5-2011

Leveraging complex event processing for smart hospitals using RFID

Wen YAO
Pennsylvania State University - Main Campus

Chao-Hsien CHU
Singapore Management University, chchu@smu.edu.sg

Zang LI
Pennsylvania State University - Main Campus

DOI: https://doi.org/10.1016/j.jnca.2010.04.020

Published in Journal of Network and Computer Applications, 2011, May, Volume 34, Issue 3, pp. 799-810.
http://doi.org/10.1016/j.jnca.2010.04.020

Leveraging complex event processing for smart hospitals using RFID

Wen Yao [a], Chao-Hsien Chu [b,*], Zang Li [a]

[a] College of Information Sciences and Technology, The Pennsylvania State University, 329B, IST Building, University Park, PA 16802, USA
[b] College of Information Sciences and Technology, The Pennsylvania State University, 316H, IST Building, University Park, PA 16802, USA

A B S T R A C T

RFID technology has been examined in healthcare to support a variety of applications such as patient identification and monitoring, asset tracking, and patient–drug compliance. However, managing the large volume of RFID data and understanding them in the medical context present new challenges. One effective solution for dealing with these challenges is complex event processing (CEP), which can extract meaningful events for context-aware applications. In this paper, we propose a CEP framework to model surgical events and critical situations in an RFID-enabled hospital. We have implemented a prototype system with the proposed approach for surgical management and conducted performance evaluations to test its scalability and capability. Our study provides a feasible solution to improve patient safety and operational efficiency for an RFID-enabled hospital, by providing sense and response capability to detect medically significant events.

Keywords:
Complex event processing
RFID
Smart hospital
Healthcare
Surgery

1. Introduction

The pressures of improving patient safety while reducing operational costs for healthcare are forcing hospitals to adopt new information technologies so as to reduce medical errors and respond quickly to critical situations. Radio frequency identification (RFID) is a rapidly developing technology and it is believed to be the next generation innovation for automatic data collection, object identification, and asset tracking. Although RFID is still in its infancy in healthcare applications, it has gained much attention during recent years from both service providers and technology vendors. Some pioneering hospitals have implemented this technology to identify and monitor patients, track assets and medical supplies, and check patient–drug compliance. A survey conducted by BearingPoint (2007) and the National Alliance for Health Information Technology on participants of more than 300 government and healthcare executives, indicates that RFID is "poised for growth in healthcare".

With the capability to capture the identity and location of any tagged object automatically and periodically, RFID data in hospitals can be in huge amount. Besides, an increasing number of embedded devices in hospitals, such as physiological sensors and environmental sensors, emit data in real time. Consequently, a hospital needs to handle a large amount of data from a variety of sources and detect medically significant events timely by correlating RFID and non-RFID data. However, RFID raw data only provides low-level information such as Electronic Product Codes (EPC) of the tagged objects, location and timestamp, which are not directly related to business processes. Physiological events such as patient body temperature and blood pressure are also in low level. Other data such as patient medical record needs to be correlated to signify actionable information for decision making. Therefore, the ability to transform raw data in health care practices into useful knowledge in order to realize the maximum value from RFID technology becomes a critical issue.

Complex event processing (CEP) (Luckham, 2002) provides an effective solution to process event streams in real time for today's dynamic business environment. Compared to the delayed-analysis methods used traditionally in relational databases, CEP involves continuous processing and analysis of high-volume and high-speed data streams such as RFID data. It also correlates distributed data to detect and respond to business-critical situations in real time. Thus, CEP helps to deal with a variety of data streams to deliver actionable information. For example, in the case of patient identification in a surgery, if a wrong patient is taken to the surgery room mounted with an RFID reader, an alert will be triggered and sent to the care provider immediately. Therefore, leveraging CEP to manage hospital events that are captured by RFID systems and embedded devices for situation detection can be helpful to solve the challenges faced by healthcare.

In this paper, we propose an RFID-enabled CEP framework for managing hospital data from a variety of sources, specifically for surgical procedures. We apply the logic of CEP to model basic events and event patterns in hospitals to detect medically

* Corresponding author. Tel.: +1 814 865 4446; fax: +1 814 865 6426.
E-mail addresses: wxy119@psu.edu (W. Yao), chc4@psu.edu (C.-H. Chu), zul110@psu.edu (Z. Li).

significant events. To verify the feasibility of our approach, we have developed a prototype system that leverages CEP to provide critical alerts for healthcare professionals. This paper aims to identify potential challenges that affect the quality of healthcare services and seeks a viable approach to improve patient safety. This study helps to answer the following questions: (1) what are the major challenges faced in hospitals, especially in surgical procedures, (2) how can RFID help to address these challenges? (3) what is CEP and how can CEP be used to handle a large number of RFID events and other medical events? and (4) what are the expected benefits of using RFID-enabled CEP framework in hospitals?

This paper is organized as follows. In Section 2, we describe background information and an application scenario. Then we present CEP preliminaries in Section 3, including events, event operators, and event rules. Section 4 describes the modeling of hospital events and patterns. The implementation and evaluation of the proposed RFID-enabled CEP prototype is described in Section 5, where system architecture and performance evaluation are presented. In Section 6, we discuss the contributions and limitations of our work, followed by the related work in Section 7. Section 8 outlines our conclusion and future work.

2. Background and motivation

2.1. An application scenario

The hospital is a large, extremely busy, and chaotic environment where hundreds of medical cases are treated every day. Hundreds of doctors and staffs are walking constantly; inpatients and outpatients are moving around under surveillance; medical devices are commonly recalled for emergent use. Although information systems have been used to support the work of medical professionals, they are not able to accurately track patient flow and asset flow in real time. Thus, sensing and responding quickly to critical situations becomes impossible in this dynamic and unpredictable environment. In particular, performing a surgery requires extensive information sharing, coordination, emergent situations detection, and immediate reactions. Achieving these requirements is even difficult if no proper technology is used to help monitor patient flows, track medical devices, and alert unexpected situations. Failure in doing these would threaten patient safety, decrease operational efficiency, and increase medical costs.

We use a typical surgical workflow (Fig. 1, adapted from Su and Chou (2008)) to illustrate the challenges in performing a surgery. This workflow generally involves three phases, namely preoperative, intraoperative, and postoperative. Five groups of participants are involved in this procedure, including patients (P1), transporters (P2), nurses (P3), anesthetists (P4), and surgeons (P5). Surgical related locations can include ward (L1), holding area of operating suit (L2), operating suit (L3), holding area of operating room (L4), operating room (OR) (L5), recovery room (L6), and intensive care unit (ICU) (L7).

