41] are implemented and safety reporting systems for EHRs become available, data on CDSS malfunctions should be collected and analyzed as an important subcategory of health IT safety events.

CONCLUSION

CDSS malfunctions are common and often go undetected. The failure of alerts to fire is particularly difficult to detect. A range of causes, including changes in codes and fields, software upgrades, inadvertent disabling or editing of rules, and malfunctions of external systems commonly contribute to these malfunctions, and current approaches for preventing and detecting CDSS malfunctions are inadequate. As CDSSs becomes more complex and widespread and clinicians increase their reliance on them, improved processes and tools for preventing and detecting CDSS malfunctions are essential.

CONTRIBUTORS

The contributions of the authors are: Wright: conception and design; acquisition, analysis, and interpretation of data; drafting of the manuscript; statistical analysis; supervision. Hickman, McEvoy, Aaron, Ai, Andersen, Hussain, Ramoni, Fiskio: acquisition, analysis, and interpretation of data; critical revision of the manuscript for important intellectual content. Bates and Sittig: acquisition, analysis, and interpretation of data; critical revision of the manuscript for important intellectual content; supervision.

FUNDING

The research reported in this publication was supported by the National Library of Medicine of the National Institutes of Health under award number R01LM011966. The content of this article is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

COMPETING INTERESTS

None.

SUPPLEMENTARY MATERIAL

Supplementary material is available online at http://jamia.oxfordjournals.org/.

REFERENCES

1. Buntin MB, Burke MF, Hoaglin MC, Blumenthal D. The benefits of health information technology: a review of the recent literature shows predominantly positive results. *Health Affairs (Project Hope).* 2011;30(3):464–471.
2. Lyman JA, Cohn WF, Bloomrosen M, Detmer DE. Clinical decision support: progress and opportunities. *J Am Med Inform Assoc.* 2010;17(5):487–492.
3. Kawamoto K, Lobach DF. Proposal for fulfilling strategic objectives of the U.S. Roadmap for national action on clinical decision support through a service-oriented architecture leveraging HL7 services. *J Am Med Inform Assoc.* 2007;14(2):146–155.
4. Garg AX, Adhikari NKJ, McDonald H, *et al*. Effects of computerized clinical decision support systems on practitioner performance and patient outcomes. *JAMA.* 2005;293(10):1223–1238.
5. Kawamoto K, Houlihan CA, Balas EA, Lobach DF. Improving clinical practice using clinical decision support systems: a systematic review of trials to identify features critical to success. *BMJ.* 2005;330(7494):765.
6. Chaudhry B, Wang J, Wu S, *et al*. Systematic review: impact of health information technology on quality, efficiency, and costs of medical care. *Ann Int Med.* 2006;144(10):742–752.
7. Office of the National Coordinator for Health Information Technology. *Step 5: Achieve Meaningful Use Stage 1 – Clinical Decision Support Rule*. 2014. https://www.healthit.gov/providers-professionals/achieve-meaningful-use/core-measures-2/clinical-decision-support-rule. Accessed February 2, 2016.
8. Centers for Medicare and Medicaid Services (CMS). *Stage 2 Eligible Professional Meaningful Use Core Measures Measure 6 of 17*. 2012. http://www.cms.gov/Regulations-and-Guidance/Legislation/EHRIncentivePrograms/downloads/Stage2_EPCore_6_ClinicalDecisionSupport.pdf. Accessed February 12, 2016.
9. Vashitz G, Meyer J, Parmet Y, Liebermann N, Gilutz H. Factors affecting physicians compliance with enrollment suggestions into a clinical reminders intervention. *Stud Health Technol Informatics.* 2010;160(Pt 2):796–800.
10. Sittig DF, Singh H. A new sociotechnical model for studying health information technology in complex adaptive healthcare systems. *Qual Safety Health Care.* 2010;19(Suppl 3):i68–i74.
11. Shackelton RJ, Marceau LD, Link CL, McKinlay JB. The intended and unintended consequences of clinical guidelines. *J Eval Clin Pract.* 2009;15(6):1035–1042.
12. McCoy AB, Waitman LR, Lewis JB, *et al*. A framework for evaluating the appropriateness of clinical decision support alerts and responses. *J Am Med Inform Assoc.* 2012;19(3):346–352.
13. Landman AB, Takhar SS, Wang SL, *et al*. The hazard of software updates to clinical workstations: a natural experiment. *J Am Med Inform Assoc.* 2013;20:187–190.
14. Campbell EM, Sittig DF, Ash JS, Guappone KP, Dykstra RH. Types of unintended consequences related to computerized provider order entry. *J Am Med Inform Assoc.* 2006;13(5):547–556.
15. Bright TJ, Wong A, Dhurjati R, *et al*. Effect of clinical decision-support systems: a systematic review. *Ann Int Med.* 2012;157(1):29–43.
16. Ash JS, Sittig DF, Poon EG, Guappone K, Campbell E, Dykstra RH. The extent and importance of unintended consequences related to computerized provider order entry. *J Am Med Inform Assoc.* 2007;14(4):415–423.
17. Ash JS, Sittig DF, Campbell EM, Guappone KP, Dykstra RH. Some unintended consequences of clinical decision support systems. *AMIA Annual Symposium Proceedings/AMIA Symposium AMIA Symposium.* 2007;26–30.
18. Aleccia J. MSNBC.com. 2011. Nurse's suicide highlights twin tragedies of medical errors. http://www.msnbc.msn.com/id/43529641/ns/health-health_care/t/nurses-suicide-highlights-twin-tragedies-medical-errors/. Accessed February 12, 2016.
19. Associated Press. *Veterans Given Wrong Drug Doses due to Glitch*. 2009. http://www.msnbc.msn.com/id/28655104/#.Tph-b3l2Y9Z. Accessed February 12, 2016.

