Best Practices for Preventing Malfunctions in Rule-Based Clinical Decision Support Alerts and Reminders: Results of a Delphi Study

Adam Wright, PhD[1,2,3,*], **Joan Ash, PhD, MLS, MS, MBA**[4], **Skye Aaron**[1], **Angela Ai**[5], **Thu-Trang T. Hickman, MPH**[3], **Jane F. Wiesen, PhD**[4], **William Galanter, MD, PhD, MS**[7], **Allison B. McCoy, PhD**[6], **Richard Schreiber, MD**[8], **Christopher Longhurst, MD**[9], and **Dean F. Sittig, PhD**[10]

[1]Division of General Internal Medicine and Primary Care, Brigham and Women's Hospital, Boston, MA

[2]Department of Biomedical Informatics, Harvard Medical School, Boston, MA

[3]Information Systems, Partners Healthcare, Boston, MA

[4]Department of Medical Informatics and Clinical Epidemiology, Oregon Health & Science University, Portland, OR

[5]School of Medicine and Public Health, University of Wisconsin at Madison, Madison, WI

[6]School of Public Health and Tropical Medicine, Tulane University, New Orleans, LA

[7]Department of Internal Medicine, University of Illinois at Chicago, Chicago, IL

[8]Department of Internal Medicine, Geisinger Holy Spirit Hospital, Camp Hill, PA

[9]Department of Biomedical Informatics, University of California San Diego, San Diego, CA

[10]School of Biomedical Informatics, University of Texas Health Science Center at Houston, Houston, TX

Abstract

Objective: Developing effective and reliable rule-based clinical decision support (CDS) alerts and reminders is challenging. Using a previously developed taxonomy for alert malfunctions, we

---

[*]**Corresponding Author:** Adam Wright, Ph.D., Brigham and Women's Hospital and Harvard Medical School 1620 Tremont St. Boston, MA 02115, Phone: (617) 525-9811, Fax: (617) 732-7072 awright@bwh.harvard.edu.

identified best practices for developing, testing, implementing, and maintaining alerts and avoiding malfunctions.

Materials and Methods: We identified 72 initial practices from the literature, interviews with subject matter experts, and prior research. To refine, enrich, and prioritize the list of practices, we used the Delphi method with two rounds of consensus-building and refinement. We used a larger than normal panel of experts to include a wide representation of CDS subject matter experts from various disciplines.

Results: 28 experts completed Round 1 and 25 completed Round 2. Round 1 narrowed the list to 47 best practices in 7 categories: knowledge management, designing and specifying, building, testing, deployment, monitoring and feedback, and people and governance. Round 2 developed consensus on the importance and feasibility of each best practice.

Discussion: The Delphi panel identified a range of best practices that may help to improve implementation of rule-based CDS and avert malfunctions. Due to limitations on resources and personnel, not everyone can implement all best practices. The most robust processes require investing in a data warehouse. Experts also pointed to the issue of shared responsibility between the healthcare organization and the electronic health record vendor.

Conclusion: These 47 best practices represent an ideal situation. The research identifies the balance between importance and difficulty, highlights the challenges faced by organizations seeking to implement CDS, and describes several opportunities for future research to reduce alert malfunctions.

---

1 Background

In 2007, AMIA published "A Roadmap for National Action on Clinical Decision Support" [1]. The roadmap defined clinical decision support (CDS) as systems which provide "clinicians, staff, patients, or other individuals with knowledge and person-specific information, intelligently filtered or presented at appropriate times, to enhance health and health care" and, further, that such a system "encompasses a variety of tools and interventions such as computerized alerts and reminders, clinical guidelines, order sets, patient data reports and dashboards, documentation templates, diagnostic support, and clinical workflow tools" [1 p.141].

Since that seminal paper was published, studies of CDS systems have shown that they can improve quality and safety and reduce costs [2–10]. However, it has also been shown that developing and maintaining CDS systems can be costly, time-consuming and complex [11–25].

As part of a larger project on CDS malfunctions, we have previously described different malfunctions in CDS systems [26–29], with a particular focus on rule-based alerts and reminders. We define a CDS malfunction as "an event where a CDS intervention does not work as designed or expected" [30]. Following our identification of four malfunctions in

rule-based CDS alerts at the Brigham and Women's Hospital (BWH) [29], we conducted a large mixed-methods study that included: site visits, interviews with CDS developers and managers, a survey of physician informaticists, and statistical analyses of CDS alert firing and override rates, all with a goal of identifying additional examples of CDS alert malfunctions [30]. Using these methods, we identified and investigated 68 alert malfunctions that occurred at 14 different healthcare provider organizations in the United States, using a wide variety of commercial and self-developed electronic health record (EHR) systems. We also asked informants at each site about best practices for preventing malfunctions, including those already in use and those being contemplated.

One result of these efforts was the development of a taxonomy of malfunctions in CDS alerts. The taxonomy was organized under four axes: the cause of the malfunction, its mode of discovery, when it began, and how it affected alert firing [30]. The taxonomy is reproduced in Figure 1.

After identifying and classifying the causes and effects of CDS alert malfunctions within a taxonomy, we turned our attention to strategies that represent best practices for preventing and detecting CDS alert malfunctions. In this paper, we present the results of a rigorous Delphi study of approaches for preventing, detecting and mitigating such malfunctions. This work is a continuation of our prior work on CDS malfunctions.

2   Methods

2.1   Initial Identification of Best Practices

To identify best practices, we began by reviewing transcripts and field notes from our mixed-methods study [30]. We also analyzed our database of 68 CDS alert malfunctions to identify practices that might have prevented each malfunction. The five core research team members did this by meeting and discussing each malfunction, brainstorming possible practices for prevention of that particular malfunction, and reaching consensus on the most important best practices.