In the preoperative stage, the transporter brings the scheduled patient along with related documents from the ward to the OR suit. Before the patient is admitted into the OR suit, the nurse verbally confirm the patient's identity and surgical information. Besides, the nurse reviews the patient's medical record such as medication, vital signs and tests, to determine her readiness. Then the patient stays in the holding area of OR until the scheduled OR is ready. However, before the patient is admitted into the OR, the nurse has to check her identity again. Before the surgery begins, the anesthetist verbally confirms the patient's identity, medical information (e.g., allergies, medical history), and the scheduled surgery. Then the anesthetist reviews her vital signs and test results, and gives the most suitable anesthesia. After anesthesia, the patient is ready for the operation but the surgeon also checks if this is the right patient and site of body for the operation. During the surgery, the surgeon and nurses focus on the operation and if any emergency happens, they can hardly know the personal information of the patient, since all the information is documented in paperwork. After the operation, the nurse takes the patient

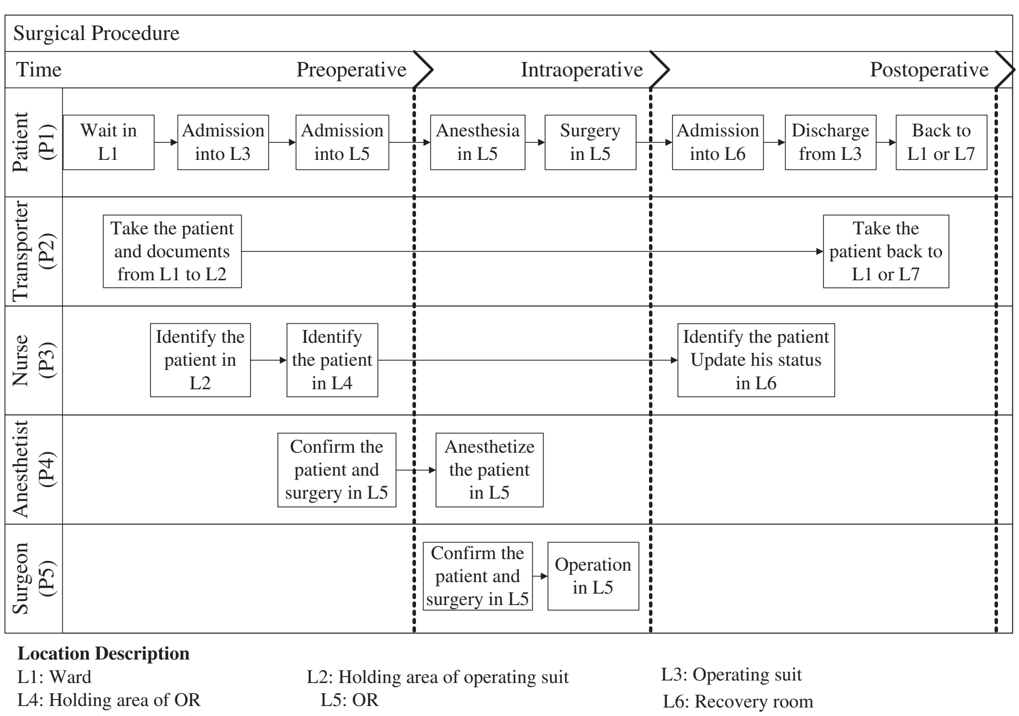

Fig. 1. A typical surgical workflow.

to recovery room within OR suit for awakening from anesthesia. At last, the patient is discharged from OR suit and transferred to the original ward or ICU, depending on her medical conditions. During this procedure, the status of the patient and the OR are both updated manually.

From the above surgical procedure, we can identify following challenges that may threaten patient safety. First, identifying patient manually at different stages by different hospital personnel is error-prone and time consuming. Since nurses are usually very busy dealing with a number of cases per day, they might forget to confirm the patient's identity in some cases. Besides, the surgeon has to verify the patient by face recognition if she is anesthetized. As a result, possible human mistakes can cause wrong patient, wrong OR, and wrong procedure for a surgery (Sandlin, 2005). Second, inability to access patient electrical health record (EHR) can cause medical errors or delay in handling emergency. Incomplete knowledge about patient medical history can cause wrong anesthesia. During the surgery, although patient monitors can show the patient's vital signs in real time, these signals should be interpreted better along with EHR (e.g., history, lab tests, medication). Third, limited resources in hospitals make high recall rate of medical devices, but tracking a device manually is almost impossible. For example, a patient encounters a heart attack in a sudden and needs an infusion pump immediately. Nurses usually have to look several places before they can find the device and this always causes delay in the treatment. Fourth, hospital staff members usually have too much workload so they might make mistakes. Surgeons may leave sponges inside the patient body if manually counting the sponge number at the beginning and the end of a surgery. Even more, they may leave scissors or other small instruments carelessly inside the patient body. Last, improper disposal of used instruments can cause waste. For instance, some reusable instruments are thrown away by housekeepers if they do not pay enough attention.

Clearly, a technology like RFID is in urgent need to automatically identify and track objects in hospitals. More importantly, hospitals should be able to sense and respond to critical situations in real time. These motivate us to propose an RFID-enabled CEP framework for modeling surgical events and respond to medically significant events in real time.

2.2. An RFID-enabled smart hospital

RFID uses radio waves to transfer data between readers and tagged objects. It is automatic, fast, and does not require line of sight for communications between readers and tags. With the capability of automatic data identification and collection, RFID technology can be used in hospitals to identify and locate patients, equipment, and medical instruments. Moreover, it has the potential to significantly improve operations by actively monitoring patient and asset flow through the hospital and enabling this data to be analyzed for process improvement. For example, St. Vincent's Hospital in Birmingham deployed RFID to track patient flow through its radiology and labs to improve patient flow (Krohn, 2005).

An RFID-enabled smart hospital is configured with pervasive RFID devices. All patients are equipped with personalized bracelets with embedded RFID tags and patients with high risk are equipped with biomedical sensors that take physical measures automatically. Medical equipment and instruments are also RFID tagged. Besides, doctors are equipped with mobile devices to track tagged items and are able to receive alerts immediately in case of emergency. The hospital is equipped with RFID readers in different locations to communicate with tags. In addition, other embedded sensors are also available for providing important data. For instance, the temperature and humidity sensors installed in operating rooms offer an optimal environment for surgeries. Patient monitoring devices such as pulse oximeters and anesthesia machines continuously provide streams of patient vital signs.

3. CEP preliminaries

An RFID-enabled smart hospital can generate a variety of data streams that are in different formats and need to be processed timely. For example, an RFID tracking system consistently generates data about the location and time of tagged items, which is in low-level semantics and not directly useful. Besides, embedded sensors and devices continuously generate environmental or medical related data. Complex event processing (CEP) has been introduced to process and correlate these data. This technique aims at processing multiple streams of data continuously and identifying meaningful events in real-time. CEP has several features. First, it can extract basic events from a large amount of data and correlate them to create business events according to user-defined rules. Second, the patterns to correlate events can include logical, temporal, and causal constructors. Third, CEP can react to critical situations in real time. In this section, we formalize the definition of events, event constructors, and CEP rules. The relationship of the concepts in CEP is illustrated by the ontology in Fig. 2.

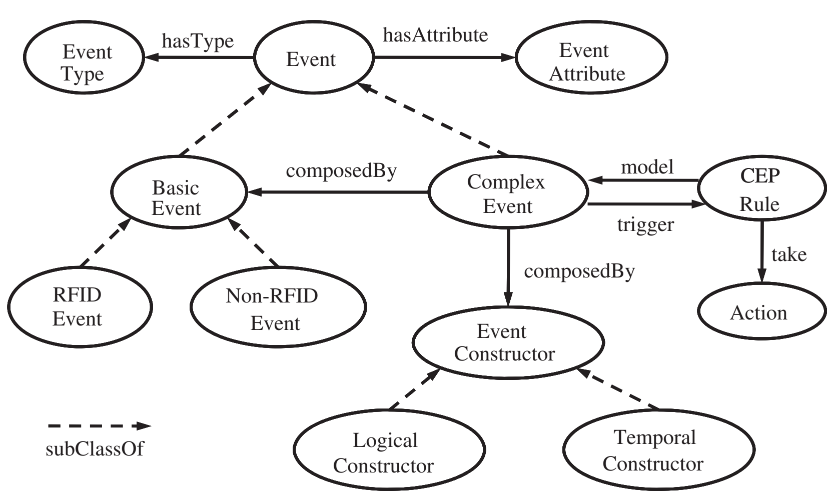

Fig. 2. CEP ontology.

3.1. Event

An *event* can be defined as a record of an activity in a system for the purpose of computer processing (Luckham, 2002), or an occurrence of interest in time (Wang et al., 2009). In general, events can be categorized into *basic event* and *complex event* (or *composite event*). We use upper case and lower case, such as E and e, to represent event type and event instance, respectively.

