

차세대 가스안전관리를 위한 RFID/USN 이벤트 모니터링 시스템의 설계

이찬근 · 박재화[†]

중앙대학교 기계공학부
(2008. 7. 14. 접수 / 2008. 10. 10. 채택)

The Design of an RFID/USN Event Monitoring System for Next Generation Gas Safety Management

Chan Gun Lee · Jae Hwa Park[†]

School of Computer Science and Engineering, Chung Ang University
(Received July 14, 2008 / Accepted October 10, 2008)

Abstract : There are many efforts toward the development of next generation gas safety management systems to improve the assurance, availability, and efficiency of the extant systems. Among them, the application of ubiquitous technology, such as RFID and USN, to the system is regarded as a promising approach for enabling such an innovation. Recently the ubiquitous technology became much affordable and available than before thanks to the reduced production cost and wide proliferation of the ubiquitous devices. In this paper, we analyze the core requirements for the next generation gas safety management systems based on the ubiquitous technology, and present an architectural design of a middleware to facilitate the realization of the system. We formally define an event model and present how to specify various situations related to gas safety management using the proposed event model.

Key Words : RFID, USN, gas safety management, architectural design, event model

1. 서론

본 논문은 차세대 가스안전 관리 시스템이 갖추어야 할 주요 기능들을 제시하고 그의 실현에 필요한 핵심 기술과 시스템의 설계를 다룬다. 특히 라디오 주파수 인식(Radio Frequency IDentification, 이하 RFID) 및 유비쿼터스 센서네트워크(Ubiquitous Sensor Network, 이하 USN)기반의 가스안전 관리 시스템이 가지는 핵심 요구 사항을 분석하고, 실시간 이벤트 모니터링 분야의 연구 결과를 응용한 가스안전 관리 시스템의 설계를 제시한다.

가스 안전사고를 예방하기 위하여 안전교육 강화, 지속적인 홍보, 제도 개선 및 정비 등 다양한 노력을 기울이고 있다. 이에 따라 점차적으로 가스안전 사고 건수가 감소하고 있으나 가스안전 관련 사고 특성상 사고 발생 시 대규모의 인적 및 물적 피해가 동반되기 때문에, 가스안전 사고를 사전에 막을 수 있는 체계적인 관리 시스템의 도입이 시급한 실정이다.

최근 정보기술(Information Technology)분야의 연구 성과들이 다양한 분야로 응용되고 있다. 그 중 유비쿼터스(ubiquitous)관련 기술들을 가스안전 분야에 적용하여 차세대 가스안전 관리시스템을 구축하려는 연구가 활발히 진행되고 있다.

2002년에서 2006년까지 미국 에너지부(Department of Energy, DOE)의 지원 하에 수행된 GASNET 프로젝트⁸⁾는 센서를 이용하여 가스배관을 구성하는 시설기들의 현재 상태를 감시하고 이러한 정보에 대해 중앙 집중적인 관리가 가능함을 보여주었다. 이것은 IT와 에너지, 안전 시스템과의 융합을 통해 체계적인 안전관리 시스템 구축과 천연가스의 분배 기능을 최적화하도록 한 대표적인 연구 개발 사례이다. 우리나라에서도 가스안전관리에 IT기술을 도입하고자 하는 노력이 시도되고 있다. 대표적으로 가스안전공사는 LPG용기에 RFID칩을 부착하여 불법 LPG 용기의 유통을 막고 효율적인 관리를 할 수 있도록 하는 방안을 추진 중이다.

우리는 본 논문에서 유비쿼터스 IT 환경의 대표적인 네트워킹 구성요소인 RFID 및 USN을 이용하여 가스안전과 관련된 이벤트를 다양한 경로를

[†] To whom correspondence should be addressed.
jaehwa@cau.ac.kr

통하여 분산 수집하고 그 결과들을 실시간으로 분석하여 현재 상황에 대한 종합적인 판단을 내릴 수 있도록 도와주는 차세대 가스안전관리 이벤트 모니터링 시스템을 제안한다. 구체적으로 시스템의 이벤트 모델, 이벤트 정의 언어, 시스템 아키텍처를 제시한다.

본 논문은 다음과 같이 구성되어 있다. 2장에서는 RFID 및 USN에 관련된 배경지식과 미들웨어 기술의 동향에 대해 다룬다. 또한 이벤트 검출 및 복합 이벤트와 관련한 최근 연구결과를 소개한다. 3장에서는 차세대 가스안전 관리시스템의 역할을 소개하고 시스템의 요구 분석을 통해 어떠한 기능들이 제공되어야 하는지 설명한다. 그리고 그러한 요구 분석의 결과로 시스템의 아키텍처를 제시한다. 4장에서는 시스템의 이벤트 모델을 제시하고 복합 이벤트를 이용하여 다양한 가스안전 관련 이벤트 기술(specification)방법을 보인다.

