The automatic collection and storage of patient information in the electronic health record (EHR) and other data repositories has enhanced healthcare providers' ability to improve patient care and outcomes. The availability of more accurate data has allowed health professionals to apply advanced computations to various clinical use cases, including automated risk scores, 1 text analyses for improved coding, 2 tumor detection in magnetic resonance imaging, 3 automatic adverse drug interaction warnings, 4 and protocol quality and compliance. 5

However, despite these improvements, persistent physiologic gaps in data access, resolution, and synchronization can hinder patient care and quality. Examples involving operational and clinical use cases include:

- Hospitals risk billions of dollars in overbilling due to traditionally nonintegrated mechanical ventilators, resulting in error-prone hand documentation. 6 Although interoperability is considered integral to the process of optimizing care, only one-third of bedside devices are integrated into a larger data picture. 7
- Neonatal intensive care units (NICUs) rely on an apnea detector that is insufficiently accurate, leading to higher patient risk, longer hospital stays, and more clinician time; however, deploying a better detector is difficult because no structure exists for deploying new alerts in real time.
- Predicting whether a patient is in danger of imminent cardiac arrest would be of value to pediatric cardiologists; however, building a predictive model is difficult because high-resolution, 240-Hz electrocardiogram (ECG) waveform data are not recorded and stored en masse for later analysis. Only an estimated five vital signs (heart rate, respiratory rate, blood pressure, temperature, and oxygen saturation) are documented at any one time in the patient record using current methods. With documentation performed once an hour at most, this approach is insufficient for building an effective model.

Ideally, data collection and delivery should provide the complete data picture at every scale, from the millisecond resolution of a patient's ECG waveform to patient flow in the command center to population management. However, due to a lack of data resolution, access, and synchronization, this process is wrought with challenges.

Fortunately, software platform services that run on existing hardware have addressed these hurdles and can provide a more complete data picture. Via the devices themselves or through middleware systems, patient data can be automatically collected and then transformed into actionable information to improve clinical care and quality. These platform services can deploy automated event detectors that can listen to particular data streams, record events, and notify clinicians. Each event detector can be built with a common framework and deployed in parallel to provide maximum coverage throughout a healthcare facility.

In this article, we describe three detectors that were built using real-life patient data to solve clinical and quality problems in healthcare facilities. All three detectors were built on the same framework and use the same components. The framework and components needed to build and deploy an automated event detector, as well as lessons learned, are also discussed.

Building an Automated Event Detector

Building an automated event detector requires a variety of stakeholders providing insight into clinical or operational challenges, as well as experts who can architect and validate it. This group can include clinicians, researchers, biomedical engineers, and information technology (IT) specialists.
A five-step process that has been leveraged by key stakeholders to build an automated event detector is described below. Although this process can be performed anew each time, it can be greatly simplified by enabling a framework that provides data integration/normalization, analytics, and visualization. This framework, which allows for faster validation and easier deployment of new detectors, is described more fully later in the article.

The five steps, and the primary stakeholders responsible for each step, are as follows:

1. Define the clinical event (clinicians)
2. Find example events in data (clinicians and researchers)
3. Leverage analytics to build and test rulesets (researchers and biomedical engineers)
4. Deploy rulesets on live streaming data (biomedical engineers or IT specialists)
5. Transform the event detector into a visual form for clinicians to use at the bedside or remotely, depending on where they are and why they need it (biomedical engineers or IT specialists)

The detector can be enhanced by further delineating the clinical event, adding more signals, and iterating over rulesets. With each change, the ever-expanding dataset must be revalidated and tested.

The following real-life case examples demonstrate the effectiveness of this five-step approach in building an automated event detector.

Case Example 1:
Ventilator Management

Invasive mechanical ventilation is a serious intervention. Knowing exactly how long the patient has been on a ventilator is essential for a variety of reasons. For example, the longer the patient is on a ventilator, the higher the risk of so-called “ventilator-associated events,” such as pneumonia. However, an audit by the Office of Inspector General for the Department of Health & Human Services showed an error rate in excess of 95% in documenting when invasive mechanical ventilators were applied to patients.6

Without definitely knowing how long any particular patient has been on a ventilator, it is difficult to know how effectively a healthcare team is detecting markers and trending measured parameters, such as readiness for extubation (when patients are ready to have the ventilator removed and breathe on their own) or changes in airway resistance or lung compliance, which may signal improvements or deterioration in the patient’s clinical condition. Relying on clinician intuition for determining when mechanical ventilation should or can be safely discontinued is not optimal. In addition, creating policies surrounding proper management of mechanical ventilation can be burdensome.

