acmqueue Computers in Patient Care: The Promise and the Challenge

**Information technology has the potential to radically transform health care.
Why has progress been so slow?**

Stephen V. Cantrill, MD, FACEP

A 29-year-old female from New York City comes in at 3 a.m. to an ED (emergency department) in California, complaining of severe acute abdominal pain that woke her up. She reports that she is in California attending a wedding and that she has suffered from similar abdominal pain in the recent past, most recently resulting in an appendectomy. The emergency physician performs an abdominal CAT scan and sees what he believes to be an artifact from the appendectomy in her abdominal cavity. He has no information about the patient's past history other than what she is able to tell him; he has no access to any images taken before or after the appendectomy, nor does he have any other vital information about the surgical operative note or follow-up. The physician is left with nothing more than what he can see in front of him. The woman is held overnight for observation and released the following morning symptomatically improved, but essentially undiagnosed.

A vital opportunity has been lost, and it will take several months and several more physicians and diagnostic studies (and quite a bit more abdominal pain) before an exploratory laparotomy will reveal that the woman suffered from a rare (but highly curable) condition, a Meckel's diverticulum. This might well have been discovered that night in California had the physician had access to complete historical information.

This case is recent, but the information problem at its root seems a holdover from an earlier age: Why is it that in terms of automating medical information, we are still attempting to implement concepts that are decades old? With all of the computerization of so many aspects of our daily lives, medical informatics has had limited impact on day-to-day patient care. We have witnessed slow progress in using technology to gather, process, and disseminate patient information, to guide medical practitioners in their provision of care and to couple them to appropriate medical information for their patients' care.

Why such slow progress? Some of the delay certainly has been technologically related, but not as much as one might think. This article looks at some of these issues and the challenges (there are many) that remain.

First, why bother with computers in health care, anyway? There are many potential advantages from the application of health information technology (or HIT, the current buzzword). These include improved communication between a single patient's multiple health-care providers, elimination of needless medical testing, a decrease in medical errors, improved quality of care, improved patient safety, decreased paperwork, and improved legibility (yes, it's still an issue). Many of these improvements have not yet come to pass and many others are nearly impossible to rigorously prove, but for the purposes of this discussion, let's assume that HIT is a good thing.

SOME HISTORY

The first challenge in applying medical informatics to the daily practice of care is to decide how computerization can help patient care and to determine the necessary steps to achieve that goal. This challenge is best summed up by Lawrence L. Weed, M.D.: to develop an information utility that has currency of information and parameters of guidance to assist medical personnel in caring for and documenting the care of patients.³ From the technology side, we need a facile interface between human and machine and a responsive, reliable system that is always available. The assumption is that there will be adequate computational power and mass memory to support such a system.

The history of the computer industry's involvement in these problems is instructive. In the late 1960s, a major computer vendor thought it could solve many hospital-based medical care issues in less than a year by deploying 96-button punch pads throughout the hospital to handle physician orders and intra-hospital communication. Button template overlays were to be used to support different types of orders. As it turned out, this was a most inadequate human interface: cumbersome, inflexible, closed-ended with limited duplex communication, etc. Not surprisingly (at least to the users), this was a nonstarter and failed.

Most of the major hardware vendors of that era also had plans to provide automation of hospital information, creating their versions of a HIS (hospital information system). They all, for various reasons, failed, often with a stunning thud. The most commonly cited deficiencies were a poor human interface, unreliable implementation, and cost. As is often the case when applying new technology to a discipline, the magnitude and complexity of the problem were initially grossly underestimated. As a result, most hardware vendors then limited themselves to the historic area for data processing: patient billing and the financial arena.

In the late 1960s and early 1970s, hardware limitations strained even demonstration systems. Limited main and mass memory, CPU speed, and communication between the CPU and user workstations were all factors that limited system usability and capacity. The human-machine interface was also an issue. Some systems used lightpens with some degree of success.