2. 관련연구

2.1. RFID/USN

RFID는 전자 태그의 일종으로 임의의 물체에 부착하여 그 물체의 식별을 위해 사용된다. RFID는 그 물체의 식별정보를 담고 있는 매우 작은 IC 칩(chip)과 전파를 이용하여 외부의 리더(reader) 및 라이터(writer)와 데이터 송수신을 할 수 있는 안테나를 내장하고 있다¹³⁾. 유사한 목적으로 현재 널리 사용되고 있는 바코드에 비해 RFID의 경우 IC 칩 내에 보관할 수 있는 정보의 규모가 훨씬 크고, 내구성 면에서도 월등하며, RFID태그와 리더 사이에 차폐물이 있는 경우에도 정보를 읽어 낼 수 있는 장점이 있다.

최근 월마트와 같은 대형 유통업 기업들이 RFID를 이용한 재고, 판매 관리에 많은 관심을 보이고 있다²¹⁾. 특히 비교적 저가이면서 크기가 매우 작은 RFID는 스마트 라벨(smart label)로 불리며 소규모 제품의 다양한 정보를 담을 수 있어 향후 많은 응용분야로 적용될 것으로 보인다. 예를 들어 약국에서 구입할 수 있는 제약품의 경우 그 성분과 복용방법, 부작용, 유효 기간, 유통 경로와 같은 정보를 보관하게 할 수 있다. RFID는 배터리를 내장하고 있는 능동형 태그(active tag)와 배터리를 내장하지 않는 수동형 태그(passive tag) 두 종류가 있다. 일반적으로 능동형 태그는 배터리의 수명에 의존하는 문제를 가지고 있고 비교적 고가이나 인식 가능한

거리가 수십 미터에서 수백 미터로 우수하다. 반면 수동형 태그는 저가이며 소형이라는 장점이 있고 인식 가능한 거리가 수 미터 이내로 제한된다.

USN은 주변의 여러 환경 요소(온도, 습도, 조도 등)를 감지 및 측정하는 센서 노드들이 구성하고 있는 소규모의 네트워크이다²⁾. 이들 센서 노드는 단순히 수동적인 측정을 하는 것이 아니라 연산 능력(computation power)을 가지고 있는 프로세서를 자체적으로 가지고 있고 소규모이기는 하지만 기억장치(memory)를 가지고 있어서 프로그래밍이 가능하다. 또한 RFID와 같이 안테나를 내장하고 있어서 수집된 데이터를 프로세서를 이용하여 가공하여 다른 센서나 네트워크 장치로 전송할 수 있는 통신 기능을 포함한다.

센서 노드에 관련된 대표적 프로젝트에는 미국 UC Berkeley 대학에서 주도한 Smart Dust²³⁾, UCLA 대학의 WINS, 카네기멜론 대학의 스마트센서 DVB 개발 프로젝트 등이 있다. 상용화된 센서노드에는 Crossbow사의 Mica 계열 제품군, Moteiv의 Tmote Sky, Intel사의 iMote 등이 있고 국내에서는 ETRI를 중심으로 KETI, 삼성, Radioplus 등이 제품 개발을 하고 있다.

2.2. RFID/USN을 위한 미들웨어 기술

RFID/USN 미들웨어(middleware)는 응용 프로그램 계층과 RFID/USN 하드웨어 계층 중간에 위치하며 둘 간의 통합이 원활하게 되도록 도와주는 역할을 한다. 최근 RFID/USN을 위한 미들웨어에 대해 산업계 전반에 걸쳐 활발한 연구와 개발 노력이 있었다. 대표적인 RFID 미들웨어 시스템에는 SAP의 Auto-ID Infrastructure³⁾, 오리클의 Sensor Edge¹⁵⁾, Sun의 EPC Network¹⁸⁾, Sybase의 RFID solutions¹⁹⁾, IBM의 Web Sphere RFID Premises Server²⁴⁾, 마이크로 소프트의 RFID 미들웨어¹⁴⁾, UCLA의 Win RFID²⁰⁾ 등이 있다. 초기의 미들웨어 시스템들은 기본 이벤트 단위의 검출 및 전송 기능에 머물렀지만 최근 개발되는 시스템들에서는 본격적인 이벤트 연관(correlation)을 이용한 복합 이벤트를 지원하고 있다.

이벤트에 관한 연구는 능동 데이터베이스 시스템과 이벤트-조건-행위(Event-Condition-Action, 이하 ECA)모델과 깊은 연관이 있다. 대표적인 능동 데이터베이스 관련 연구에는 Ode¹⁰⁾, HiPAC⁶⁾, Areil¹⁷⁾, REACH⁴⁾, Starbust²⁵⁾, Sentinel⁵⁾, SAMOS⁹⁾ 등이 있다. 능동 데이터베이스 프로토타입들에 대한 광범위한 조사가¹⁶⁾ 이루어졌다.

3. 차세대 가스안전 관리시스템의 설계

3.1. 시스템의 역할

차세대 가스안전 관리시스템은 배관, 밸브, 정압기, 운반용 탱크, 저장소 등 가스 시스템을 이루는 핵심적인 장치들의 상태를 실시간으로 감시함과 동시에 다양한 경로를 통하여 발생하는 가스안전 관련 정보를 분산 수집하고 그 결과들을 온라인으로 분석한다. 분석 결과, 중대한 안전상의 문제가 발생될 가능성이 발견되면 시스템은 사전에 정의된 대처 계획에 따라 자동적으로 적절한 조치를 취하거나 관리자 및 상위 시스템에게 즉각 통보하여 신속한 의사결정에 도움을 준다.