Definition 1 (Basic event). A basic event can be denoted as $E{=}E$ (id, a, t), where *id* is the unique ID of an event, $a{=}\{a_1, a_2, \ldots a_m\}$, $m > 0$, is a set of event attributes and t is the event occurrence time. A basic event is atomic, indivisible, and occurs at a point sin time.

Definition 2 (RFID event). An RFID event can be denoted as $E{=}e$ (o, r, t), where o is the tag EPC, r is the reader ID that captures this tag, and t represents the observation timestamp. Although the time of RFID reading might be earlier than the time when the event is captured, we assume this difference is too small to be recognized. An RFID event is a basic event.

Definition 3 (Complex event). A complex event can be defined as $E{=}E$ (id, a, c, t_b, t_e), $t_e >= t_b$, where $c{=}\{e_1, e_2, \ldots, e_n\}$, $n > 0$, is the vector that contains the basic events and complex events that cause this event happen, t_b and t_e are the starting and ending times of this complex event. It can happen over a period of time (i.e., from t_b to t_e).

A complex event is aggregated from basic events or complex events using a specific set of event constructors such as disjunction, conjunction, and sequence that are explained in the next section. It signifies or refers a set of other events to indicate a situation described in the application scenario. Complex events contain more semantic meaning and are more useful for decision making in business applications.

3.2. Event constructors

Event constructors or *event operators* are used to express the relationships among events and correlate events to form complex events. Wang et al. (2006) gives a comprehensive set of event constructors and classify them into temporal and non-temporal constructors. In Table 1, we adapt and extend these event constructors and list the most frequently used ones for complex event detection.

These event constructors can be used to define event patterns that catch meaningful information from real-time data streams. For example, the pattern within $(E_1 \wedge E_2, 5\ \mathrm{s})$ matches when both events E_1 and E_2 occur within time interval less than 5 seconds; the pattern $(E_1 \wedge \neg E_2)$ matches when event E_1 occurs but event E_2 does not occur; the pattern $(E_1^*;\ E_2)$ matches when every occurrence of event E_1 is followed by event E_2.

3.3. CEP rules

Based on the formalization of events and event constructors described above, CEP rules are defined to specify domain syntax and semantics. A rule is the predefined inference logic or pattern for detecting complex events. Several studies (Zang et al., 2008; Wang et al., 2009) have described different syntax for CEP rules. We use ECA (event–condition–action)-like rule expression language to describe event patterns since it is easier to use and more understandable. The generic syntax of ECA can be expressed as follows:

Rule *rule_id, rule_name, rule_group, priority*
  ON *event*
  IF *condition*
  THEN *action1*, *action 2*, … *action n*
END

where *rule_id* and *rule_name* are unique for each rule, suggesting the id and name for a rule; *rule_group* is a group of semantically related rules; *priority* defines the priority of this rule; *event* specifies the event of interested; *condition* is a boolean combination of user-defined functions; *action* defines a user defined procedure (e.g., to send out alarms) or an update in the database (e.g., update of patient status). With CEP rules, we can provide sufficient support for processing RFID and other sensor data, such semantic data filtering and real-time monitoring.

4. Modeling events in an RFID-enabled smart hospital

A smart hospital enabled by RFID can track movements of doctors, patients, and objects carrying RFID tags. With RFID technology, it is possible to create a physically linked world in which every object is identified, cataloged, and tracked (Wang et al., 2006). To achieve these advantages, the first task for an RFID application is to map objects and their behaviors in the physical world into their virtual counterparts by semantically interpreting and transforming data from RFID systems and other sensors. We use the CEP techniques described above to model complex events in hospitals, such as medical related activities and emergencies. Since most studies focus on the correlation of events in supply chain (Wang et al., 2009), such as the aggregation of events based on containment relationships, we argue that event patterns in hospitals can be quite different from those in supply chains. For

Table 1
Expression and semantics of event constructors.

Type	Constructor	Expression	Meaning
Logical (non-temporal)	AND ($\wedge$)	$E_1 \wedge E_2$	Conjunction of two events E_1 and E_2 without occurrence order
	OR ($\vee$)	$E_1 \vee E_2$	Disjunction of two events E_1 and E_2 without occurrence order
	NOT ($\neg$)	$\neg E_1$	Negation of E_1
Temporal	sequence (;)	$(E_1; E_2)$	*E1* occurs followed by E_2
	window ( )	*window*(E_1,t)	Event E_1 occurs for time period t (s, m, h)[s: second, m: minute, h: hour]
		window(E_1, n)	Event E_1 occurs n times ($n > 0$)
	within ( )	*within*(E_1, t)	Event E_1 occurs within less than t
		within(E_1,t_1, t_2)	Event E_1 occurs within interval t_1 and t_2
	at ( )	*at* (E_1, t)	Event E_1 occurs at time t [system time]
	every (*)	E_1^*	Every occurrence of E_1
	during( )	*During*(E_1, E_2)	Event E_2 occurs during event E_1

example, hospitals are interested in resources used for a surgery and all the personnel performing this surgery. The CEP rule is an effective mechanism to filter basic events and extract meaningful information, in order to identify medically significant events.

4.1. Location transformation

RFID readings can imply object movements and location change, which is the basis to identify activities in a clinical workflow and detect medically significant events. For example, when an RFID observation indicates a wrong patient is taken into an OR, a mismatch between the patient and the OR can be detected. In response to the *wrong patient* event, the medical staff is automatically and instantly warned of this mismatch. In addition, location transformation should be recorded for historical data analysis, since patient and asset flows can be traced by a sequence of location changes. Table 2 lists all the complex events related to the location change of objects. An object can be any RFID-tagged entity such as a person, a device, or a bottle of medicine. Based on these events, we can infer the current location of an object, and the period during which the object stays in a location.

4.2. RFID semantic data filtering

Two types of data filtering should be considered for RFID data before they can be further processed: low level data filtering and semantic data filtering (Wang et al., 2009). We assume that incoming RFID data has already been filtered with enough quality by the middleware since most middleware provides the functionality of low-level data filtering and aggregation (e.g., Alien and Symbol). Thus, we only consider rules that perform semantic data filtering in a smart hospital. For example, although a medicine bottle is always being detected present in a smart cabinet which is mounted with an RFID reader, we are only interested when it is put into the cabinet, and when it is taken out of the cabinet, in order to automatically update the status of this medicine bottle and the person performing this action. If an unauthorized person is moving a medicine bottle, medical staff will be alerted automatically and immediately. Table 3 captures complex events involving semantic data filtering. Other complex events can be derived from these as well.

4.3. RFID real-time monitoring

CEP rules can provide effective support for real-time monitoring of RFID-tagged objects, especially medical devices and patients. It is well known that hospitals own a great number of expensive medical equipment and part of them are stolen on a regular basis (Fuhrer and Guinard, 2006). RFID can improve theft prevention by tracking equipment to reduce severe consequences caused by the lack of vital equipment. If a reader at the building exit detects a piece of tagged equipment without detecting an authorized user, then it implies the equipment is being taken out illegally, and an alert is sent to a hospital security personnel. Another common application is patient tracking before a surgery. If a reader r mounted at the OR door detects a tagged patient who is not authorized to have a surgery within 45 min from current time (CT), then an alarm is triggered to inform this mismatch. This rule can be represented as follows:

Rule *R1*, *patient_identification*
  ON *within (e (p_epc, r, t)* ∧ *type (p_epc*=“*patient*”, 10 *s)*
  IF NOT (SELECT * from *SURGERY*
    WHERE *patient_epc*=*p_epc AND location_epc*=*r AND*
      CT ≤ *scheduled_time* ≤ *CT*+45 *min*)
  THEN *trigger_alarm*
END

4.4. Patient monitoring

In addition to the RFID event stream in hospitals, patient monitoring systems continuously track patient physiological data. For example, vital signs monitors can track heart rate and blood pressure; pulse oximeter monitors the blood oxygen saturation levels of patients. The value of an individual physiological parameter is in low level and generally does not provide much semantic meaning in terms of the patient status. To detect medical situations, CEP rules are used to correlate various physiological events with temporal reasoning. Patient medical records may be combined to trigger alarms since patients have

Table 2
Common RFID location change events.