In many cases, a practice which had already been described by an interview subject in the mixed-methods study could have prevented the malfunction. In other cases, none of the practices provided by interview subjects would have prevented the malfunction, but another method was identified through existing recommendations in the literature or brainstorming and discussion by members of our research team.

2.2   Expert Consensus Development Using a Modified Delphi Method

2.2.1   Delphi Method Overview—To refine, enrich and prioritize the list of practices, we selected the Delphi method [31]. The Delphi method is one of several expert consensus methods, along with the nominal group technique and consensus development conferences [32]. The Delphi method was developed by the RAND Corporation under the name "Project Delphi", an allusion to the Oracle at Delphi which prophesied the future. In 1951, RAND conducted a classified experiment using the Delphi method for the United States Air Force to assess the number of bombs that would be needed in a hypothetical war between the

United States and the Soviet Union. The results of the experiment were originally classified, but were declassified in 1963 [33].

The Delphi method has been described in detail in many articles [31, 33–35]. Briefly, a Delphi study uses a panel of experts who answer questionnaires in two or more rounds. Between each round, the facilitator shares anonymous feedback from the experts and develops a revised questionnaire which takes the expert feedback into account [31]. The goal of the Delphi method is that the experts will reach a consensus that is more accurate than any of their original perspectives. Moreover, since the process is anonymous and asynchronous, it is less likely to be dominated by particularly vocal or persuasive experts, allowing each expert to share his or her perspective without being influenced by the other experts.

The method has been used widely in clinical guideline development, public health and healthcare applications [36–46]. It has also been effective in informatics research [47, 48]. We selected it for our study because of its strong performance and the fact that it did not require in-person collaboration, given that our experts were geographically distributed.

2.2.2 Survey Development—We used a web based data collection platform for administering each round of our Delphi study. After completing the qualitative study and database analysis described above, our research team identified an initial list of practices and sorted them into categories. A prototype of the Round 1 survey instrument was developed using this list. The instrument was presented to pilot testers (members of the research team and additional experts) who agreed to provide early feedback as pilot users of the collection instrument. After pilot testing, the research team revised the survey instrument, eliminated or consolidated some best practices and modified others to create the final survey instrument used in Round 1. A similar process was used after Round 1 to develop the Round 2 survey instrument. Our study was reviewed by the Partners Healthcare Human Subjects Committee, which found it to be exempt.

2.2.3 Expert Panel—To begin our Delphi process, we identified 36 experts through a literature search and expert nomination process representing five areas of EHR and CDS expertise:

1. Representatives of medical specialty societies and policymakers involved in CDS

2. Employees of EHR software vendors who focus on CDS

3. Employees of clinical content vendors (including online medical references, drug compendia and providers of ready-made CDS content)

4. Biomedical informaticians who conduct research on CDS

5. Applied clinical informatics specialists who develop and manage CDS in healthcare organizations

To form our expert panel, we contacted potential experts by email, and personally invited them to participate in the study. Those who agreed received two rounds of surveys to complete. Many Delphi studies use only a few experts (from 6 to 12), but we felt that

representation from each of the five groups was critical, so we invited several experts in each category.

2.3 Delphi Round 1

In Round 1, each category of best practice items was presented as one screen in the survey, and respondents could freely move forward and backward between the screens. For each practice, the experts were asked to indicate "whether you think the item should be included on a list of best practices to prevent CDS malfunctions" by selecting one of five responses: "Keep," "Leaning toward keep," "Leaning toward delete," "Delete," or "I don't understand what this means." Respondents were able to provide open-ended feedback about each item. On each screen, respondents were also asked to list additional practices in the category that were not included, and to give additional comments or feedback about the items on the screen. General feedback about the survey was solicited on the final screen. Experts were asked to focus on the value of each practice and were informed that feasibility would be evaluated in Round 2.

2.4 Delphi Round 2

In Round 2, experts were asked to rate each practice in terms of importance. A five-point Likert scale was used, with responses ranging from "Not at all important" to "Very important." Additionally, experts were asked to rate "how difficult or easy you believe it would be to implement the best practice based on your experience." A five-point Likert scale was used, with responses ranging from "Very difficult" to "Very easy." For both questions, experts were instructed to "imagine you are answering for a hypothetical, well-resourced organization." Because the goal of Round 2 was to rank the existing list, rather than to revise it, experts were not asked to provide open-ended feedback about each item as in Round 1. There was, however, room for open-ended feedback at the end of each category and at the end of the survey.

2.5 Data Analysis

Analysis of the Round 1 results consisted of meetings of 5 researchers discussing each item and respondent comments to reach consensus on keeping, eliminating, merging, or modifying each best practice. Once the Round 2 responses were collected, the importance and difficulty scores were averaged over all respondents for each item. Best practice items were sorted by mean importance in decreasing order. Each item was assigned a difficulty rating of one, two or three depending on the difficulty ratings provided by the experts. A rating of three dots (see Table 1) indicates the item was thought to be relatively difficult, receiving a mean ease score up to 2.5; a rating of two indicates the item was thought to be moderately difficult, receiving a mean ease score between 2.5 and 3.5; and a rating of one indicates the practice was thought to be relatively easy, receiving a mean ease score 3.5 or greater.

Qualitative data analysis of comments provided by experts was conducted using a grounded hermeneutic approach [49, 50].

3 Results

After completing pilot testing of the initial list of 72 best practices in 9 categories, our core team eliminated seven practices (primarily because they were duplicates of other practices, or because of poor ratings during internal pilot testing), revised 12 for clarity and specificity, added 1 item, and eliminated 1 category (moving the practices it contained to other categories). The final survey instrument used in Round 1 contained 66 best practice items in 8 categories.