특히 각종 센서에서 측정되어 실시간으로 발생하는 대규모의 이벤트들로부터 시스템에 의미가 있는 이벤트로 변환하여 가스안전 시스템과 연동되는 여러 응용에 효과적으로 전달함으로써 가스 사고 방지를 위한 24시간 자동 모니터링, 조기 이상 발견, 그리고 체계적이며 신속한 대응을 가능케 한다.

3.2. 시스템의 요구사항 분석(Requirement Analysis)

3.2.1. 대규모의 이벤트 처리

RFID리더가 태그를 읽을 때 마다 이벤트가 발생되며 USN의 경우 센서에 설정되어 있는 매 폴링(polling) 시간마다 이벤트가 발생된다. 시스템의 설정에 따라 막대한 규모의 이벤트가 발생할 수 있다. 예를 들어 시스템이 관리하고 있는 가스관이 1,000개가 있고 센서의 폴링 시간이 매 1분으로 설정되어 있다면 각 가스관에 센서를 하나씩 설치하더라도 매분 1,000개의 이벤트가 발생하고, 매일 144만개의 이벤트가 축적될 것이다. 따라서 이와 같은 대규모의 이벤트를 체계적이면서도 효과적으로 처리할 수 있는 기능이 필요하다.

3.2.2. 실시간 이벤트 분석

앞 절에서 언급된 바와 같이 RFID와 센서에서 발생하는 대규모의 이벤트를 실시간으로 분석하여 시스템에게 의미 있는 이벤트를 추출하는 기능이 필요하다. 예를 들어 1분 내에 하나의 가스관에 일정 간격으로 설치되어 있는 압력 센서로부터 밸브의 압력이 현저히 낮아지는 이벤트를 다수 개 받았다면 해당 가스관에서 안전에 심각한 영향을 끼치는 요인이 발생한 것으로 파악할 수 있을 것이다. 이것은 단순히 개별적인 센서 이벤트로부터 추출될

수 있는 결론이 아니고 다수의 센서 이벤트들 간의 시간적 관계(timing relation), 의미적 관계(semantic relation), 위치적 관계(location relation) 등을 실시간에 분석해야만 도출될 수 있다. 따라서 실시간 이벤트 분석기능이 필요하다.

3.2.3. 노이즈 및 중복 이벤트 제거

일반적으로 RFID리더는 태그를 읽을 때 상당한 인식 오차율을 가지고 있는 것으로 알려져 있다. Floerkemeier와 Lampe는 최근의 연구⁷⁾에서 RFID리더가 주위의 태그 중 60~70% 만을 올바르게 검출한다는 것을 보여주었다. 또한 미국 버클리 소재 인텔 연구소의 무선 센서 네트워크 연구¹⁾에서는 센서들에 요구된 데이터 중 단지 42% 만의 데이터가 평균적으로 전달되는 것을 보여주었다. 그러므로 이와 같은 중복되어있거나 노이즈가 포함된 이벤트들을 정제시키는 기술이 필요하다.

3.2.4. 하드웨어 비 종속적 서비스 계층

차세대 가스안전 관리시스템은 다양한 제품군의 센서 및 RFID로 이루어져 있을 수 있다. 따라서 시스템의 말단에 있는 센서와 RFID는 매우 이질적이고 다양한 물리적인 특성을 가지고 있을 수 있다. 이와 같은 환경에서 미들웨어가 플랫폼의 변화에 즉각 영향을 받는 종속적 시스템으로 구성된다면 말단에 있는 부품의 교체 시 항상 상위의 시스템이 바뀌어야 하므로 시스템 유지보수 측면에서 매우 불합리하다. 따라서 응용계층에게 서비스를 제공하는 서비스계층은 플랫폼에 독립적이며 또한 하드웨어 변화에 대해 유연해야 한다.

3.2.5. 선언적 질의(declarative query) 기능

위에서 언급된 바와 같이 현 플랫폼을 구성하는 말단의 센서와 RFID는 기술의 발전과 시장의 변화에 의해 새로운 특성을 가진 다른 제품으로 교체될 수 있다. 따라서 응용 계층에서 직접적으로 사용하는 질의 기능의 경우 절차적인(procedural) 언어보다는 선언적인 것이 되어야 한다. 그럼으로써 새로운 기능의 중간 부품이 설치되면 선언적 질의로부터 최적화된 구체적 질의로 변환되어 시스템에 적용된다.

3.2.6. 이벤트에 대한 저장 및 퍼징 기능

대규모의 이벤트를 안정적으로 저장하는 기능이 필요하다. 발생하는 이벤트는 이력 데이터(history

data)로써 실시간 분석 톨과 같이 연동되어 쓰일 수 있으며, 데이터 마이닝과 같은 오프라인(off-line) 분석을 위해서 사용될 수 있다. 또한 더 이상 필요가 없어진 데이터를 영구적으로 시스템에서 삭제하거나 퍼징(purging)하는 기능이 필요하다.