At that time, I was a member of a small group that was implementing a demonstration electronic medical record system that used touch-sensitive screens: a 24-line by 80-character CRT display allowed two columns of 12 text selections each to be presented to the user, with a branch taken to a new display based upon the selection. This "branched-logic" approach allowed medical users to concatenate a series of selections to create complex text entries for storage into a patient's medical record, as well as to order medications and lab tests and to retrieve previous entries from the patient's medical record.^{1,2} (The ability to type in information was supported for those situations where the displays did not contain the desired medical content.) The major performance goal of this system (and its 20 workstations) was to provide a new text display to any user within 300 milliseconds at least 80 percent of the time, which was quite advanced for its time. This system was designed to be available 24/7/365 with no scheduled downtime.

This demonstration system presented several challenges. First and foremost was the interface between machine and medical providers (physicians, nurses, etc.), as well as patients (for entering their own medical histories). Medicine as a discipline is not known to be necessarily forward looking in adopting new technology, so convincing these individuals to use a revolutionary technology to replace pen and paper was not easy.

The mass-storage limitations were real. The system would support only 144 active patients at any

one time (which was adequate for operation on a single hospital ward but would preclude initially supporting an entire hospital). There was also a limit of 32K individual text screens of information (fancy that!), and there were limits on how far the dumb terminals could be placed from the CPU.

This demonstration system was able to support an entirely computerized medical record (now called an EMR, or electronic medical record) and allowed physicians to use the touchscreen and branched logic displays to enter a patient's history, physical examination, problem list (those unique medical issues for each patient), and progress notes, including patient assessments and orders. For many specific problems, the system would offer a range of recommended treatments (e.g., the appropriate list of drugs for hypertension). As part of the physician ordering sequence for each specific drug, the system would present side effects to watch for, recommended drug monitoring parameters, drug-drug interactions, etc. (This is an obvious precursor to having this checking done automatically.) This level of guidance was possible because of the structured nature of the data entry; it is much more difficult when free text is entered via the keyboard instead.

Why didn't this demo system catch on as hardware and operating systems improved? There were several reasons. At the time, computers were not well understood and, thus, were considered a little scary by the general public, so there was a degree of user hesitation. Also, the level of medical documentation needed and the support of patient safety issues that this system was based upon were not, unfortunately, appreciated at that time. Cost also continued to be an issue. Though this system never caught on, many of the concepts it demonstrated are present in currently evolving commercial systems.

Several other early attempts were made to apply computerization to health care. Most were mainframe-based, driving "dumb" terminals. Many dealt only with the low-hanging fruit of patient order entry and results reporting, with little or no additional clinical data entry. Also, many systems did not attempt to interface with the information originator (e.g., physician) but rather delegated the system use to a hospital ward clerk or nurse, thereby negating the possibility of providing medical guidance to the physician, such as a warning about the dangers of using a specific drug. This is a nontrivial issue that is still a problem with some systems today, illustrating the challenge of an effective user interface.

There were also some efforts to automate the status quo with no attempt to structure the data input. This usually meant having the health-care provider enter free text via a keyboard. Unfortunately, this automation of unstructured data yields only (legible) unstructured data. This may be acceptable when dealing with a system of limited scope but does not work well with massive amounts of information such as a patient record.

These computer systems were quite expensive to install and operate. With this foray into the clinical realm of acute medical care, the requirements for increased reliability of both hardware and software became clear, along with the need for constant accessibility.

AREAS OF REAL TECHNICAL PROGRESS OVER THE YEARS

We have made significant technological advances that solve many of these early shortcomings. Availability of mass storage is no longer a significant issue. Starting with a 7-MB-per-freezer-size-disk drive (which was not very reliable), we now have enterprise storage systems providing extremely large amounts of storage for less than \$1 per gigabyte (and they don't take up an entire room). This advance in storage has been accompanied by a concomitant series of advances in file structures,

database design, and database maintenance utilities, greatly simplifying and accelerating data access and maintenance.