Of the 36 experts we invited to participate in the study, 32 agreed to participate and four declined (three because they were too busy, and one because of technical issues with the survey software). Of the 32 experts who agreed to participate as experts in Round 1, 28 completed the Round 1 survey.

After reviewing and summarizing the feedback from the 28 Round 1 experts, the research team eliminated 19 additional practices, revised 37 for specificity or clarity, and eliminated 1 category. We then developed and pilot tested the survey instrument which was used for Round 2. The final survey instrument used in Round 2 contained 47 best practice items in 7 categories.

Of the 28 experts who completed the Round 1 survey, 1 was recruited to be an additional pilot tester for Round 2. The remaining 27 were invited to complete the Round 2 survey. Of the 27 experts invited to complete Round 2, 25 of them completed it.

After collecting Round 2 responses, the results of Round 2 were analyzed as described above. The final results are shown in Tables 1–3.

3.1 Pre-Implementation Best Practices

Table 1 shows best practices from the pre-implementation phases of CDS alerts. This phase included practices for knowledge management, the design and specification of CDS, building CDS alerts in an EHR, and testing these alerts, both prior to go-live and once it has gone live in production [51]. The sequence of the best practices is in descending order of the mean importance scores across all respondents, but because some practices were eliminated during prior rounds, all practices in the tables should be considered reasonably important. The pre-implementation best practices phase is the largest, with 26 listed in it vs. 16 for deploying and monitoring and 6 for people and governance. The main emphasis of the 26 pre-implementation practices is organization of processes to assure that the CDS is accurate and useful.

3.2 Deployment, Monitoring and Feedback Best Practices

Table 2 summarizes the best practices identified for deploying CDS alerts, as well as for monitoring alerts and collecting feedback from end users. While Table 1 lists practices that lay the groundwork for assuring implementation of high quality alerts, Table 2 lists the best practices the expert panel thought help sites to identify and mitigate malfunctions.

3.3 People and Governance Best Practices

Table 3 shows the identified best practices for people and governance. These practices are presented in a separate table, as they span the pre- and post-implementation phases of the CDS alert lifecycle.

Figure 2 presents a scatter plot of all the best practices, grouped by category. The items were grouped close together and tended to be considered important and moderately difficult to implement. Items in the upper right of the figure represent items that were on average considered more important and easier to implement (although at any one organization, the importance and ease of an item may vary depending on needs and resources).

3.4 Qualitative Results

In addition to providing scores for importance and difficulty, the Delphi experts provided rich feedback on many of the items, and on the survey as a whole.

Many comments provided by the experts related to the challenge of assessing the importance and difficulty of the items in a general context. The experts noted that the degree of difficulty of some of the best practices depends on the EHR an organization uses. For example, some commercial EHRs have tools for migrating CDS alert content between environments – a best practice – while others require alert developers to rebuild content in each environment. Experts also indicated that some best practices, such as maintaining a detailed inventory of CDS alerts, matter more in organizations that have a large number of CDS interventions or artifacts and are less important when the size of the CDS alert inventory is smaller. The experts also pointed out that some best practices (particularly those related to knowledge management and testing) are easier for organizations with a larger and more experienced knowledge management staff and would be more difficult in organizations without such staff.

Experts also noted that not all practices need to be used for all types of alerts. For example, a detailed specification and thorough testing would be necessary for a complex sepsis identification alert, but might be excessive for a simple reminder to weigh a patient at each visit.

Several best practices involved correlation of alerts using data from an organization's data warehouse, including using the data warehouse to assess how often an alert will fire, to identify situations where an alert was expected to fire but did not, or to monitor the effects of the CDS. Many experts rated these approaches very highly, but some thought they were "overkill" or not feasible for all but the largest organizations.

One key issue that was raised several times was the issue of shared responsibility [52, 53]. Although malfunction-free alerting is ultimately the responsibility of the healthcare organization implementing the alerting system, EHR and clinical content vendors can provide tools and guidance to help system implementers. Our experts thought that vendors should provide better tools for testing and deploying CDS alerts and for monitoring their use.

4 Discussion

Based on this feedback, we recommend that organizations carefully consider their local context and resources as they identify and prioritize best practices to adopt.

As the experts noted, not all practices are needed for all types of alerts. Organizations adopting these best practices should be thoughtful about the extent to which each best practice is used, and whether it is used uniformly, or only for certain kinds of alerts. We also recommend that organizations that plan to offer a robust CDS alerting program strongly consider investing in a data warehouse and use it to inform the alert development process. Some EHRs also offer more direct links between their data warehouse and CDS alerts by, for example, allowing knowledge engineers to find patients in the data warehouse for which a particular CDS alert would fire or, conversely, allowing knowledge engineers to construct a data warehouse query that identifies a cohort of patients, and then to reuse the logic or results of the query to trigger CDS alerts. We consider these features to be useful and encourage EHR vendors to make them available.

The experts also raised the issue of shared responsibility. Although we believe that healthcare provider organizations are ultimately responsible for the quality of their CDS alerts, EHR vendors play an important role in facilitating success. At a minimum, EHR vendors should provide robust tools for building, maintaining, monitoring and evaluating CDS alerts, as well as thorough documentation and responsive technical support. EHR vendors should also consider providing their customers implementation and analysis services, or at least advice, relating to CDS alerts. Practically, EHR vendors should provide tools that allow for direct migration of CDS alert content between EHR system environments, and should also provide easy methods for viewing and exporting alert logic, evaluating the effectiveness of alerts, and detecting potential issues or anomalies with alerts. On the other hand, healthcare provider organizations should ensure that their staff are knowledgeable and properly trained and certified before allowing them to build alerts, should perform appropriate analysis, quality assurance and monitoring on CDS interventions, and should be responsive to user feedback. An important implication is that many health information technology departments are understaffed for CDS alert work – a finding echoed by a recent study on informatics staffing at critical-access hospitals[54].