3.2.7. 이벤트들의 타임스탬프 보정

센서와 RFID리더들은 각기 독립된 클럭을 갖고 있으며 이들에게서 발생하는 이벤트는 그 이벤트가 발생된 시간(예를 들어 RFID가 태그를 읽은 시간)을 나타내는 타임스탬프를 수반하게 된다. 이들 센서와 RFID리더들은 분산되어 설치될 뿐 아니라 일반적으로 안정적인 전원을 공급받지 못하고 배터리에 의존하여 동작하기 때문에 모든 장치들 간에 클럭 동기화(synchronization)가 되어 있지 못한 경우가 많다. 따라서 이러한 이벤트를 기반으로 동작하는 시스템의 경우 동기화가 되지 않은 타임스탬프를 가진 이벤트들에 대한 질의 방법을 제공해야 한다.

3.3. 시스템 아키텍처

Fig. 1은 차세대 가스안전 시스템의 아키텍처를 보여준다. 본 아키텍처의 특징은 응용프로그램의 의미적 특성에 따라 적합한 데이터 청소(data cleaning) 알고리즘, 스트림 그룹화 방안(stream grouping scheme), 질의 방식(query scheme)을 지원한다는 것이다.

Jeffery 등은 최근의 연구¹²⁾에서 응용 프로그램마다 적합한 데이터 청소 알고리즘이 달라질 수 있으므로 확일적으로 스트림의 경계(edge)에서 데이터 청소를 하는 방법보다 필요에 따라 응용프로그램의 근방에서 처리할 수 있도록 허용하는 것이 더욱 효과적임을 보여주었다. 우리는 이와 같은 접근 방법을 확장하여 다단계의 공유 가능한 필터링과 이벤트 스트림 그룹화 기능을 포함한다.

이벤트 모델: 시스템의 모든 모듈의 근간이 되는 이벤트 모델이 정의된다. 시스템에서 사용되는 이벤트에는 기본 이벤트와 복합이벤트가 있다. 이벤트를 합성하여 복합이벤트를 만들어내기 위해 이벤트 연산자와 함수가 또한 정의된다.

임의 질의(ad-hoc query) 처리기: 시스템에 대한 질의 중 지속적인 평가(evaluation)가 필요하지 않은 질의를 위한 처리기이다. 보통 시스템의 현재 상태를 알아보거나 한번만 수행하고 싶은 질의들은 임의 질의로 표현되고 임의 질의 처리기에 의해 수행 된다.

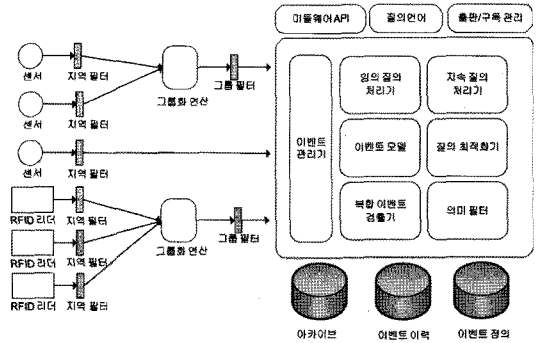


Fig. 1. 차세대 가스안전 시스템 아키텍처.

지속 질의(continuous query) 처리기: 지속적인 평가가 필요한 질의를 처리해 주는 모듈이다. 지속 질의는 이벤트에 기반 한 시스템들에서 대부분을 차지하는 질의이다. 지속 질의에 명시된 시간 윈도우를 슬라이딩 시켜가면서 그 안에 속한 이벤트들을 해당 질의의 후보로 삼아 처리하게 된다.

이벤트 관리기: 외부에서 시스템으로 전송되는 이벤트들을 가공 및 저장하여 다른 모듈들이 처리할 수 있도록 한다. 예를 들어 원시 이벤트를 가공하여 기본 이벤트로 만들고 각각을 아카이브와 이벤트 이력 데이터베이스에 저장시킨다.

질의 최적화기: 임의 질의 처리기 및 지속 질의 처리기의 입력으로 들어온 질의를 가장 최적으로 수행할 수 있는 물리적 질의 계획(physical query plan)을 작성한다. 시스템에 입력되는 이벤트의 도착 패턴(이벤트 입력 속도, 이벤트 발생 분포 등)과 질의 처리기에서 가용한 물리적 연산자, 이벤트 이력 데이터베이스의 통계치 등을 이용한다.

각종 필터: 필터는 지역 필터, 그룹 필터, 의미 필터로 나뉜다. 일반적으로 단말에 위치하는 있는 RFID 리더 및 센서는 상이한 하드웨어적인 특성을 가지고 있다. 따라서 각 단말에 적합한 필터링 알고리즘이 필요하며 이때의 필터링은 다른 장치에서 발생하는 이벤트와 독립적으로 수행되어 지역 필터링이라 불린다. 여러 소스에서 발생하는 이벤트 스트림을 그룹화 할 경우 더욱 효과적인 필터링을 할 수 있다. 두 개 이상의 RFID 리더에서 동일한 태그를 읽었을 경우 둘 중의 하나만이 전달될 필요가 있다. 따라서 그룹 필터링은 같은 스트림으로 입력될 이벤트들에 대한 필터링을 수행한다. 모든 소스에서 발생하는 이벤트가 모아지면, 중앙에서는 응용프로그램에서 요구하는 특성화된 필터링을 수행할 수 있다. 이는 응용프로그램의 의미(semantics)에 따라 달라지므로 미들웨어 내부에 위치한다.