The human-machine interface has seen some improvement with the evolution of the mouse as a pointing device and now the partial reemergence of the touchscreen. We have also seen the development of the graphical user interface, which has facilitated user multitasking.

Overall system architectures have followed an interesting course: from a centralized single CPU and dumb workstations to networks with significant processing capabilities at each workstation. In some situations we are now seeing a movement to the so-called thin-client architecture, again with limited power and resources at each workstation, but a significant improvement in ease of system maintenance.

Of course, all of this has been made possible by improvements in transmission speed of data both between systems and within a single network. These advances in potential system responsiveness, however, have been attenuated by the ever-increasing computational demands of the software, sometimes legitimately, but often caused by the proliferation of bloatware: cumbersome, poorly designed, and inefficiently coded software serving as a CPU-cycle black hole.

An additional complicating factor has been the migration of many pieces of application software to Web-based processes. This does provide the advantage of platform semi-independence, but any slowness of the browser or the Web server is inflicted on the user, and in some cases, may be a dealbreaker in terms of user acceptance. For example, say I use a Web-based system to order a series of medications on a patient and it takes me 10 mouse clicks/screen flips to order a single medication. If it takes one second to move from screen to screen, that is 10 seconds (plus my human processing time). Not bad for a single order, but multiply that by 20 orders per patient over 10 sick patients in a busy emergency department at 1 a.m. on a hectic Saturday night, and you begin to appreciate the issue. There are approaches to minimizing this negative impact, but these require a degree of sophistication of system design that is not always present. In fact, a common complaint of medical users is that “it’s too many clicks to do something simple.”

A very significant area of technological improvement has been in the acquisition, processing, transmission, and presentation (display) of graphical images. This capability has, over the past decade, given us increasingly sophisticated CAT scan and MRI results and has allowed most hospitals to discontinue the use of X-ray film almost completely, using digitally stored images instead. These PACS (picture archiving storage) systems have revolutionized radiology and improved patient care by allowing easy distribution of these images to all care providers of a specific patient, putting an end to the endless problem of trying to chase down the original physical X-ray film.

AREAS OF LIMITED PROGRESS

If we truly want to develop an information utility for health-care delivery in an acute care setting (such as an intensive care unit or emergency department), we need to strive for overall system reliability at least on the order of our electric power grid, ideally with no perceived scheduled or unscheduled downtime. Some health-care information computer systems have achieved a high degree of reliability, but many have not. These lower-performing systems often had their beginnings, as noted earlier, in non-mission-critical applications such as patient billing. This, unfortunately, established a system culture that is permissive of system failure, and this culture is difficult to upgrade.

The culture of system reliability begins with the hardware architecture and progresses through the operating system, the application programs, and the supporting institution-wide infrastructure, physical deployment, and extensive failure mode analysis. This means simple things such as supporting rolling data backups and system updates without taking the system down (from the user's point of view). Some systems boast that they have uptimes of 99.99 percent, but that means they are still unavailable for an hour per year.

Reliability and availability remain ongoing challenges. Certainly, manual procedures for use during system unavailability are necessary, but the goal should be not to have to use them. This is an increasingly important issue as we attempt to develop systems that are more intimately involved in patient care (such as online patient monitoring of vital signs and realtime patient tracking). In fact, we should not even attempt to support mission-critical operations unless we have the hardware, software, and support systems in place that will guarantee extreme overall reliability. Even then it is a risk. I remember the promises from our "state of the art" enterprise RAID 5 storage vendor: "It will never go down." These promises were used to convince me to move off of my dual-write standby server configuration to the enterprise storage system to serve up block storage for my emergency department network. This system is critical, providing realtime ED patient tracking, clinical laboratory result access, patient-care protocol information, emergency department log access, hospital EMR retrieval, metropolitan area hospital ambulance divert status, and physician and nurse order communication, among other functions. Unfortunately, the storage system that was promised to "never go down" had two 5-hour failures over a two-year period, thoroughly dispelling the myth of reliability promised by the vendor. These episodes, unfortunately, are not unique. Through careful design and adequate component redundancy, we have been able to achieve high levels of reliability in safety-critical systems; our patients and health-care providers deserve no less reliability.