Many of the best practices identified represent ideal situations, and no organization we encountered implements all of them. They point to numerous opportunities for future research in minimizing alert malfunctions and improving CDS reliability. Some of these were considered too novel to be included as best practices at present, but likely to be beneficial if research proves their feasibility and utility. The first opportunity is the development of dependency management tools to identify and mitigate the effects of changes in related clinical systems. For example, a tool that identifies all alerts that depend on a specified medication code could be used to automatically notify personnel about alerts that need to be updated when a medication code is changed or deleted in the medication dictionary, or when a new code is added which contains medications similar (e.g., in the same therapeutic category) to those in existing rules. Without this kind of tool, both the medication knowledge engineer and the owners of impacted alerts may not even be aware

that an alert has been affected by the medication dictionary change. The panel of experts also identified research on tools for automated testing of alerts to be a high research priority.

Our study has several strengths. First, it draws on the collective experience of diverse sites across the country and the expertise of a diverse range of experts from across the CDS alerting landscape. It also used a rigorous Delphi method to identify and prioritize best practices. It also has several limitations. First, all sites and experts in the study were based in the United States. Although we anticipate that the findings would apply internationally (in part because some of the EHR and content vendors represented have been implemented internationally), we have not assessed this hypothesis, which is a ripe opportunity for future work. We asked the experts to adopt the perspective of a "hypothetical well-resourced organization," but it is likely that the importance and difficulty of some best practices vary depending on the size and experience of an organization and its knowledge management resources, as well as the amount of CDS the organization uses, and the extent to which it develops it in-house or purchases it. The Delphi method is not well-suited to studying the influence of these variables; a larger survey would be required – also an opportunity for future work.

Another issue that came up during the Delphi discussions was the scope of a malfunction. Some cases, such as when a correctly-designed CDS alert stops firing due to a software upgrade, are clearly malfunctions. In other cases, the alert is functioning as designed, and the design is largely reasonable, but there are still opportunities to optimize its logic and function. In other cases, the alert is working as well as it possibly could, but focuses on an unimportant topic, causing users to perceive it as distracting. In our Delphi, we asked the expert to consider events "where a CDS intervention does not work as designed or expected" as malfunctions. However, just because a particular alert is free from malfunctions does not mean that it is necessary reliable or effective in improving quality of care. We believe that, taken together, the best practices we outlined should promote reliability and effectiveness of alerts; however, the focus of the Delphi was on malfunctions, so additional best practices may be useful for dealing with issues of alert reliability and effectiveness beyond malfunctions.

As the capabilities of alerting within all types of health care organizations increase along with the optimization of electronic health record systems, more of the best practices listed in Tables 1 to 3 will need to be seriously considered and eventually implemented. Organizations must be willing to devote increasing resources to this effort and vendors must help customers by providing more sophisticated tools.

5 Conclusion

CDS alert malfunctions are a key challenge to the safety and reliability of EHRs. Best practices for preventing and managing malfunctions in alerting systems are known, and organizations should consider adopting them according to their own institutional needs and resources. At the same time, EHR vendors should provide tools and guidance to assist healthcare organizations in implementing these practices.

Acknowledgements

References

1. Osheroff JA, Teich JM, Middleton B, et al. A roadmap for national action on clinical decision support. J Am Med Inform Assoc 2007;14(2):141–45 [PubMed: 17213487]

2. Bright TJ, Wong A, Dhurjati R, et al. Effect of clinical decision-support systems: a systematic review. Ann Intern Med 2012;157(1):29–43 [PubMed: 22751758]

3. Wolfstadt JI, Gurwitz JH, Field TS, et al. The effect of computerized physician order entry with clinical decision support on the rates of adverse drug events: a systematic review. J Gen Intern Med 2008;23(4):451–58 [PubMed: 18373144]

4. Pearson S-A, Moxey A, Robertson J, et al. Do computerised clinical decision support systems for prescribing change practice? A systematic review of the literature (1990-2007). BMC health services research 2009;9(1):1 [PubMed: 19118494]

5. Kawamoto K, Houlihan CA, Balas EA, et al. Improving clinical practice using clinical decision support systems: a systematic review of trials to identify features critical to success. BMJ 2005;330(7494):765 [PubMed: 15767266]

6. Kaushal R, Shojania KG, Bates DW. Effects of computerized physician order entry and clinical decision support systems on medication safety: a systematic review. Arch Intern Med 2003;163(12):1409–16 [PubMed: 12824090]

7. Jaspers MW, Smeulers M, Vermeulen H, et al. Effects of clinical decision-support systems on practitioner performance and patient outcomes: a synthesis of high-quality systematic review findings. J Am Med Inform Assoc 2011;18(3):327–34 [PubMed: 21422100]

8. Hunt DL, Haynes RB, Hanna SE, et al. Effects of computer-based clinical decision support systems on physician performance and patient outcomes: a systematic review. J Am Med Assoc 1998;280(15):1339–46

9. Garg AX, Adhikari NK, McDonald H, et al. Effects of computerized clinical decision support systems on practitioner performance and patient outcomes: a systematic review. J Am Med Assoc 2005;293(10):1223–38

10. Roshanov PS, Fernandes N, Wilczynski JM, et al. Features of effective computerised clinical decision support systems: meta-regression of 162 randomised trials. BMJ 2013;346:f657 [PubMed: 23412440]

11. Wang D, Peleg M, Tu SW, et al. Representation primitives, process models and patient data in computer-interpretable clinical practice guidelines:: A literature review of guideline representation models. Int J Med Inform 2002;68(1):59–70 [PubMed: 12467791]

12. Sittig DF, Wright A, Osheroff JA, et al. Grand challenges in clinical decision support. J Biomed Inform 2008;41(2):387–92 [PubMed: 18029232]