그룹화 연산: 그룹화 연산에는 파괴적(destructive) 그룹화와 비파괴적(non-destructive) 그룹화가 있다. 파괴적 그룹화는 입력되는 이벤트 스트림의 데이터 일부가 삭제되거나 다른 스트림의 데이터와 병합될 수 있는 방법이다. 예를 들어 같은 소스에서 같은 태그에 대한 정보가 1분 안에 도착하면 최초의 것만 전송하고 나머지는 삭제하는 방법이다. 비파괴적 그룹화는 입력되는 이벤트 스트림의 데이터들이 전혀 손실되지 않고 단지 순서가 바뀔 수 있는 방법이다. 타임스탬프에 의한 정렬 방법이 비파괴적 그룹화의 한 예이다.

미들웨어 API: 응용프로그램이 미들웨어의 각 모듈과 직접 연결되지 않고 미들웨어 API를 통해서만 하여 특정 모듈에 종속적인 응용프로그램의 작성을 막고 차후 모듈 중 일부가 동일한 기능을 제공하는 다른 모듈로 대체되어도 응용프로그램을 수정할 필요가 없도록 한다.

질의 언어: 선언적 질의(declarative query)기능을 제공하여 응용 계층에서 요구하는 기능을 절차적인(procedural)방식으로 받아들이는 것을 피하고 선언적인 표현을 사용하도록 함으로써 질의 최적화를 통해 효과적인 연산을 수행할 수 있도록 한다. Jeffery 등이¹¹⁾ 제안한 바와 같이 특히 센서 데이터 필터링을 위해 선언적인 기술 방법이 요구된다.

출판/구독 관리(publish/subscribe management): 시스템에서 발생하는 특정한 이벤트에 관심이 있는 여러 응용프로그램이 있을 수 있다. 이러한 시스템 간 상호작용을 가장 효과적으로 지원하기 위한 통신 형태는 출판/구독이다. 출판/구독 관리를 통하여 응용프로그램들은 이벤트의 발생 시 자동적으로 통보(notification) 받는 형태로 작성된다.

4. 이벤트 모델과 질의 언어

RFID 및 USN에서 발생하는 데이터를 효과적으로 처리하기 위해서는 그 데이터를 좀 더 추상화된 의미적 데이터(semantic data)로 변환해야 한다. 이를 위해서는 노이즈 필터링(filtering), 위치 변환(location transformation), 그리고 데이터 통합(aggregation) 등이 필요하다. 이러한 작업들이 모두 응용계층에서 이루어 질 수 있지만, 데이터의 통합에 대한 부담을 안게 되며, 이렇게 설계된 응용 프로그램은 쉽게 다른 응용으로 적응시키기 힘들다. 본 절에서는 시스템에서 사용되는 이벤트 모델을 정형적으로 정의하고, 이러한 이벤트 모델을 기반으로 질의 언어를 설계한다.

4.1. 이벤트 종류

이벤트는 일반적으로 어떤 시스템의 상태 변화(change of status)를 나타낸다. 이벤트의 필수적인 속성으로 해당 이벤트가 발생한 시간을 타임스탬프(timestamp)로 가지게 된다. 이벤트에는 다음과 같은 세 종류의 이벤트가 있다.

원시 이벤트(raw event): 원시 이벤트는 유비쿼터스 장치(RFID 리더 혹은 센서)에서 발생되었고 아직 아무런 부가적 처리를 가하지 않은 이벤트를 말한다. 일반적인 경우, 원시 이벤트는 노이즈를 포함하는 경우가 많이 있기 때문에 응용프로그램에서 직접적으로 원시 이벤트를 처리하도록 하지 않고 노이즈 필터링을 통해 기본(primitive) 이벤트로 변환한다. 경우에 따라 아카이빙(archiving) 목적으로 대규모 로깅(logging) 장치에 기록한 후 기본 이벤트로 변환하기도 한다.

기본 이벤트(primitive event): 원시 이벤트에 포함된 노이즈를 제거한 이벤트들은 기본 이벤트가 된다. 또한 원시 이벤트 중 중복(duplication)이 검출되면 그 중 일부를 제거하고 적합한 기본 이벤트로 변환할 수 있다. 따라서 기본 이벤트는 응용 프로그램이 직접 인식할 수 있는 수준의 이벤트이다. 기본 이벤트로 변환하기 위한 필터링은 특정 응용 프로그램이 요구하는 의미적(semantic) 필터링을 사용하지 않았기 때문에 지역 필터링에 해당된다.