The objective is to initiate mechanical ventilation in a timely fashion when warranted and, in an effort to prevent ventilator-associated respiratory infections and other risks associated with mechanical ventilation, to discontinue it as soon as it is safe to do so.

Beyond clinical considerations, any hospital without complete documentation may be at risk when trying to collect reimbursement if it needs to prove when the ventilator was on or off. If a patient is billed for ventilator usage and challenges the charge, then the hospital, in the absence of thorough and accurate documentation, may be liable for the charges.

With the vast number of ventilator cases per year and the cost per day of mechanical ventilation, this problem could amount to billions of dollars—and this is just one small piece of overall hospital care.9 One can begin to imagine the financial consequences facing an institution from a documentation error over the length of stay for one patient—or for all patients during the course of an entire year.

To help alleviate the challenges regarding time on a ventilator, an automated event detector was built using real-world patient data at one institution. Traditionally, the ventilator is turned on to start a patient’s ventilation and turned off when mechanical ventilation is no longer needed. In this case, the automated event detector receives a data stream from the ventilator; from this, one can determine when the ventilator is on or off. Interpretation of the timestamps associated with these signals can be pushed out via various views.

The automated event detector receives a data stream from the ventilator; from this, one can determine when the ventilator is on or off. Interpretation of the timestamps associated with these signals can be pushed out via various views.
Case Example 2: Detecting Neonatal Nonobstructive Apnea

Babies in NICUs diagnosed with apnea cannot be discharged without specific safeguards in place. Policies vary from hospital to hospital, but generally, an infant who has had a recent apneic event must remain in the NICU for a few additional days before being discharged. This helps prevent the discharge of an infant who is at risk for experiencing an apneic event, and potentially death, postdischarge due to a hard-to-detect problem. The challenge is to accurately detect apneas that are actually occurring without false alarms unnecessarily keeping the patient hospitalized beyond what is clinically necessary.

Accurately detecting apnea is difficult for existing monitors because leads are in close proximity to an infant’s body, which can cause the cardiac and respiratory signals to interfere with one another. Trained medical staff (e.g., nurses, physicians, respiratory therapists) are adept at detecting apnea events at the bedside; however, when clinicians are attending to more than one patient, the level of vigilance required can be difficult to maintain.

An expert panel at the University of Virginia worked on an event detector for nonobstructive apnea. In nonobstructive apnea, a patient stops breathing but not as a result of an obstructed airway. The panel members first determined a gold standard for a neonatal apnea event. They then screened events across medical records for apnea as example events, labeling the data as appropriate.

The next step was programming the computer to recognize these events automatically; however, the raw monitoring signals contained artifacts from the heart rate. To manage this signal quality issue, advanced signal processing techniques were used to remove the heart rate artifact. This allowed clinical nonobstructive apneas to be recognized with greater accuracy. The group then was able to demonstrate that this new event detector could perform better than the monitors themselves.

Ultimately, this type of event detector has the potential to not only improve documentation but also to generate a smart alert that can be transmitted to an end device (i.e., phone, pager, other type of monitor). The alert goes directly to the team member(s) caring for the patient so they can intervene immediately.

Case Example 3: Predicting Cardiac Arrest in a Subset of Patients

The previous examples involved detecting events from a monitor or observing difficult-to-detect events with a patient. Compared with those cases, the following example is the most complex because it involves predicting or recognizing the precursors to an event—in this case, cardiac arrest in patients with hypoplastic left heart syndrome (HLHS). These patients are at risk of deadly cardiac arrest between the two surgeries needed to ameliorate the condition.

This is an interesting problem because in addition to predicting arrest, it also involves predicting arrest in a particular subset of patients. HLHS patients cannot be defined as “normal.” Their cardiac signals are...
different from the rest of the patient population. Only a clinician specially trained in HLHS is likely to be able to detect if one patient’s set of signals is considered sufficiently “abnormal” to be alarming.