13. McCoy A, Wright A, Eysenbach G, et al. State of the art in clinical informatics: evidence and examples. Yearb Med Inform 2013;8(1):13–9 [PubMed: 23974543]

14. Boxwala AA, Rocha BH, Maviglia S, et al. A multi-layered framework for disseminating knowledge for computer-based decision support. J Am Med Inform Assoc 2011;18(Supplement 1):i132–i39 [PubMed: 22052898]

15. 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 Inform 2010;79(1):44–57 [PubMed: 19828364]

16. Kakabadse NK, Kakabadse A, Kouzmin A. Reviewing the knowledge management literature: towards a taxonomy. Journal of knowledge management 2003;7(4):75–91

17. Bali RK. Clinical knowledge management: opportunities and challenges: IGI Global, 2005.

18. 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 Inform Decis Mak 2011;11(1):1 [PubMed: 21211015]

19. Glaser J, Hongsermeier T. Managing the investment in clinical decision support Clinical decision support: the road ahead: Elsevier, Inc., Burlington, MA, 2006.

20. Ash JS, Sittig DF, Guappone KP, et al. Recommended practices for computerized clinical decision support and knowledge management in community settings: a qualitative study. BMC Med Inform Decis Mak 2012;12(1):1 [PubMed: 22217121]

21. Wright A, Bates DW, Middleton B, et al. Creating and sharing clinical decision support content with Web 2.0: Issues and examples. J Biomed Inform 2009;42(2):334–46 [PubMed: 18935982]

22. Wright A, Sittig DF, Ash JS, et al. Governance for clinical decision support: case studies and recommended practices from leading institutions. J Am Med Infrom Assoc 2011;18(2):187–94

23. Sirajuddin AM, Osheroff JA, Sittig DF, et al. Implementation pearls from a new guidebook on improving medication use and outcomes with clinical decision support: effective CDS is essential for addressing healthcare performance improvement imperatives. J Healthc Inf Manag 2009;23(4): 38

24. Osheroff JA. Improving Medication Use and Outcomes with Clinical Decision Support: A Step by Step Guide; 2009 HIMSS.

25. Lyman JA, Cohn WF, Bloomrosen M, et al. Clinical decision support: progress and opportunities. Journal of the American Medical Informatics Association 2010;17(5):487–92 [PubMed: 20819850]

26. McCoy AB, Thomas EJ, Krousel-Wood M, et al. Clinician Evaluation of Clinical Decision Support Alert and Response Appropriateness. AMIA Annual Symposium Proceedings 2015 2015

27. 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):346–52 doi: 10.1136/amiajnl-2011-000185. [PubMed: 21849334]

28. McCoy AB, Thomas EJ, Krousel-Wood M, et al. Clinical decision support alert appropriateness: a review and proposal for improvement. Ochsner J 2014;14(2):195–202 [PubMed: 24940129]

29. Wright A, Hickman TT, McEvoy D, et al. Analysis of clinical decision support system malfunctions: a case series and survey. J Am Med Inform Assoc 2016;23(6):1068–76 doi: 10.1093/jamia/ocw005. [PubMed: 27026616]

30. Wright A, Ai A, Ash J, et al. Clinical decision support alert malfunctions: analysis and empirically derived taxonomy. J Am Med Inform Assoc 2017 doi: 10.1093/jamia/ocx106.

31. Linstone HA, Turoff M. The Delphi method: Techniques and applications: Addison-Wesley Reading, MA, 1975.

32. Black N, Murphy M, Lamping D, et al. Consensus development methods: a review of best practice in creating clinical guidelines. Journal of health services research & policy 1999;4(4):236–48 doi: 10.1177/135581969900400410. [PubMed: 10623041]

33. Dalkey N, Helmer O. An experimental application of the Delphi method to the use of experts. Management science 1963;9(3):458–67

34. Okoli C, Pawlowski SD. The Delphi method as a research tool: an example, design considerations and applications. Information & management 2004;42(1):15–29

35. Dalkey NC, Brown BB, Cochran S. The Delphi method: An experimental study of group opinion: Rand Corporation Santa Monica, CA, 1969.

36. Hearnshaw HM, Harker RM, Cheater FM, et al. Expert consensus on the desirable characteristics of review criteria for improvement of health care quality. Qual Health Care 2001;10(3):173–8 [PubMed: 11533425]

37. Marshall M, Klazinga N, Leatherman S, et al. OECD Health Care Quality Indicator Project. The expert panel on primary care prevention and health promotion. Int J Qual Health Care 2006;18 **Suppl** 1:21–5 doi: 10.1093/intqhc/mzl021. [PubMed: 16954512]

38. O'Boyle C, Jackson M, Henly SJ. Staffing requirements for infection control programs in US health care facilities: Delphi project. Am J Infect Control 2002;30(6):321–33 [PubMed: 12360140]

39. Zhang W, Moskowitz RW, Nuki G, et al. OARSI recommendations for the management of hip and knee osteoarthritis, Part II: OARSI evidence-based, expert consensus guidelines. Osteoarthritis Cartilage 2008;16(2):137–62 doi: 10.1016/j.joca.2007.12.013. [PubMed: 18279766]

40. Conchin S, Carey S. The expert's guide to mealtime interventions - A Delphi method survey. Clin Nutr 2017 doi: 10.1016/j.clnu.2017.09.005.