복합 이벤트(complex event): 기본 이벤트들은 원시 이벤트의 노이즈와 중복을 제거한 것이기는 하지만, 응용프로그램에서 처리하기에 기본 이벤트 발생량이 지나치게 많을 수 있다. 그리고 기본 이벤트는 모든 응용 프로그램에서 공통적으로 인식할 수 있는 수준으로 추상화된 정보를 담고 있다. 따라서 응용에 특화된 의미(application specific semantics)를 가지고 추상화된 이벤트가 필요하게 된다. 복합 이벤트는 기본 이벤트들과 다른 복합 이벤트들을 이벤트 대수(algebra)를 이용하여 정의될 수 있다. 일반적으로 그 양적 규모 또한 기본 이벤트에 비해 훨씬 작고 발생 빈도(occurrence frequency)도 낮아서, 대부분의 응용레벨 프로그램은 복합이벤트를 기반으로 작성된다.

4.2. 타임스탬프(Time stamp) 모델링

시스템의 특성에 따라 이벤트의 타임스탬프는 전통적인 형태인 스칼라로 정의되거나 확률 분포(probability distribution)와 더불어 인터벌(interval)의 형태로 표현될 수 있다. 예를 들어 10시 30분 20초

와 같이 이벤트가 일어난 정확한 발생 시간(exact occurrence time)을 표시하는 것은 스칼라의 형태로 정의된 타임스탬프이다. 하지만 시스템의 구성에 따라 정확한 이벤트의 발생 시간을 알 수 없는 경우 혹은 다른 이벤트와 조합되어 정의되는 복합이벤트의 경우에는 인터벌 형태의 타임스탬프가 사용될 수 있다. 예를 들어 [10시 30분 20초, 10시 30분 50초]의 경우에는 해당 이벤트가 10시 30분 20초와 50초 사이에서 발생되었으나 정확한 발생 시간은 알 수 없다는 것을 의미한다. 인터벌 내에서의 확률분포는 시스템에서 정의된다.

4.3. 이벤트 대수(Event Algebra)

복합 이벤트는 기본 이벤트들과 다른 복합 이벤트들을 이벤트 연산자를 이용하여 정의하게 된다. 이벤트 연산자를 정의하기 위해 이벤트 함수를 소개한 후, 최근 UCLA와 Siemens의 연구²²⁾에서 개발된 이벤트 연산자를 확장하여 본 시스템에서 사용되는 이벤트 연산자를 정의한다.

4.3.1. 이벤트 함수

@ 함수: 주어진 이벤트의 타임스탬프를 돌려준다. 예를 들어 @e는 이벤트 e가 발생한 시간을 나타낸다. 이때 함수의 반환 값은 시스템의 타임스탬프 정의에 따라 스칼라(scalar)이거나 혹은 인터벌(interval)일 수 있다.

begin 함수: 시스템의 타임스탬프가 스칼라이면 begin(e)는 @e를 돌려주고, 인터벌이면 @e의 시작 시간을 돌려준다.

end 함수: 시스템의 타임스탬프가 스칼라이면 end(e)는 @e를 돌려주고, 인터벌이면 @e의 끝 시간을 돌려준다.

A 함수: 주어진 이벤트의 해당 속성의 값을 돌려준다. 예를 들어 A(e, "tagid")는 RFID 이벤트 e가 가지고 있는 tagid 값을 돌려준다.

4.3.2. 비시간적 이벤트 함수

AND 연산자: AND(e1, e2)는 이벤트 e1과 e2가 모두 발생된 경우에 발생된다. e1과 e2의 발생시간의 순서는 상관이 없다.

~연산자: ~e는 이벤트 e가 발생되지 않았을 경우 발생된다. 이벤트 처리기의 성능상의 문제 때문에 대개의 경우 ~연산자는 다른 이벤트 연산자와 함께 사용된다.

OR 연산자: OR(e1, e2)는 이벤트 e1 혹은 e2 둘

중의 하나라도 발생된 경우에 발생한다.

4.3.3. 시간적 이벤트 함수

SEQ 연산자: SEQ(e1, e2)는 이벤트 e1과 e2가 모두 발생하였고 @e1 ≤ @e2인 경우에 발생된다.

TSEQ 연산자: TSEQ(e1, e2, t1, t2)는 SEQ(e1, e2)이고 t1 ≤ |@e1 - @e2| ≤ t2인 경우에 발생된다.

NSEQ 연산자: NSEQ(e, n)은 이벤트 e가 n번 이상 검출되었을 때 발생된다.

TNSEQ 연산자: TNSEQ(e, n, t1, t2)는 시간 [t1, t2] 사이에 이벤트 e가 n번 이상 검출된 경우에 발생된다.

WITHIN 연산자: WITHIN(e, t)는 end(@e) - begin(@e) ≤ t인 경우에 발생한다.

시스템 상에서 이벤트의 타임스탬프가 인터벌로 정의되는 경우, 위에 정의된 시간적 이벤트 연산자에는 추가적인 매개변수로 확신 임계치(confidence threshold)를 추가하여 기술할 수 있다. 확신 임계치가 설정된 복합 이벤트는 그 이벤트가 발생할 수 있는 확률이 설정된 확신 임계치 보다 높은 경우 발생된다고 정의된다. 예를 들어 SEQ 연산자는 SEQ(e1, e2, c)의 형태로 e1 혹은 e2의 타임스탬프가 인터벌인 경우 세 번째 매개변수로 확신 임계치를 추가하여 기술할 수 있다. 해당 이벤트는 prob(@e1 ≤ @e2) ≥ c인 경우에 발생한다. 유사하게 TSEQ(e1, e2, t1, t2, c)은 prob(t1 ≤ |@e1 - @e2| ≤ t2) ≥ c인 경우에 발생한다.