Complex event	Expression
E_1: Object o_1 enters place l[a]	E_1=within ($\neg e(o_1, r, t_1)$; $e(o_1, r, t_2)$, 10 s[b])
E_2: Object o_1 leaves place l	E_2=within ($e(o_1, r, t_1)$; $\neg e(o_1, r, t_2)$, 10 s)
E_3: Object o_1 moves from l_1 to l_2 ($l_1 \neq l_2$)	E_3=within ($e(o_1, r_1, t_1)$; $e(o, r_2, t_2)$, 10 s)

[a] Assume location l is mounted with RFID reader r, l_1 with reader r_1, and l_2 with reader r_2 (same for Table 3).
[b] Assume that the readers are scheduled to bulk-read all objects every 10 seconds (same for Table 3).

Table 3
Common RFID semantic filtering events.

Complex event	Expression
E_4: Object o_1 enters proximity of object o_2	E_4=within ($E_1 \wedge$ e(o_2, r, t_3), 10 s)
E_5: Object o_1 is touching/next to object o_2	E_5=within ($e(o_1, r, t_1) \wedge e(o_2$, r, t_2), 10 s)
E_6: Object o_1 leaves proximity of object o_2	E_6=within ($E_2 \wedge$ e(o_2, r, t_3), 10 s)
E_7: Object o_1 and object o_2 move to distance d (m) apart	E_7=within (e(o_1, r_1, t_1) ∧ $e(o_2, r_2, t_2)$, 10 s) ∧ dist(r_1, r_2) ≥ d
E_8: Person o_1 put object o_2 to location l_1	E'_5=within ($e(o_1, r_1, t_3) \wedge e(o_2, r_1, t_4)$, 10 s) E_8=within (E_5;E'_5, 10 s) ∧ type(o_1)=“person”
E_9: Person o_1 takes object o_2 away from location l_1	E_9=within (E'_5;E_5, 10 s) ∧ type(o_1)=“person”

different medical backgrounds. In the operating room, if the detection of critical situations is delayed, the patient's life will be threatened.

We use hypovolemia danger detection as an example to illustrate the modeling of patient monitoring. Suppose a patient with hypovolemia history is being operated and his vital signs are being tracked by monitoring devices. If his heat rate increases over 5% and his blood pressure decreases over 6% within a 5 minutes time period, an alarm is sent to the medical staff for action. The rule can be represented as:

Define E_1=*HeartRate (epc, value, t1)*
Define E_2=*BloodPressure (epc, value, t2)*
Define E_3=*observation (epc, r, t3)*
Rule *R*2, hypovolemia_danger
ON (E_1.*epc*=E_2.*epc*=E_3.*epc*)
∧ *type(epc)*=“*patient*” ∧ *type(r)*=“*OR*”
∧ *window (increase(*E_1*.value)* > *5%* ∧ *decrease(*E_2*.value)* > 6%, *5 min)*
IF SELECT * *from MedicalRecord*
WHERE patient_epc=*p_epc AND hypovolemia*=*true*
THEN *send_alarm*
END

4.5. Data aggregation

Hospitals are flooded with massive flux of data from RFID systems and other medical monitors. To avoid data overload and missing important events, CEP rules are used to aggregating data in an automatic fashion. For example, if we detected the presence of correct patient and medical staff in a surgery room and the light of this room is turned on, an aggregated event *surgery begin* can be inferred. As a result, we can update the status of the surgery room and related persons automatically.

5. System implementation and assessment

The RFID-enabled CEP framework is designed to collect basic events from heterogeneous sources and correlate them for situation detection. We have implemented a prototype system that aims to offer sense-and-response capability to a smart hospital so they can react quickly to emergencies, especially for time critical scenarios.

5.1. Architecture

Fig. 3 presents the physical and semantic data flow in an RFID-enabled CEP framework. At the lowest level, raw readings from location tracking systems are captured by RFID readers and then filtered (i.e., smoothing and aggregation) by the middleware to remove noisy and redundant data. The produced RFID events along with data from other embedded sensors or devices are then passed on to the CEP engine for further processing. These events are basic events since they are captured directly from their sources and have not been aggregated. In addition, data from other information systems or database is needed for complex event pattern matching. They are inserted into the working memory as facts. Facts and events from sensors can be correlated by event constructors. CEP rules are stored in the rule base, so that the CEP engine can detect complex events to signify critical situations. As a result, the complex events contain semantic meanings and can be used by applications. For example, if a certain threat to patient safety is identified, an alert will be sent to the care provider.

5.2. CEP engine—Drools 5.0

We selected the open source software Drools 5.0 (Bali, 2009), which include Drools expert and Drools fusion, as our CEP engine, since it provides an integrated platform for modeling rules, events, and processes. Drools expert is a forward chaining inference engine, using an enhanced implementation of RETE algorithm (Forgy and Rete, 1982). To support complex event processing, Drools fusion was developed to support processing of multiple events from an event cloud for event detection, correlation, and abstraction. It has several advantages. First, it supports asynchronous multi-thread streams in which events may arrive at any time and from multiple sources. Second, since temporal reasoning is an essential part of CEP, we examined the capability of Drools fusion in support of temporal relationships. It

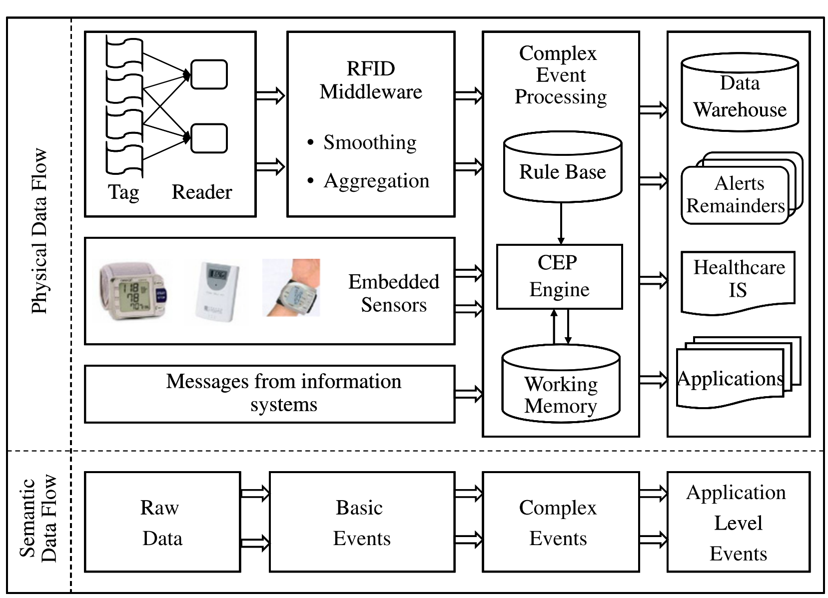

Fig. 3. Physical and semantic data flow in RFID-enabled hospital.