Patient data entry in any health information system is labor intensive. Health-care providers (especially physicians) have little tolerance for systems that serve as impediments to getting their work done, often regardless of what positives might accrue from using such a system. This represents a failure of interface and software design and may explain why we are seeing increased use of "scribes" in institutions that have implemented electronic health records. These scribes are individuals who act as recorders for the health-care professionals so they do not have to interface directly with the computer system. Obviously, this greatly diminishes the power of any system since there is no longer an interface with the information originator. The incorporation of dynamic medical guidance (advice rules based upon individual patient data such as checking a drug order for interactions with the patient's other drugs) is of limited utility if the data is entered by someone other than the information originator.

It is also interesting to note that many institutions that had early success with even poorly designed systems were those where the majority of the care was supplied by physicians in training. They were told to use the system "or else" and did not have the flexibility to move to another institution. To maximize user acceptance of any system, we need to continue to improve the human-machine interface, allowing for branched logic content, templated data entry, voice recognition, dynamic pick lists, and when absolutely necessary, free text entry. Physicians care greatly about their patients; if an institution's attempts at computerization do not result in improved patient care and/or improved speed or other significant advantages, acceptance of any system will be problematic. This issue has resulted in the demise of many hospital-based systems.

Even where successfully implemented, computerized health information systems have sometimes had unanticipated side effects. One significant issue is the explosion of data that may be stored in the patient record. This can quickly escalate beyond the capability of the human mind. The challenge remains how best to present the data to a health-care provider in an efficient and comprehensive fashion.

Another potential problem with electronic medical records is abuse of privacy. With old paper medical records, control was somewhat easier: unless copied, they were in only one place at one time. This barrier is removed with computerization, mandating enhanced restrictions to protect data. Unfortunately, we have witnessed several instances of inappropriate access to an individual's medical data. This is most commonly seen when a celebrity is hospitalized and human curiosity results in patient privacy violations (and often subsequent firings). The challenge is to limit inappropriate access but not make legitimate data retrieval burdensome or difficult.

ONGOING BARRIERS TO THE SUCCESS OF HIT

As we continue to strive for advances in health information technology, we must confront several barriers to its success. One significant issue is the balkanization of medical computerization. Historically, there has been little appreciation of the need for an overall system. Instead we have a proliferation of systems that do not integrate well with each other. For example, a patient who is cared for in my emergency department may have his/her data spread across nine different systems during a single visit, with varying degrees of integration and communication among these systems: EDIS (emergency department information system), prehospital care (ambulance) documentation system, the hospital ADT (admission/discharge/transfer) system, computerized clinical laboratory system, electronic data management (medical records) imaging system, hospital pharmacy system, vital-signs monitoring system, hospital radiology ordering system, and PACS system. Ideally, these different systems should be integrated into a seamless whole, at least from the user's point of view, but each has a different user interface with different rules, a different feel, and different expectations of the user. It really is just a bunch of unconnected pieces, which may, in certain situations, actually increase the time and effort for patient care. In this case, the full capability of data integration clearly has not been achieved.

This leads to other concerns: Are we creating health-care computer systems that are so complex that no one has a complete understanding of their vulnerabilities, thus making them prone to failure? Do we have an adequate culture of mission-critical and fault-tolerant design and system support to achieve expected levels of reliability in all hospitals that attempt a high degree of computerization? Is there sophisticated failure analysis to ensure growth, improvement, and success in all of these institutions? Or will the tolerance for unexplained failure actually pose a risk to our patients?