4.3.4. 복합 이벤트의 정의

본 시스템에서 복합 이벤트는 다음과 같이 정의된다.

- CEVENT 복합이벤트 이름
- AS 복합이벤트 정의
- Attribute 속성 정의 및 초기값
- [CONDITION 절]
- [ACTION 절]

CEVENT절에는 정의하고 있는 복합 이벤트의 이름을 제시한다. 그리고 AS 절에는 앞서 기술된 기본 이벤트 혹은 다른 복합 이벤트들을 이벤트 연산자로 연결하여 복합 이벤트를 정의한다.

CONDITION과 ACTION절은 생략 가능하다. CONDITION절은 이 복합 이벤트가 발생하기 위한 부가

적인 조건을 기술하며 ACTION절은 이 복합 이벤트가 발생했을 때 반응으로 수행하고 싶은 행위(action)가 있을 경우에 기술한다. ACTION절은 특히 가스안전 관리 시스템과 같이 안전과 밀접하고 위험 상황 발생 시 즉각적으로 반응해야 하는 시스템의 경우 유용하게 사용될 수 있다.

앞서 제시된 이벤트 모델을 사용하여 가스안전에 관계된 복합이벤트들을 다음과 같이 정의할 수 있다.

```
CEVENT LeakSign
AS SEQ(pressure_down, ordr, 2 min)
CONDITION Nearby(A(pressure_down, "location"),
A(ordr, "location"))
```

```
CEVENT SignificantLeakSign
AS TNSEQ(LeakSign, 5, 0, 20 min)
Action DispatchERT
```

LeakSign 이벤트는 거리상으로 가까운 압력센서와 냄새센서로부터 2분 이내에 이상 징후가 발생되면 발생된다. 또한 이렇게 정의된 LeakSign 이벤트가 20분 이내에 5번 이상 발생하면 중대한 이상이 발견된 것으로 SignificantLeakSign을 발생시킨다. 부가적으로 SignificantLeakSign 발생 시 비상대응팀(Emergency Response Team)을 급파하는 것을 Action절에 명시할 수 있다.

또 다른 예로써, 가스탱크가 감독자의 인가 없이 외부로 반출되었을 경우 경비원을 출동시키는 규칙을 다음과 같이 정의할 수 있다.

```
CEVENT GASTANK_FILLED_INSTOCK
AS SEQ(refill_gastank, in_gastank)
Condition A(refill_gastank, "tagid") = A(in_gastank,
"tagid")
Attribute tagid as string := A(refill_gastank, "tagid")
```

```
CEVENT GASTANK_CARRYOUT_WITHOUT
APPROVAL
AS SEQ(~supervisor_approve, out_gastank)
Attribute tagid as string := A(out_gastank, "tagid")
```

```
CEVENT DispatchSecurityTrigger
AS TSEQ(GASTANK_FILLED_INSTOCK, GAS-
TANK_CARRYOUT_WITHOUTAPPROVAL, 0, 2 hour)
```

```
Condition A(GASTANK_FILLED_INSTOCK, "tagid")
= A(GASTANK_CARRYOUT_WITHOUTAPPROVAL,
"tagid")
```

Action DispatchSecurity

5. 결론

본 논문에서 우리는 가스 안전관리를 혁신적으로 개선시킬 수 있는 유비쿼터스 기술 기반의 차세대 관리시스템을 제안하고 그러한 시스템이 필요로 하는 핵심적인 요구사항을 분석하였다. 특히 RFID 및 USN에서 발생하는 대규모의 실시간 데이터를 처리하기 위해 RFID/USN 이벤트 모니터링 미들웨어의 아키텍처를 설계하고 주요 모듈이 제공해야 할 기능을 정리하였다.

유비쿼터스 장치에서 발생하는 데이터에 내재된 노이즈 제거, 데이터 중복 제거, 응용에 적합한 의미적 데이터 변환을 위해 다단계의 필터링, 데이터 병합 연산, 이벤트를 통한 의미 변환 기능 등을 제공하였다. 시스템의 상태 변화를 나타내는 이벤트를 원시 이벤트, 기본 이벤트, 복합 이벤트의 3가지로 분류하였다. 다수의 이벤트 연산자를 제공하여 기본 이벤트들과 다른 복합 이벤트들의 조합으로 새로운 복합 이벤트를 정의할 수 있도록 하였다. 기존에 연구된 이벤트 연산자를 확장하여 이벤트가 가지는 시간적 불명확성(uncertainty)을 표현 할 수 있도록 하고 특정 이벤트가 최소 발생 횟수를 정의할 수 있는 이벤트 연산자를 두었다.