The same five-step process used for the ventilation recorder and apnea detector also can be applied in this case. Experts determined the rate and nature of cardiac arrest in a set of patients. The signals from the bedside monitors were sent to the event detector to be transformed into a predictor of arrest (Figure 2). In this case, a statistical model was created and validated. If the transformed data indicate that an arrest is imminent, an alert goes out prior to the event so that the care team can make proactive decisions and preemptively intervene on behalf of the patient.

By observing these backup, or justifying, signals, clinicians can and quickly determine how a given patient’s condition compares with other patients with HLHS. This information can be sent by the event detector to a number of different locations (e.g., the patient monitor, care team members’ pagers and phones). The data also are sent to the EHR flow sheet.

Of note, the automated event detector described in this example is not yet commercially available. At the time this article was written, it was undergoing validation via a multisite study and being evaluated by the Food and Drug Administration as a new Class II medical device. Additional information about the algorithm behind this event detector can be found in the article by Rusin et al. (also see sidebar on p. 36).

**Framework for Building Automated Event Detectors**

To reduce risk without increasing care team burden, hospitals can develop automated event detectors to efficiently leverage all data and send accurate documentation of events to the right sources at the right time. Detectors can be implemented through external software platform services or with a homegrown system that fits the framework described here. This framework has three main capabilities: data integration, analytics, and visualization.

**Data Integration and Normalization**

A variety of sources within the health information technology infrastructure (e.g., bedside biomedical devices, labs, medications) produce patient data. Additional data sources include staff assignments, alarm notifications, and messages from middleware systems. Although these data pass back and forth through a variety of streams, most data tend to be siloed. To optimize the effectiveness of an automated event detector, it must be able to access these data streams.

Accessing data from devices is challenging. Ideally, devices have a protocol for reading data through either a data bridge or some type of application programming interface (API). To allow for data capture and transfer from a single device, many manufacturers build this capability into their devices. However, considering that manufac-

---

**Figure 2.** The score created by the data transformation appears at the top of the image; this score indicates the likelihood of cardiac arrest. In this case, the hypoplastic left heart syndrome (HLHS) patient is 9.3 times more likely to have an arrest than a normal patient. The signals that help justify this value appear in the middle and lower parts of the image. Clinicians who may not be accustomed to the data transformation can understand the source of the risk score by looking at these more traditional signals coming from the monitor. A comparison of the patient’s signals with the general HLHS population’s signals appears on the left of the image. Abbreviations used: ABP, arterial blood pressure; HR, heart rate; LAP, left atrial pressure; RR, respiratory rate; RSO2, regional cerebral oxygen saturation; SPO2, peripheral capillary oxygen saturation.
turers tend to work in isolation in developing protocols for capturing and transferring data for given devices, the challenge of normalizing data across different manufacturers’ devices quickly becomes apparent.

Algorithms that have been “trained” on a certain type of data can only run effectively on similar data. As a result, the collected data must be normalized before an algorithm can be trained and deployed. Otherwise, a situation may arise where an algorithm cannot be reused. For example, if a device is upgraded or modified, the algorithm will fail unless it is retrained to process the new data streams. However, if data are normalized before algorithmic transformation occurs, then the algorithms will be resilient to upstream device changes.

The normalization of data involves three steps. First, all “like data” (e.g., ECG waveforms, low SpO2 [peripheral capillary oxygen saturation] alarms) are grouped together. Second, these data streams are time synchronized so that a given measurement can be compared with another measurement based on a known “master clock.” Third, the grouped, time-synchronized data are stored so that the algorithm can retrieve them.

The first two steps are challenging given the access issues described above—each device is different. Grouping like data is difficult because data streams from different manufacturers’ devices do not match, either in the set of measurements taken or in their nomenclatures. For example, alarms can vary from one manufacturer’s monitor to that of another manufacturer. Time synchronizing data is challenging because it isn’t clear how to reconcile one set of timestamps with another.

These problems can be amplified if different units in a hospital are using monitors from different manufacturers.

These challenges can be addressed in a few ways. First, device manufacturers can agree on a standard set of data streams to allow for the aggregation of the most commonly used measurements. For example, one manufacturer might reference a standard list of signals in its monitors. However, this list does not apply to every possible signal and does not address time synchronization. Second, third-party device integrators can be used to collect and time synchronize data, but installing these systems can be expensive.