As mentioned, most of these component systems have a medical content piece, as well as a technology piece. It is this creation of the medical logic and structured content in many of these systems (especially the EMR systems) that remains a time-consuming and exacting process, often requiring many person-years of effort for a single institution. Unfortunately, because of the perceived differences in practice patterns among different locales, institutions, and physician groups, only a modicum of the work done in any one location is applicable to other locations. There should be efforts to standardize some of these patterns to allow more synergy between locations and products.

Although grand claims are often made about the potential improvements in the quality of care, decreases in cost, etc., these are very difficult to demonstrate in a rigorous, scientific fashion. Fortunately, the body of positive evidence is slowly increasing, although there are occasional signs of adverse effects from computerized patient data systems. For example, there is evidence that it may be easier to enter the wrong order for the wrong patient in a computerized system than in an old hard-copy, manual system.

THE FUTURE

Although difficult to scientifically prove, the benefits from an electronic medical record and the attendant methodologies to create and maintain it are potentially significant. Yet, we have not come very far conceptually in the past several decades in realizing the potential. Nonetheless, I feel that the future is quite bright for several reasons.

First and foremost, the federal government has championed these concepts with promises of financial support for physicians and institutions that implement the concepts in a meaningful way within a specific timeframe. Second, the use of computers in most aspects of our daily lives has become commonplace, resulting in increased computer literacy and decreasing hostility to their use in a medical environment. Third, with increased national emphasis on patient safety and quality of medical-care indicators, computerization of health care offers the best and easiest approach to provide medical guidance and allow appropriate data capture to comply with these initiatives (which will be ongoing and increasing in number and complexity).

The achievement of desired goals, however, will continue to provide a challenge to system creators and implementers. They have the difficult job of designing, developing, and supporting systems that provide improved reliability and responsiveness and a facile human-machine interface to help us provide better health care to our citizens.

Let us return to the 29-year-old patient with acute abdominal pain in the California emergency department, now under an improved computerized health-care system. The physician in California has instant access to the operative note and medical workup for the appendectomy done many months before. This reveals that, in fact, no radiographs were taken prior to the surgery, which was done laparoscopically. This implies that the finding on the CAT scan is not, because of the surgical technique, an artifact, but an abnormal finding. This would lead in short order to surgical consultation and surgical repair, markedly decreasing the patient's period of morbidity and suffering. Such improvements are the promise of integrating computers in patient care. With effort and skill, I feel we can meet this challenge. ◻

REFERENCES

1. Schultz, J. R. 1988. A History of the PROMIS Technology: An Effective Human Interface. In *A History of Personal Workstations*, ed. A. Goldberg. ACM Press. Reading, MA: Addison-Wesley Publishing Co., Inc.
2. Schultz, J. R., Cantrill, S. V., Morgan, K. G. 1971. An initial operational problem-oriented medical record system—for storage, manipulation and retrieval of medical data. *AFIPS—Conference Proceedings* 38.
3. Weed, L. L., M.D. 1972. Problem-oriented System. In *Background Paper for Concept of National Library Displays*, eds. J. W. Hurst and H. K. Walker. Medcom Press.

LOVE IT, HATE IT? LET US KNOW

feedback@queue.acm.org

DR. STEPHEN CANTRILL has been practicing emergency medicine for more than 30 years. He recently retired from the position of associate director of emergency medicine at Denver Health Medical Center, a safety-net Level I trauma center, where he developed and supported EMeSIS (Emergency Medical Services Information System). He is an associate professor of emergency medicine at the University of Colorado. He first started writing code related to medical informatics in 1966 at Brown University, where he received his A.B. in physics, and he helped develop one of the early computerized medical record systems while working with Dr. Lawrence L. Weed at PROMIS Laboratory. Dr. Cantrill received his M.D. degree from the University of Vermont.

© 2010 ACM 1542-7730/10/0800 \$10.00