41. Ueda K, Ohtera S, Kaso M, et al. Development of quality indicators for low-risk labor care provided by midwives using a RAND-modified Delphi method. BMC Pregnancy Childbirth 2017;17(1):315 doi: 10.1186/s12884-017-1468-4. [PubMed: 28938879]

42. Sganga G, Tascini C, Sozio E, et al. Early recognition of methicillin-resistant Staphylococcus aureus surgical site infections using risk and protective factors identified by a group of Italian surgeons through Delphi method. World J Emerg Surg 2017;12:25 doi: 10.1186/s13017-017-0136-3. [PubMed: 28616060]

43. Paquette-Warren J, Tyler M, Fournie M, et al. The Diabetes Evaluation Framework for Innovative National Evaluations (DEFINE): Construct and Content Validation Using a Modified Delphi Method. Can J Diabetes 2017;41(3):281–96 doi: 10.1016/j.jcjd.2016.10.011. [PubMed: 28279617]

44. Rodriguez-Manas L, Feart C, Mann G, et al. Searching for an operational definition of frailty: a Delphi method based consensus statement: the frailty operative definition-consensus conference project. J Gerontol A Biol Sci Med Sci 2013;68(1):62–7 doi: 10.1093/gerona/gls119. [PubMed: 22511289]

45. Graham B, Regehr G, Wright JG. Delphi as a method to establish consensus for diagnostic criteria. J Clin Epidemiol 2003;56(12):1150–6 [PubMed: 14680664]

46. Kelly CM, Jorm AF, Kitchener BA. Development of mental health first aid guidelines on how a member of the public can support a person affected by a traumatic event: a Delphi study. BMC Psychiatry 2010;10:49 doi: 10.1186/1471-244X-10-49. [PubMed: 20565918]

47. Liyanage H, Liaw ST, Di Iorio CT, et al. Building a Privacy, Ethics, and Data Access Framework for Real World Computerised Medical Record System Data: A Delphi Study. Contribution of the Primary Health Care Informatics Working Group. Yearb Med Inform 2016(1):138–45 doi: 10.15265/IY-2016-035. [PubMed: 27830242]

48. Meeks DW, Smith MW, Taylor L, et al. An analysis of electronic health record-related patient safety concerns. J Am Med Inform Assoc 2014;21(6):1053–9 doi: 10.1136/amiajnl-2013-002578. [PubMed: 24951796]

49. Crabtree B, Miller W, editors. Doing Qualitative Research. 2 ed. Thousand Oaks, CA: Sage, 1999.

50. Berg BL, Lune H. Qualitative Research Methods for the Social Sciences. 8 ed. Boston: Pearson, 2012.

51. Wright A, Aaron S, Sittig DF. Testing electronic health records in the "production" environment: an essential step in the journey to a safe and effective health care system. J Am Med Inform Assoc 2017;24(1):188–92 doi: 10.1093/jamia/ocw039. [PubMed: 27107450]

52. Singh H, Sittig DF. Measuring and improving patient safety through health information technology: The Health IT Safety Framework. BMJ Qual Saf 2016;25(4):226–32 doi: 10.1136/bmjqs-2015-004486.

53. Sittig DF, Belmont E, Singh H. Improving the safety of health information technology requires shared responsibility: It is time we all step up. Healthc (Amst) 2017 doi: 10.1016/j.hjdsi. 2017.06.004.

54. Adler-Milstein J, Holmgren AJ, Kralovec P, et al. Electronic health record adoption in US hospitals: the emergence of a digital "advanced use" divide. J Am Med Inform Assoc 2017;24(6): 1142–48 doi: 10.1093/jamia/ocx080. [PubMed: 29016973]

Highlights

- CDS malfunctions are common and may go undetected for long periods of time, putting patients at risk. Institutions have inadequate tools and processes to detect and prevent malfunctions.

- Experts identified, evaluated and prioritized 47 best practices for preventing CDS malfunctions in 7 categories: knowledge management, designing and specifying, building, testing, deployment, monitoring and feedback, and people and governance.

- Experts also pointed to the issue of shared responsibility between the healthcare organization and the electronic health record vendor.

Summary Points

What was already known on the topic

- CDS alert malfunctions are common and may go undetected for long periods of time, putting patients at risk

- Institutions have inadequate tools and processes to detect and prevent alert malfunctions

What this study added to our knowledge

- An expert consensus-based list of best practices in 7 categories for preventing, detecting, and mitigating alert malfunctions

- Estimated difficulty of implementation and importance of implementation is provided for each best practice, to aid institutions in prioritizing which best practices to put in place

Axis 1: What caused the malfunction?

- **Build error:** The clinical conceptualization and design of the rule were sound, but there was an error building the rule that caused it to malfunction
- **Conceptualization error:** There was an error in the clinical conceptualization of the rule which, when implemented, led to a malfunction
- **New code, concept or term introduced, but rule not updated:** A new item was added to one of the data dictionaries used to encode clinical concepts within the EHR's database
- **Defect in EHR software:** There is a programming error in the source code of the EHR which causes the EHR to function other than as designed or documented and this error causes CDS to malfunction
- **Environment migration:** Moving the rules from one environment to the other (e.g., test to production) leads to a malfunction
- **New value:** The set of allowable values or the meaning of the values of a coded concept is changed
- **Alert text mismatch:** The text displayed by the rule and the logic of the rule do not match
- **External service issue:** An external software routine that is used in CDS either fails to function or cannot be reached, creating an error
- **Inadvertent enabling/disabling:** A rule in the production environment is inappropriately enabled or disabled
- **Unaware of component reuse:** A "building block" for CDS, such as a criteria specification or a value set is reused in multiple rules and a knowledge engineer modifies it intending to change one rule, but inadvertently also causes a change in another rule that uses the same component

Axis 2: How was the malfunction discovered?

- **Reported by an end user:** A user notifies the help desk, files a safety report or directly contacts the team responsible for CDS
- **Review of firing data:** Retrospective examination of system-generated alert firing logs reveals anomalies (unusual patterns, such as large spikes or no firings for a long period of time)
- **Review of override reasons:** Retrospective examination of user-entered alert override reasons
- **Testing:** Testing of CDS revealed malfunctions
- **Review of input data element usage:** Reviewing changes in the pattern of use of data elements that are used by the rule logic (e.g., orderable items or lab result)
- **Demonstration of system:** An error was identified while demonstrating how the system worked
- **Found while investigating other error:** Malfunction in one rule found incidentally while investigating an error in another rule

Axis 3: When did the malfunction start?