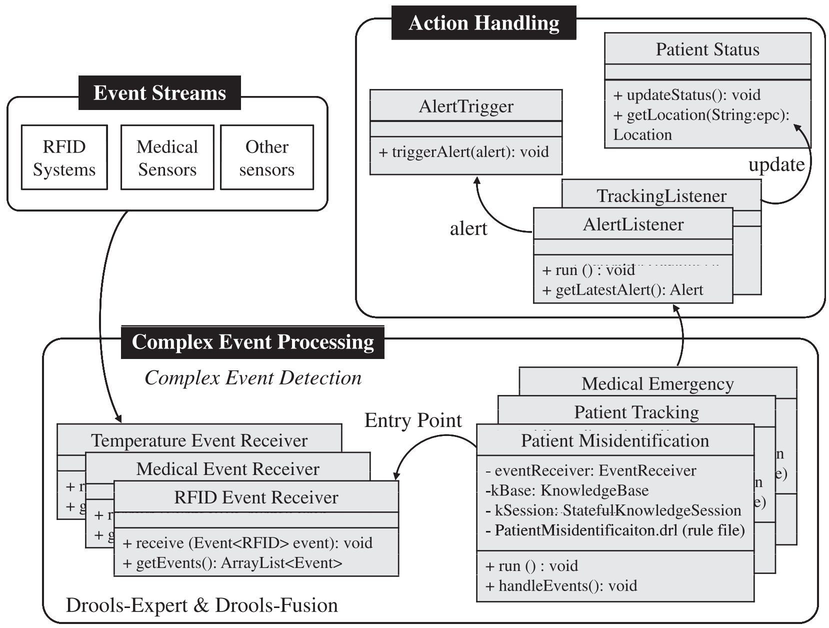

Fig. 4. Walkthrough of event processing implementation.

has a complete set of temporal operators to allow modeling and reasoning over temporal relationships between events. Third, it allows complete and flexible reasoning over the absence of events (i.e., negation). Lastly, aggregation of events over temporal or length-based windows is supported by sliding windows.

Fig. 4 presents an overall walkthrough of how we implemented complex event processing of RFID and medical data with Drools. The *event receivers* are defined as Java classes and they are considered as input adapters. They can be fed with data streams from physical data sources or simulation programs. Event receivers respond to *complex event detectors* (e.g., patient misidentification) with events of interest. An *entry point* is a channel through which the events of interest are asserted into the working memory. In this way, all event streams can be independent of each other and multiple threads can run in parallel. Complex event patterns are defined using CEP rules. Upon detection of predefined patterns, alerts are sent out as messages and actions are triggered. Complex events are also fed back into the engine and treated as incoming events.

5.3. *Temporal reasoning for complex event detection*

Traditionally, the RETE algorithm is used in expert systems for production-based logical reasoning, matching a set of facts against a set of inference rules. The basic idea of RETE algorithm is to construct an acyclic network of rule premises for forward chaining. By default it does not support temporal operators. We are not describing the details of RETE algorithm since we focus on the extension of RETE with temporal reasoning. To support temporal constraints in rules, events can be modeled as facts with timestamps and they are inserted into the working memory at runtime. In addition, events need to be discarded when they are no longer of interest or cannot contribute to complex events. Temporal relationships can be realized by explicitly stating the conditions in the rules (Walzer et al., 2008). For instance, if a complex event E_3 is created when an event E_1 occurs followed by event E_2, we can use the following rule to express this: IF *t_begin*(E_1) < *t_begin*(E_2), THEN *create* E_3. Then this rule can be processed by the traditional rule engine. Other temporal constructors can be realized in the same way.

Fig. 5 presents 13 temporal relationships between events as well as their semantic meanings, including *before* (*after*), *meets* (*metby*), *overlaps* (*overlappedby*), *during* (*includes*), *starts* (*startedby*), *finishes* (*finished by*), and *coincides*. All of them are supported by Drools fusion. Besides, sliding windows are supported for correlating events on temporal or length-based time windows. Our proposed event constructors can be easily supported by these operators. The logical event constructors *AND* ($\wedge$), *OR* ($\vee$), and *NOT* ($\neg$) are supported, respectively, by the operators *and* (&&), *or* (||), and *not*. Temporal constructors are also transformable. For example, *sequence*(;) is often combined with *within* to constraint the time distance between two events. The pattern (within (E_1; E_2), 10 s) can be expressed as E_2 (this after [0 s, 10 s] E_1). Similarly, the rest of temporal constructors can be transformed as well.

5.4. *Performance evaluation*

To test the performance of the proposed CEP-enabled RFID applications, we implemented a prototype in Java language for testing and evaluation. Our testbed is a PC with 2.0 GB of RAM and 1.86 GHz Genuine Intel CPU running Windows XP Professional operating system.

In the prototype, we simulated three event streams to generate basic events. The RFID event stream has four event types, the patient monitoring event stream has eight event types, and the environmental sensor stream has two event types. Each event type has its own attributes and methods to get these attributes. Each event type is used by at least one complex event expression. We used prepared data files to simulate the continuous generation of events instances.

Currently, there are approximately 120 complex event expressions involving different kinds of semantic meanings and complexities. All the complex events defined (i.e., $E_1, E_2, \ldots, E_9$) and illustrated (i.e., events related to RFID monitoring and patient monitoring) in Section 4 are included. Each complex event expression has at least one logical constructor (i.e., *AND*, *OR*, or *NOT*) and one temporal constructor. The most frequently used temporal constructors are *window* and *sequence* (or *after*), both of which are used by more than 60 complex events. On average, each complex event has approximately two logical and two temporal constructors. Based on these complex event expressions, we have

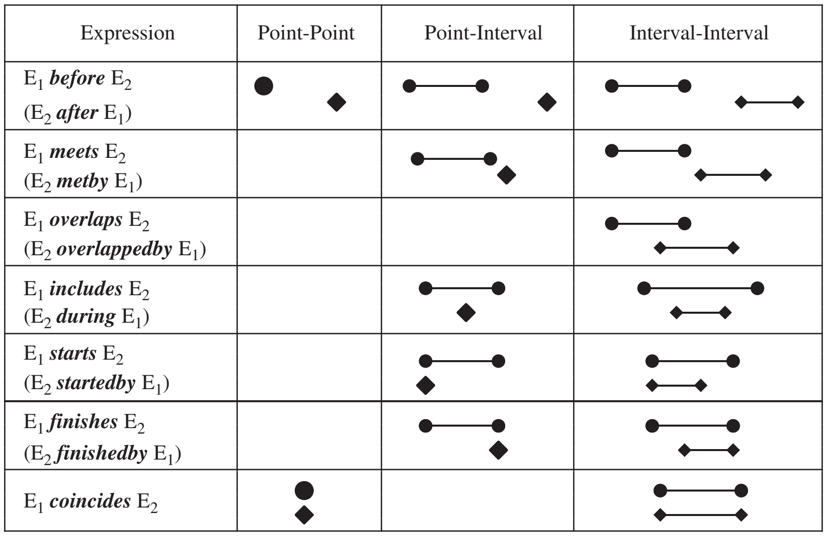

Fig. 5. Temporal constructors in Drools fusion.

defined 41 rules in total for detecting situations. Each rule uses 1–3 complex event expressions and two-thirds of rules involve the correlation of facts, e.g., the medical history of a patient.

We evaluate the scalability and the situation detection ability of our system with the Drool-fusion rule engine. Fig. 6 shows the results in terms of processing time and number of detected complex events. We define *event processing time* as the total time of processing the number of incoming events. In Fig. 6(a), when the number of basic events increases from 2500 to 50,000, the event processing time increases from 4800 to 352,000 ms, if we use 24 rules. However, if we increase the number of rules to 41, the processing time is increased from 5500 to 560,000 ms. Thus, the number of event rules can have significant impact on the performance of event processing time, especially if they involve complicated temporal and logical reasoning. Fig. 6(b) shows the number of complex events that are detected for an increasing number of basic events. Obviously, when the rule number is small (i.e., 24), we only detect fewer complex events (i.e., from 38 to 1582). When the rule number increases to 41, we can detect 44 complex events by 2500 basic events and 2577 complex events by 50,000 basic events.