향후 시스템의 구현을 통해 실제적인 작업 환경 하에서 제안된 시스템이 얼마나 효과적으로 사용되는지 실험하는 후속 연구가 필요하다. 인적 자원과 긴밀한 협력이 요구되는 안전 관리 시스템의 특성상 시스템의 실제 운용을 통해 분석과 설계 시에 누락된 요구사항과 개선점을 발견할 수 있으며 제안된 이벤트 모델이 충분히 현실적인 시나리오를 표현할 수 있는지에 대한 검증이 필요하다. 또한 도시 가스 파이프의 매설 정보를 담고 있는 지리 정보 시스템 등과의 연동으로 보다 효과적인 시스템으로 발전시킬 수 있을 것이다.

감사의 글 : 본 연구는 지식경제부의 에너지기술혁신 프로그램으로 지원되었으며 이 논문은 “차세대에너지안전연구단”의 연구 결과입니다(세부와 제번호 : 2007- M-CC23-P-06-1-000). 또한 본 연구는 서울시 산학연 협력사업(CR070019)의 지원을 받았

습니다.

참고문헌

- 1) Sonoma Redwood Sensor Network Deployment, <http://www.cs.berkeley.edu/~get/sonoma/>.
- 2) I. F. Akyildiz et al., "A Survey on Sensor Networks", *IEEE Communications Magazine*, 40(8), pp. 102 ~ 114, 2002.
- 3) C. Bornhevd et al., "Integrating automatic data acquisition with business processes experiences with SAP's auto-ID infrastructure", In Proc. of the International Conference on Very Large Data Bases(VLDB), pp. 1182 ~ 1188, 2004.
- 4) A. P. Buchmann et al., "The REACH Active OODBMS", In Proc. of the ACM SIGMOD International Conference on Management of Data, pp. 476, 1995.
- 5) S. Chakravarthy et al., "Composite Events for Active Databases: Semantics, Contexts and Detection", In Proc. of the International Conference on Very Large Data Bases(VLDB), pp. 606~617, 1994.
- 6) U. Dayal et al., "The HiPAC Project: Combining Active Databases and Timing Constraints", *ACM SIGMOD RECORD*, Vol. 17, No. 1, pp. 51 ~ 70, 1988.
- 7) C. Floerkemeier and M. Lampe, "Issues with RFID usage in ubiquitous computing applications", In Proc. of the International Conference on Pervasive Computing, pp. 188 ~ 193, 2004.
- 8) GASNET: Gasline Network Sensor System, http://www.netl.doe.gov/technologies/oil-gas/NaturalGas/Projects_n/TDS/TD/T&D_A_41320GASNET.html.
- 9) S. Gatzu and K. R. Dittrich, "SAMOS: An active, Object-Oriented Database System", *IEEE Database Engineering Bulletin*, Vol. 15, No. 1, pp. 23 ~ 26, 1992.
- 10) N. H. Gehani and H. V. Jagadish, "Ode as an Active Database: Constraints and Triggers", In Proc. of the International Conference on Very Large Data Bases (VLDB), pp. 327 ~ 336, 1991.
- 11) S. R. Jeffery et al., "Declarative Support for Sensor Data Cleaning", In Proc. of the International Conference on Pervasive Computing, pp. 83 ~ 100, 2006.
- 12) S. R. Jeffery et al., "Adaptive cleaning for RFID data streams", In Proc. of the International Conference on Very Large Data Bases(VLDB), pp. 163 ~ 174, 2006.
- 13) Klaus Finkenzeller, *RFID-Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification 2nd edition*, Wiley & Sons LTD, 2003.
- 14) Microsoft RFID Technology Overview, <http://msdn.microsoft.com/en-us/library/aa479362.aspx>.
- 15) Oracle Sensor Edge Server, http://www.oracle.com/technology/products/iaswe/edge_server.
- 16) N. W. Paton and O. Diaz, "Active Database Systems", *ACM Computing Surveys*, Vol. 31. No. 1, pp. 63 ~ 103, 1999.
- 17) U. Schreier et al., "Alert: An architecture for transforming a passive DBMS into an active DBMS", In Proc. of the International Conference on Very Large Data Bases(VLDB), pp. 469 ~ 478, 1991.
- 18) The Sun™ EPC Network Architecture, http://www.sun.com/solutions/documents/white-papers/re_EPC_NetArch_wp_dd.pdf.
- 19) SybaseRFID Solutions, <http://www.sybase.com/rfid>.
- 20) UCLA WinRFID, <http://www.wireless.ucla.edu/rfid/winrfid/>.
- 21) Walmart Supplier Information: Radio Frequency Identification Usage, <http://www.walmartstores.com>, 2005.
- 22) Walmart Supplier Information: Radio Frequency Identification Usage, <http://www.walmartstores.com>, 2005.
- 23) F. Wang et al., "Bridging Physical and Virtual Worlds: Complex Event Processing for RFID Data Streams", *LNCS 3896, EDBT*, pp. 588 ~ 607, 2006.
- 24) WebSphere RFID Premises Server, http://www-306.ibm.com/software/pervasive/ws_rfid_premises_server.
- 25) J. Widom and S.Ceri, "Active Database Rules", *Active Data base Systems: Triggers and Rules For Advanced Database Processing*, Morgan Kaufmann Publishers, 1996.