- **From its implementation in production:** Rule never fired correctly in production environment
- **After deployment into production:** Rule initially worked correctly, but began to fail at some later point

Axis 4: What was the effect of the malfunction on rule firing?

- **Rule fires in situations where it should not fire:** Rule fires for patients that do not meet the intended criteria

- **Rule does not fire in situations where it should:** Rule fails to fire for all patients, or a subset, that do, in fact, meet the intended criteria
- **Rule fires in the correct situations, but suggests the wrong action or displays the wrong text:** The rule fires for the correct patients, but the display text of the rule, or the actions that it offered, are incongruent with its purpose
- **Rule fires in the correct situation, but causes the EHR to operate slowly:** The EHR operates more slowly due to an issue with the way the CDS logic is constructed (commonly because it fetches too much data, or depends on an external service which is slow)

Figure 1:

Taxonomy of Clinical Decision Support Malfunctions (reproduced with permission from Wright A, Ai A, Ash J et al. Clinical decision support alert malfunctions: analysis and empirically derived taxonomy. *Journal of the American Medical Informatics Association*. 2017 Oct 16.)
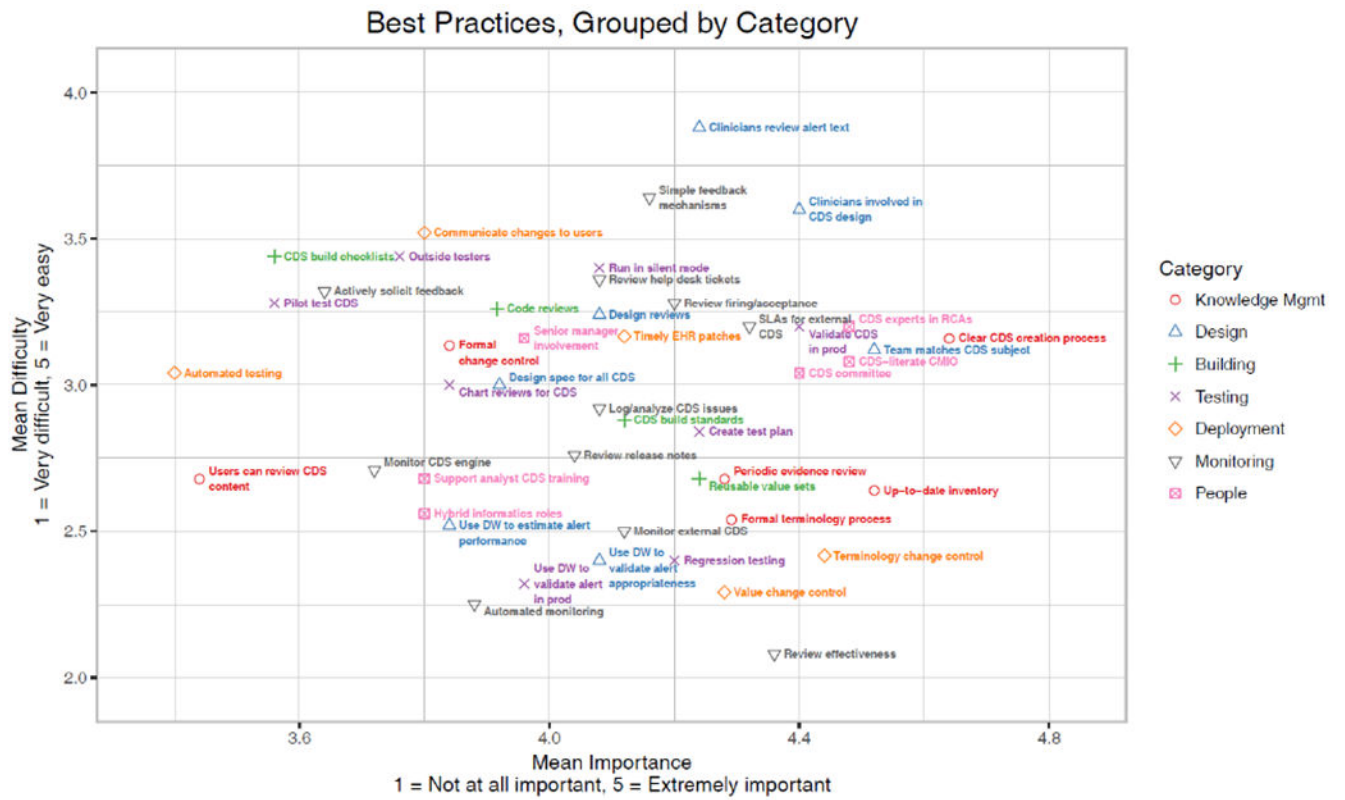
Figure 2:
Scatter plot of all best practices, grouped by category

Table 1:

Best practices for knowledge management and designing, specifying, building and testing CDS alert (ranked by mean importance)