Since delay and false positive alarms are two important indicators for the hospital practice, we also evaluate the performance of this prototype on the basis of latency and detection accuracy. Within the 2577 complex events that we have detected (when we applied all the 41 rules), around one third are related to patient identification at a series of locations they need to go through for a surgery. Another one third of complex events signify a variety of medical threats to the patient, e.g., high fever and heart attack. The remaining one third of situations concerns sending reminders to surgical personnel beforehand so they can get prepared, access control of medical equipments, and improper disposal of reusable instruments. Since we give a clear definition of rules, the true positive alerts are identified with 100% accuracy. However, in the real case, the definition of event patterns is fuzzy so it is not so easy to provide such positive results. The latency of detection time of these scenarios is all below 1 s, which is quite acceptable. However, we need to cut down the number of unnecessary or inconductive reminders and alerts. Otherwise, healthcare professionals can easily get disturbed.

5.5. A use case of surgical workflow

In general, the performance of the CEP approach to process RFID and sensor data using Drools fusion is acceptable in the hospital environment. In this section, we illustrate how this approach can improve the surgical workflow described in Section 2.

Fig. 7 presents the simplified surgical workflow modeled by Drools-flow, which is a part of the Drools integration framework. Two types of basic events are considered including RFID event and patient physiological event. Hospital database is accessed to retrieve patient medical records and surgery schedule. These data are inserted into the CEP engine as facts and associated with event streams for complex event detection. When various resources are involved in a surgical workflow, events generated from their activities can impact each other and result in different surgical outcomes. In such a time critical environment, temporal relationships of these events need to be captured to detect critical situations.

In this workflow, we define CEP rules to model critical situations in surgical management and embed these rules in workflow activities. We focus on two scenarios: patient identification and patient monitoring. As described, patient identification should be conducted at several stages: admission to L3, admission to L5, anesthesia, and leaving from L6. Accurate and timely identification of patients can avoid adverse situations such as "wrong patient", "wrong OR", and "wrong procedure". When a patient enters an OR, a complex event is generated. The patient identification rule also checks surgery schedule to see if the patient is scheduled in this OR. If not, an alert is triggered. Otherwise, the subsequent activity is activated.

Patient monitoring captures the change of a patient's physiological parameters and sends alerts if any emergency happens. When surgeons are operating on a patient, they are always too focused on the operation to be aware of the patient's condition. Thus, the detection of critical situations may be delayed and the patient's life will be threatened. The physiological events of a patient will evolve with time and needs to be associated with the patient's medical background. Thus, temporal reasoning is critical to capture this feature.

We present a screenshot from our prototype system in Fig. 8 for patient monitoring. The left panel presents the patient in the current OR is "Jack Miller" and his physiological parameters are changing continuously over time. These physiological data are simulated by a computer program following a normal distribution function. For control purpose, we altered some data points to play the role of complex event detection. For example, in the CTCO2 chart, a peak is created and in the SvO2 chart, a smaller peak and valley are created. With these variations, we are able to detect different kinds of situations. Besides, the banner bar in the bottom keeps rolling and shows variations of these parameters in real

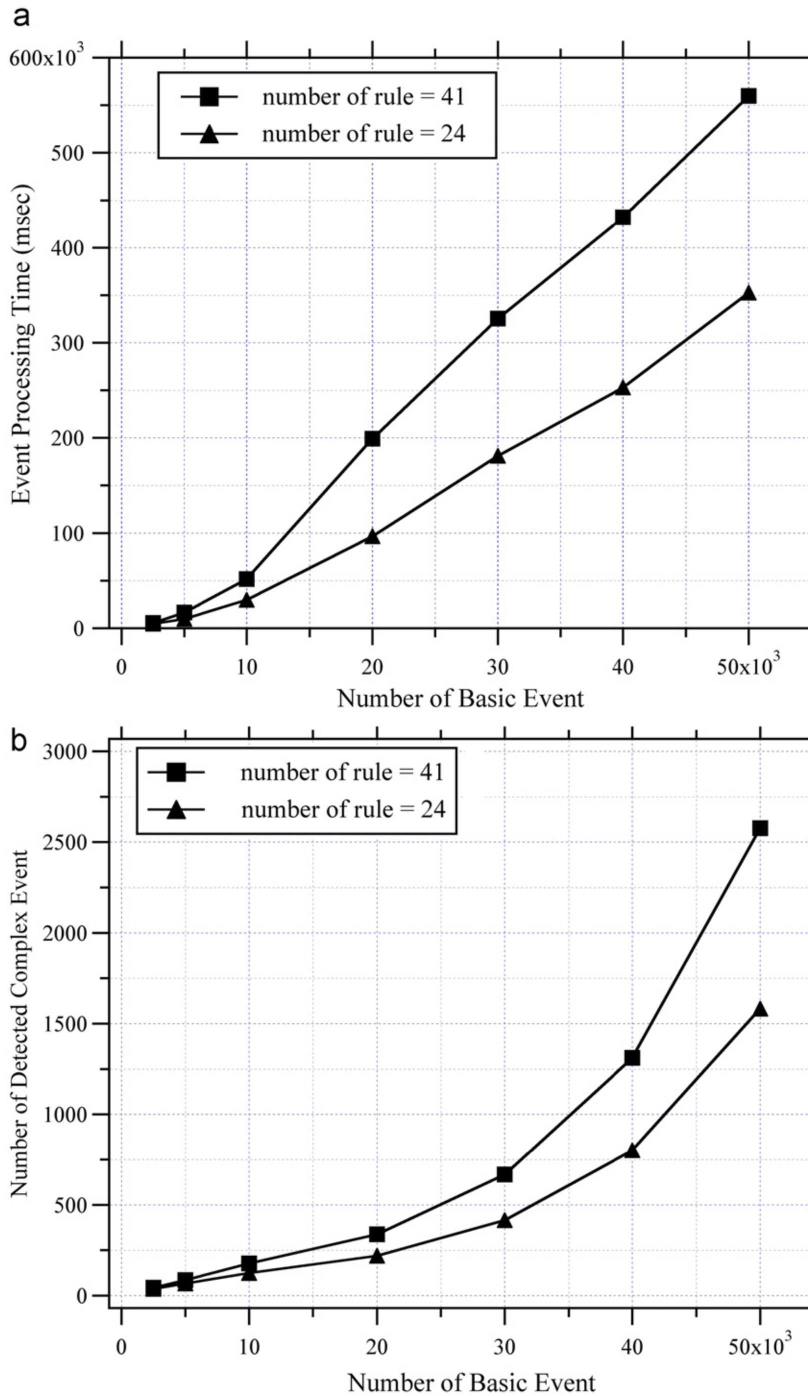

Fig. 6. Performance evaluation: (a) event processing time and (b) detection of complex events.

time, such as increasing and decreasing rates. The right panel shows raw RFID readings that have not been processed. We can see that their semantics are in a very low level and do not make sense to people. The central panel presents detected complex events like reminders and alerts. These events contain more semantic meanings and help healthcare personnel to respond to critical situations quickly.

6. Discussion

According to our study, a number of benefits can be obtained by using RFID-enabled CEP framework for hospitals to improve patient safety and reduce operational costs. The experimental results show that our proposed approach is feasible in practice. Our proposal of integrating CEP logic with RFID technology has several advantages over conventional methods.