20. Graham J, Dizikes C. Baby's death spotlights safety risks linked to computerized systems. 2011. http://www.latimes.com/health/ct-met-technology-errors-20110627,0,2158183.story. Accessed February 12, 2016.
21. Schulte F, Schwartz E. *As Doctors Shift to Electronic Health Systems, Signs of Harm Emerge*. 2010. http://huffpostfund.org/stories/2010/04/doctors-shift-electronic-health-systems-signs-harm-emerge. Accessed September 9, 2013.
22. Silver JD, Hamill SD. *Doctor, Nurse Disciplined by UPMC in Kidney Transplant Case*. 2011. http://www.post-gazette.com/pg/11147/1149429-114-0.stm#ixzz1amZZEw6Y. Accessed February 12, 2016.
23. US News. *E-prescribing Doesn't Slash Errors, Study Finds*. 2011. http://health.usnews.com/health-news/family-health/articles/2011/06/30/e-prescribing-doesnt-slash-errors–study-finds. Accessed February 12, 2016.
24. Institute of Medicine Committee on Patient Safety and Health Information Technology. *Health IT and Patient Safety Building Safer Systems for Better Care*. Washington, DC: Institute of Medicine; 2012.
25. McCoy AB, Wright A, Sittig DF. Cross-vendor evaluation of key user-defined clinical decision support capabilities: a scenario-based assessment of certified electronic health records with guidelines for future development. *J Am Med Inform Assoc*. 2015;22(5):1081–1088.
26. Matheny ME, Sequist TD, Seger AC, *et al*. A randomized trial of electronic clinical reminders to improve medication laboratory monitoring. *J Am Med Inform Assoc*. 2008;15(4):424–429.
27. Hripcsak G. Monitoring the monitor: automated statistical tracking of a clinical event monitor. *Comp Biomed Res*. 1993;26(5):449–466.
28. Sittig DF, Pace NL, Gardner RM, Beck E, Morris AH. Implementation of a computerized patient advice system using the HELP clinical information system. *Comp Biomed Res*. 1989;22(5):474–487.
29. Office of the National Coordinator for Health Information Technology, Ash JS, Singh H, Sittig DF. *System Interfaces SAFER Guide*. 2014. https://www.healthit.gov/safer/guide/sg005. Accessed February 12, 2016.
30. Institute of Medicine. Committee on Patient Safety and Health Information Technology. *Health IT and Patient Safety: Building Safer Systems for Better Care*. Washington, DC: National Academies Press; 2012.
31. Garg AX, Adhikari NK, McDonald H, *et al*. Effects of computerized clinical decision support systems on practitioner performance and patient outcomes: a systematic review. *JAMA*. 2005;293(10):1223–1238.
32. Wright A, Goldberg H, Hongsermeier T, Middleton B. A description and functional taxonomy of rule-based decision support content at a large integrated delivery network. *J Am Med Inform Assoc*. 2007;14(4):489–496.
33. Collins SA, Bavuso K, Zuccotti G, Rocha RA. Lessons learned for collaborative clinical content development. *Appl Clin Informatics*. 2013;4(2):304–316.
34. Sittig DF, Wright A, Meltzer S, *et al*. Comparison of clinical knowledge management capabilities of commercially-available and leading internally-developed electronic health records. *BMC Med Informatics Decis Mak*. 2011;11:13.
35. Sittig DF, Wright A, Simonaitis L, *et al*. The state of the art in clinical knowledge management: an inventory of tools and techniques. *Int J Med Informatics*. 2010;79(1):44–57.
36. Hongsermeier T, Glaser J. Managing the investment in clinical decision support. In: Greenes RA, ed. *Clinical Decision Support. The Road to Broad Adoption*, 2nd edn. New York: Elsevier; 2014:665–688.
37. Sittig DF, Classen DC, Singh H. Patient safety goals for the proposed Federal Health Information Technology Safety Center. *J Am Med Inform Assoc*. 2015;22(2):472–478.
38. US Department of Health and Human Services. 45 CFR Part 170: 2015 Edition Health Information Technology (Health IT) Certification Criteria, 2015 Edition Base Electronic Health Record (EHR) Definition, and ONC Health IT Certification Program Modifications. 2015. https://s3.amazonaws.com/public-inspection.federalregister.gov/2015-06612.pdf. Accessed February 12, 2016.
39. Sittig DF, Ash JS, Singh H. The SAFER guides: empowering organizations to improve the safety and effectiveness of electronic health records. *Am J Managed Care*. 2014;20(5):418–423.
40. Sittig DF, Classen DC. Safe electronic health record use requires a comprehensive monitoring and evaluation framework. *JAMA*. 2010;303(5):450–451.
41. Office of the National Coordinator for Health Information Technology. *Health Information Technology Patient Safety Action & Surveillance Plan*. 2014. https://www.healthit.gov/sites/default/files/safety_plan_master.pdf. Accessed February 12, 2016.

AUTHOR AFFILIATIONS

[1]Brigham & Women's Hospital, Boston, MA, USA

[2]Harvard Medical School, Boston, MA, USA

[3]Partners HealthCare, Boston, MA, USA

[4]The Dartmouth Institute for Health Policy and Clinical Practice, Lebanon, NH, USA

[5]Harvard School of Dental Medicine, Boston, MA, USA

[6]University of Texas Health Science Center, Houston, TX, USA