Third, device manufacturers can “open” their data with APIs and other tools that allow data to be captured and processed with specific programs written and maintained by biomedical staff or clinical IT specialists. This solution may prove laborious, as adding more devices requires additional code to be written and maintained.

Analytics
The analytics capability requires two needs to be considered. The first is that a toolset is needed to interrogate the data and develop event detectors. For example, rule sets for detecting events must first be built and then validated at scale. (This process is described in more detail below.) The second need is to transform data to perform event detection so the detector can be deployed in a real-time clinical setting.

Visualization
Although detecting events in the background is sufficient for documentation, more is required for active clinical care. Clinicians need to have context and information at their fingertips to take action. Furthermore, analytics must be delivered to clinical team members in a way that fits their workflow. This is critical for any technical solution but especially for automated event detection. The capability should exist for delivering the information in multiple ways (e.g., to an EHR, quality improvement report, middleware system, and/or an endpoint device such as a personal computer, tablet, or phone). For example, a patient risk monitor could send alerts directly to the care team through the use of a pager or smartphone. The visualization layer should push the results of the analytics and intuitive event detection to the care team when and where they need it so immediate action can be taken.

Discussion
The automated event detectors described in the case examples provided a variety of benefits. The ventilation management detector showed that a detector does not have to be sophisticated to be impactful. A simple model can have a great effect, with...
the added benefit of being easy to explain and justify.

The apnea detector showed that detectors can be used to reduce alarm fatigue—a growing concern in healthcare facilities. The detector provides more true alarms and fewer false alarms, reducing alarm noise in the unit. A suite of these detectors tailored for the needs of a particular unit can help reduce alarm fatigue substantially.

The main benefit of the cardiac arrest predictor is providing an avenue toward proactive care. Furthermore, the detector and monitor provide deeper insight into patient epidemiology that did not exist previously. Over time, clinicians can refine this detector to provide deeper insights.

The three event detectors had similar framework-based barriers to adoption and implementation. After an event detector is developed and working, buy-in is straightforward. Because of the siloing of data from devices and barriers that prevent clinicians from accessing and extracting data, the challenge lies in establishing the framework needed to train detectors.

Stakeholder buy-in and technical integration are necessary for facilitating the adoption and implementation of automated event detectors. Clinicians must want to have these types of detectors in their facility, and biomedical engineering and IT specialists require the freedom to build out the technical components of the hospital for maximum connectivity. We have found that clinicians have an abundance of ideas for improving patient care in their units. Linking these clinicians with interested technical staff provides the support needed for project approval.

Developing an effective algorithm requires clinicians and biomedical engineers to work closely throughout the process, from building to deployment. Clinicians generally have three main considerations. First, algorithms need to provide as many true alerts as possible while minimizing the number of false alerts. Second, the alerts and associated information need to be presented in an intuitive manner. Finally, the justification for the alert needs to make clinical sense.

Biomedical staff need to provide technical support in order to ensure that the events detected are clinically relevant and that the events that are eventually found can be trusted as real and understood quickly. Working together, these two groups can ensure that an effective algorithm is built and deployed properly for effective use.

**Conclusion**

Recent technological advances are allowing the convergence of data, analytics, and expertise to be realized in healthcare facilities. New sets of tools and opportunities exist for hospitals to improve both operational and clinical workflows at an unprecedented rate. These tools allow for iterative, measurable improvements tailored to a particular hospital’s needs. One of these tools is the capability of building and deploying automated event detectors.

Three key factors can expedite the development of automated event detectors: 1) data integration/normalization, 2) analytics, and 3) visualization. After these are enabled, automated event detectors can be applied to nearly any type of event for a given patient population by following the five-step process described here.

The authors recommend that hospital biomedical and analytics teams follow this five-step process to realize the benefits of automated event detection—benefits that include mitigated risk, increased accuracy of documentation, and, most importantly, improved delivery of care.

**References**


Don’t be the victim of a cybersecurity disaster.

Learn how to implement a security risk management process with TIR57.

Order today at www.aami.org/store
Use product code: TIR57 or TIR57-PDF

Not a member? Join AAMI at www.aami.org/membership.