	Mean Importance	Difficulty
Best Practices for Knowledge Management		
Implement a clear, standard process for submission, review, evaluation, prioritization, and creation of all new CDS.	4.6	●●
Maintain an up-to-date inventory of all CDS, including type (e.g., alert, order set, and charting templates), owners, creation and review dates, sources of evidence, clinical area(s) affected, and a short description.	4.5	●●
Manage terminologies and value sets (groupers or classes) using formal processes.	4.3	●●
For internally developed CDS alerts, periodically review the clinical evidence for each alert, make sure the alert is still clinically relevant, and update the alert as needed.	4.3	●●
Use a formal software change control process for all CDS updates	3.8	●●
Enable interested clinical end users to review the logic in human-readable format for CDS rules (e.g., in a portal or repository).	3.4	●●
Best Practicing for Designing and Specifying CDS Alerts		
The teams designing and reviewing a CDS alert specification should be appropriate for the complexity and subject matter of the intervention.	4.5	●●
Have a clear, efficient process for CDS design that includes clinicians from the beginning.	4.4	●
Before building CDS interventions, have clinicians read the alert text and ask them to explain what it means, then compare that to the actual logic of the rule.	4.2	●
Conduct design reviews of CDS specifications with clinical and IT experts prior to building CDS.	4.1	●●
During the alert design phase, query the data warehouse to identify patients for whom a proposed alert would fire and verify that the alert is appropriate for those patients.	4.1	●●●
Prepare a human-readable design specification for each new or updated CDS intervention (the detail of the design specification should match the complexity of the alert logic).	3.9	●●
During the design phase, query the data warehouse to estimate firing rates, sensitivity and specificity of the proposed CDS logic.	3.8	●●
Best Practices for Building CDS Alerts		
Where appropriate, implement CDS using reusable value sets (e.g., groupers or medication classes) rather than lists of concepts hard-coded in the rule.	4.2	●●
Develop and enforce internal CDS build standards (i.e., naming conventions, display text consistency, key definitions).	4.1	●●
Conduct "code review" for CDS alert configuration and logic.	3.9	●●
Builders of CDS should use a checklist of best practices and common pitfalls.	3.6	●●
Best Practices for Testing CDS Alerts		
Test and re-validate CDS after it's moved into the production EHR environment.	4.4	●●
Develop a test plan for each new CDS intervention (the detail of the test plan should depend on the complexity of the alert logic).	4.2	●●
Test all CDS after any significant changes to the EHR or interfaced software to ensure it all still works (regression testing) before putting the EHR changes into production.	4.2	●●●
Before bringing a new alert live, run it in "silent mode" (i.e., send alert messages to a file, rather than to users) to make sure it's working as expected.	4.1	●●
For key alerts, query the data warehouse to identify instances where an alert should have fired and compare that to the instances where the alert fired.	4.0	●●●
After releasing a new alert, conduct chart reviews for a random sample of alert firings (e.g., on 10 patients) to validate alert functionality and design.	3.8	●●

	Mean Importance	Difficulty
For each CDS intervention, have additional testing conducted by an individual other than the person who built it.	3.8	●●
Deploy new CDS to a small number of pilot users and obtain feedback on the CDS before deploying it more broadly.	3.6	●●

Table 2.

Best Practices for deploying and monitoring CDS alerts, and collecting feedback (ranked by mean importance)

	Mean Importance	Difficulty
Best Practices for CDS Alert Deployment		
Use a process where changes to vocabulary codes (e.g. codes that refer to medications or laboratory tests) made by ancillary departments (e.g. pharmacy or laboratory) are communicated to CDS builders and analyzed for impact before changes are made.	4.4	●●●
Employ a process where changes to attribute values (e.g. reference range for a test, units for a measurement, or text string to describe a quantity) are communicated to CDS builders and analyzed for impact before changes are made.	4.3	●●●
Test and deploy EHR vendor patches and upgrades in a timely manner.	4.1	●●
Inform users of significant CDS changes.	3.8	●
Have IT staff use automated tools to migrate CDS rules between EHR system environments (e.g. test and production).	3.4	●●
Best Practices for Monitoring and Feedback		
Regularly review CDS for effectiveness (i.e. is it having a measurable impact on quality, safety, workflow, or efficiency?).	4.4	●●●
For CDS that uses external services, agree upon and enforce a Service Level Agreement (SLA) about uptime, response time, and accuracy.	4.3	●●
Regularly review data on alert firing and acceptance rate data	4.2	●●
Provide simple mechanisms (e.g. telephone or on-screen form) to allow users to provide feedback on or report issues with CDS.	4.2	●
For CDS that uses external software services, employ automated tools for uptime and performance (i.e. response time) monitoring and notification of problems.	4.1	●●●
Review help desk issues reported (e.g. in ticket tracking software) to identify potential CDS issues.	4.1	●●
Maintain a log of CDS alert issues and malfunctions. Review the log for patterns which could help prevent future issues.	4.1	●●
Promptly review all release notes from your EHR vendor and monitor for changes that could impact CDS.	4.0	●●
Use automated monitoring tools to identify and alert CDS developers to potential CDS malfunctions (e.g. anomaly detectors).	3.9	●●●
Monitor the performance of your CDS engine (e.g. number of alerts processed per minute or per hour, or length of the processing queue) and its impact on overall EHR response time.	3.7	●●
Actively solicit feedback on CDS features and functions from users on a regular basis.	3.6	●●

Table 3:

Best Practices for Peoole and Governance of CDS alerts (ranked by mean importance)

	Mean Importance	Difficulty
Best Practices for People and Governance of CDS Alerts		
When safety events occur, have individuals who are knowledgeable about CDS participate in root cause analyses to identify potential EHR issues and CDS opportunities.	4.5	●●
Employ a CMIO who is knowledgeable about CDS and the specific CDS features of the organization's EHR.	4.5	●●
Form a multi-disciplinary CDS committee (e.g., nursing, physicians, lab, pharmacy, risk management, IT, informatics, quality) composed of both administrators and clinical users.	4.4	●●
Involve and update appropriate senior management about CDS's effect on quality and safety on a regular basis.	4.0	●●
Create hybrid positions that blend responsibilities, knowledge, and experience in clinical quality, patient safety, and informatics.	3.8	●●
Train EHR support analysts on CDS so they can triage issues or help end users as appropriate.	3.8	●●