First, we identify current challenges in hospitals and model an RFID-enabled smart hospital. With the tracking capability offered by RFID technology, the smart hospital has the promise to track people, equipment, and even the medicines. Thus, these objects have the power to express themselves. By associating their EPC with the hospital database, we can get more detailed information of these objects.

Second, CEP enables semantic interoperability for a variety of sensors, embedded devices, and information systems. CEP

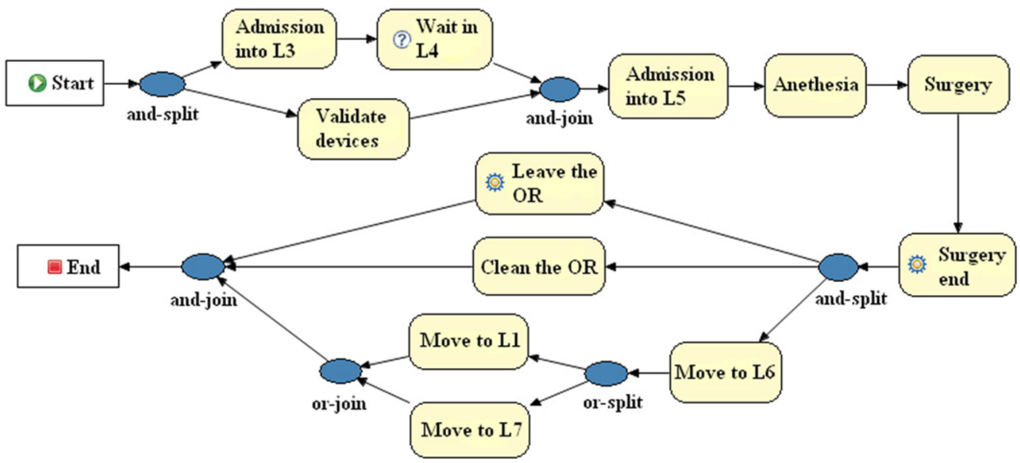

Fig. 7. Simplified surgical workflow modeled by Drools-flow.

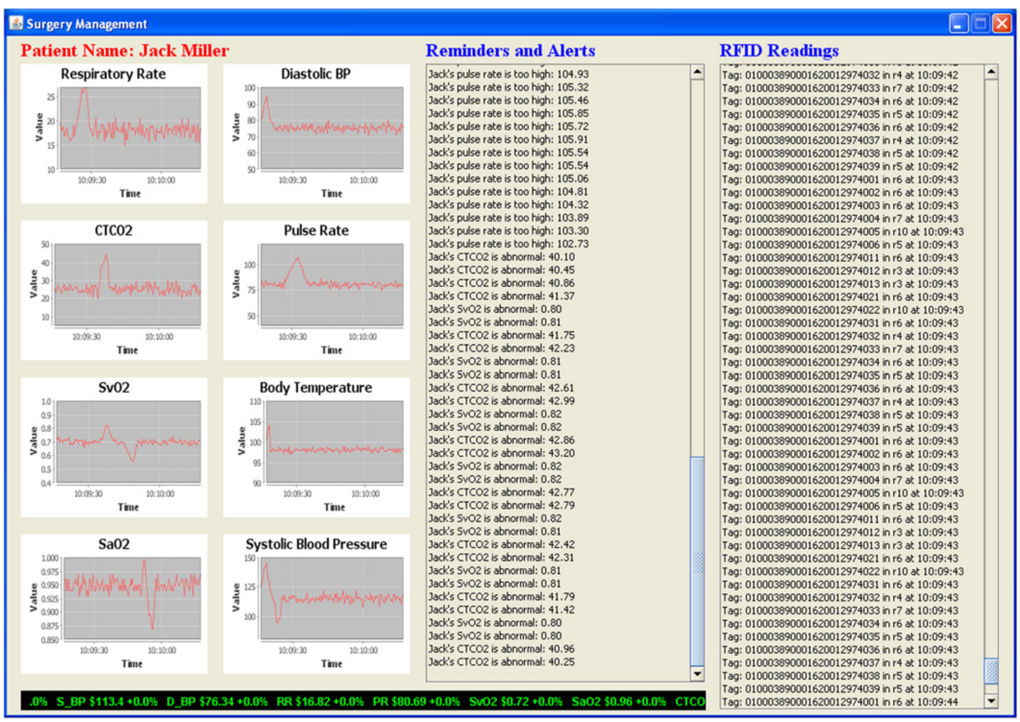

Fig. 8. A screenshot of our prototype.

can deal with data streams from different sources and correlate them to identify critical situations. Although the data from sensors, RFID systems, and messages from other systems are in different formats and incoming rates, CEP can easily integrate these basic events with event patterns. For example, RFID system itself only provides location and time information of tagged objects while humidity sensors only offer room conditions. In the real world scenario, the room humidity needs to be adjusted according to what that room is used for. Monitoring everything manually is time and effort consuming, and error-prone.

Third, with the complex events detected by CEP, service providers are able to sense and respond to unexpected changes immediately. That is, they can identify situations that require immediate attention to increase real-time responsiveness. Given CEP's degree of access and visibility, hospitals can benefit a lot as it has unpredictable and chaotic environments.

Last, our proposed framework can improve the quality of care services and reduce operational costs. Fig. 9 summarized a hierarchy of possible benefits that can be brought to the hospitals. For example, the system can reduce human errors and improve medical treatment quality. Among all the benefits brought by RFID technology, the advantages of CEP are grayed out to show its benefits to hospital practices. Most of these benefits can be captured and realized by our proposed approach.

However, there are some limitations in our approach. Our current rule base uses fixed combination of parameters and values to detect complex events. That is, it is not able to detect uncertain situations. For example, in patient monitoring, a single value cannot determine whether the patient's heart rate is definitely high or low. We need a range of values to model these fuzzy characteristics. Using fuzzy logic to partition the range of values for these physiological parameters can improve the detection accuracy in practice.

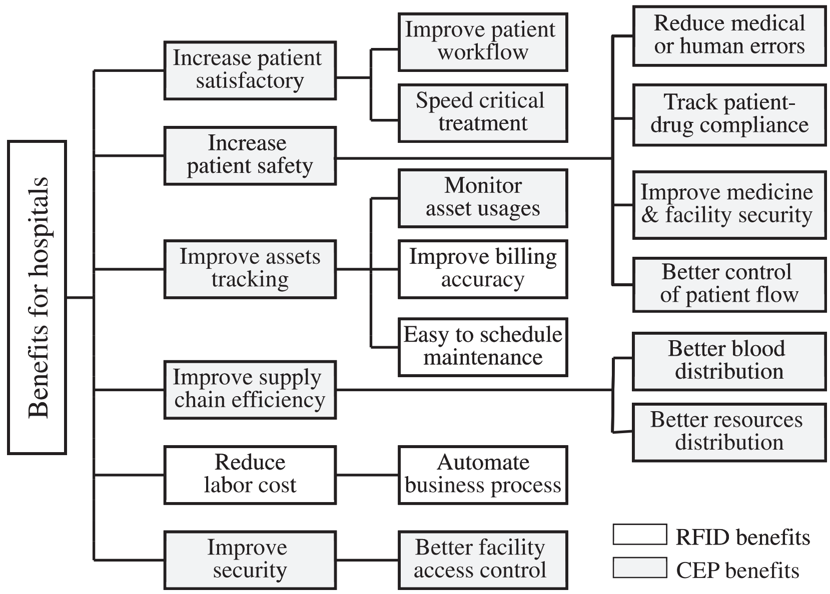

Fig. 9. Benefits of CEP-embedded RFID solution.

7. Related work

The CEP model, which was designed specifically to address issues around processing real-time events in distributed systems, has been studied extensively in the past (Luckham, 2002; Wu et al., 2006). Some studies focused on detecting complex events for rule-based business activity management in ubiquitous computing environment (Chakravarthy et al., 1994; Jeng et al., 2004). However, these studies are generic in nature, and do not address characteristics that are unique to RFID technology and the healthcare domain.

With the rapid development of RFID technology, major IT vendors have been developing RFID middleware to collect RFID data from readers and emit them to applications. Researchers have studied the issue of RFID data management, such as data filtering and aggregation (Jeffery et al., 2006). Recently, more studies have started to explore the use of CEP in RFID data management. The traditional ECA model (McCarthy and Dayal, 1989), based on active database systems, could not be directly applied to RFID systems because RFID data is temporal and in large volume. Wang and Liu (2005) and Wang et al. (2006) used CEP as a means to create complex events from an RFID data stream that can then be used as a foundation for business application logic. However, their focus was on general RFID applications instead of domain specific systems. Son et al. (2007) proposed an efficient method to create business-level events using CEP based on RFID standards. Other works have provided solutions and developed systems to integrate non-RFID data to detect complex events (Bornhövd et al., 2004; Zang et al., 2008).

The above studies provide a solid foundation for building RFID-enabled CEP framework for hospitals. RFID has been proposed as a technology to provide ubiquitous computing support for medical work in hospitals (Bardram, 2003; Want, 2004). Although a number of researchers have been working on developing context-sensitive and situation awareness services in the pervasive computing environment (Jeong and Kim (2005); Moon et al., 2006; Song and Kim, 2006), these services are not specific for healthcare domain. They need to be tailored to fit the medical knowledge and special needs in hospitals. Only a few works have been conducted to develop context-aware systems for hospitals. For example, Bardram (2004) presented applications of context-aware computing in hospitals but their study is still in conceptual stage. Agarwal et al. (2007) proposed a pervasive computing system for the operating room to detect medically significant events and automatically construct an electronic medical record. Their results are very interesting but they use the traditional rule-based approach; thus, the delay of detection time is from a few seconds to 56 s.

8. Conclusions and future work

This paper presents a novel approach to process surgical events and provide sense and response capability for hospitals. Although the idea of using CEP to process RFID events and correlate non-RFID events are not new, we use this RFID-enabled CEP framework to model complex events in a smart hospital and solve the current challenges faced by healthcare. This approach can help to accelerate the adoption of RFID technology in hospitals and provide a feasible way to solve the interoperability problem. The performance evaluation in terms of processing delay and detection accuracy shows that our approach is reasonable and acceptable. With the aim to improve patient safety and reduce operational costs, our study suggests a possible solution to handle problems encountered in surgery.

However, modeling medical knowledge with RFID and CEP is quite complicated. Most of medical knowledge comes from experts and is fuzzy and hard to verify for accuracy. In the future, we will reference more medical literatures and interview medical professionals to increase its completeness and apply fuzzy sets theory to model uncertainty associated with surgical events. This can improve the accuracy of situation detection and reduce unnecessary alerts.

References

Agarwal S, Joshi A, Finin T, Yesha Y, Ganous T. A pervasive computing system for the operating room of the future. Mobile Networks and Applications 2007;12:215–28.
Bali M. Drools JBoss Rules 5.0 developer's guide. Birmingham, UK: Packt Publishing; 2009.
Bardram JE. Hospitals of the future—ubiquitous computing support for medical work in hospitals. In: Proceedings of the second international workshop on

ubiquitous computing for pervasive healthcare applications (UbiHealth'03), 2003.
Bardram JE. Applications of context-aware computing in hospital work: examples and design principles. In: Proceedings of the ACM symposium on applied computing, Nicosia, Cyprus; 2004. p. 1574–9.
Bornhövd C., Lin T, Haller S, Schaper J. Integrating automatic data acquisition with business processes experiences with SAP's auto-ID infrastructure. In: Proceedings of the 30th international conference on very large data bases (VLDB), 2004. p. 1182–8.
Chakravarthy S, Krishnaprasad V, Anwar E, Kim SK. Composite events for active databases: semantics, contexts and detection. In: Proceedings of the 20th VLDB Conference, Santiago, Chile; 1994. p. 606–17.
Forgy CL. Rete: a fast algorithm for the many pattern/many object pattern match problem* 1. Artificial intelligence 1982;19:17–37.
Fuhrer P, Guinard D Building a smart hospital using RFID technologies. In: Proceedings of the first European conference on eHealth (ECEH'06), Fribourg, Switzerland, vol. 5, 2006. p. 10–28.
Jeffery SR, Alonso G, Franklin MJ, Hong W, Widom J Berkeley UC. A pipelined framework for online cleaning of sensor data streams. In: Proceedings of the 22nd international conference on data engineering (ICDE'06) Atlanta, Georgia; 2006. p. 140.
Jeng JJ, Flaxer D, Kapoor S. RuleBAM: a rule-based framework for business activity management. In: Proceedings of the 2004 IEEE international conference on service computing (SCC'04), Shanghai, China; 2004. p. 262–70.
Jeong D, Kim YG, In H. New RFID system architectures supporting situation awareness under ubiquitous environments. Journal of Computer Science 2005;1:114–20.
Krohn R. RFID: it's about more than asset tracking. Journal of Healthcare Information Management 2005;19:20.
Luckham D. The power of events: an introduction to complex event processing in distributed enterprise systems. Springer; 2002.
McCarthy D, Dayal U. The architecture of an active database management system. ACM SIGMOD Record 1989;18:215–24.
Moon M, Kim Y, Yeom K. Contextual events framework in RFID system. In: Proceedings of the third international conference on information technology: new generations (ITNG'06), Las Vegas, Nevada; 2006. p. 586–7.
BearingPoint. RFID in health care. Poised for growth publication; 2007. 32–35.
Sandlin D. SurgiChip—new technology for prevention of wrong site, wrong procedure, wrong person surgery. Journal of PeriAnesthesia Nursing 2005;20:144–6.
Son B, Lee J, Park K, Kim C, Kim H, Kim S. An efficient method to create business level events using complex event processing based on RFID standards. Software technologies for embedded and ubiquitous systems. Berlin/Heidelberg: Springer; 2007. p. 1–10.
Song J, Kim H. The RFID middleware system supporting context-aware access control service. In: Proceedings of the eighth international conference on advanced communication technology (ICACT'06), Phoenix Park, Korea, vol. 1, 2006. p. 863–6.
Su C-J, Chou T-C. Improving patient safety and control in operating room by leveraging RFID technology. In: Proceedings of the international multiconference of engineers and computer scientists (IMECS'08), Hong Kong; 2008. p. 2.
Walzer K., Breddin T., Groch M. Relative temporal constraints in the Rete algorithm for complex event detection. Proceedings of the second international conference on Distributed event-based systems. Rome, Italy: ACM; 2008. p. 147–55.
Wang F, Liu P. Temporal management of RFID data. In: Proceedings of the 31st international conference on very large data bases, Trondheim, Norway; 2005. p. 1128–39.
Wang F, Liu S, Liu P. Complex RFID event processing. The VLDB Journal 2009;18:913–31.
Wang F, Liu S, Liu P, Bai Y. Bridging physical and virtual worlds: complex event processing for RFID data streams. In: International conference on extending database technology (EDBT), Munich, Germany, vol. 3896, 2006. p. 588–607.
Want R. Enabling ubiquitous sensing with RFID. Computer 2004;37:84–6.
Wu E, Diao Y, Rizvi S. High-performance complex event processing over streams. In: Proceedings of the 2006 ACM SIGMOD international conference on management of data, 2006. p. 418.
Zang C, Fan Y, Liu R. Architecture, implementation and application of complex event processing in enterprise information systems based on RFID. Information Systems Frontiers 2008;10:543–53.