RFID/EPCIS를 이용한 실시간 서비스 정보 시스템 설계

신명숙*, 박영옥*, 방극인**, 안성수***, 민혜란*, 박도준*, 이준*
*조선대학교 컴퓨터공학과
**나주대학 컴퓨터정보과
***동신대학교 컴퓨터공학과
e-mail:msshin@chosun.ac.kr

Design of Service Information System with Real Time using RFID/EPCIS

Myeong-Sook Shin*, Young-Ok Park*, Keug-In Bang**,
Seong-Soo Ahn***, Hyeran-Ran Min*, Do-Joon Park*, Joon Lee*
*Dept. of Computer Engineering, Chosun University
**Dept. of Computer Information, Naju College
***Dept. of Computer Engineering, Dongshin University

요 약

RFID 기술이 발전하면서 다양한 분야에서 이를 활용하기 위한 연구들이 진행되고 있다. EPC 네트워크 시스템은 EPC로 구동되는 모든 이벤트를 캡쳐하는 EPCIS를 가지고 있으며, 제품 정보를 포함한다. EPC 태그가 다른 곳으로 이동될 때마다, 리더는 태그를 읽으며 태그 데이터가 EPCIS에 저장된다. EPCIS는 데이터 저장 시스템으로써 클라이언트가 서비스를 요구할 때 원하는 데이터를 제공한다. 모든 서비스가 웹 서비스를 통하여 EPCIS로 접근되며 EPCIS는 클라이언트로 하여금 제품의 실시간 상태를 얻을 수 있게 한다. 본 논문에서는 네트워크에서 실시간으로 제품 상태를 파악하고, 재고 관리를 위해 EPCIS를 이용한 서비스 정보 시스템을 제안하였다.

1. 서 론

컴퓨터 기술과 다양한 응용분야에서 사용자들이 인지하기 힘들 정도로 빠른 속도로 발전하고 있다. 최근 이러한 사용자 중심의 최적 컴퓨팅 조건을 만족시키기 위한 새로운 개념의 환경을 제시하고 있는데, 이를 유비쿼터스 컴퓨팅 환경(Ubiquitous Computing Environment)[1]이라고 할 수 있다. 또한 RFID(Radio Frequency IDentification)[2] 기술이 발전하면서 다양한 분야에서 이를 활용하기 위한 연구들이 진행되고 있다. 특히 물류 분야에서 모든 물건에 RFID 태그가 부착되면 이를 통해 물건 각각을 구분할 수 있다.

본 논문에서는 네트워크에서 효율적으로 실시간 서비스를 위해 EPCIS(EPC Information Service)[3]를 이용한 서비스 정보 시스템을 제안한다. 이를 위해 EPC(Electronic Product Code)[4] 네트워크에 있는 각각의 EPCIS는 EPC 이벤트 데이터를 포함하며 이벤트 데이터는 주로 리더 ID, EPC, 확보된 각각 이벤트 타임을 가지고 있으며 리더 ID는 제품의 위치를 인식한다. 또한 실시간으로 제품을 검색하고 로컬 데이터베이스를 조사하여 원하는 제품의 현재 위치를 실시간으로 얻을 수 있다.

본 논문의 구성은 다음과 같다. 2장은 관련 연구로서 RFID 시스템에 대한 전반적인 소개를 하고 3장에서는 실시간 서비스 정보 시스템을 설계한다. 마지막으로 4장에서는 결론 및 향후 연구 방향을 제시한다.

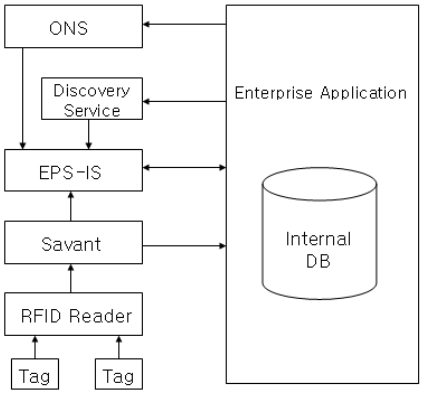
2. 관련 연구

본 논문에서 제안한 실시간 서비스 정보 시스템을 설계하기 위해 RFID, EPCIS에 대하여 알아본다.

2.1 RFID

RFID 시스템은 ID 시스템 중 통신매체로 전자파를 이용한 비 접촉 방식 시스템이다. 배터리가 내장된 것을 액티브 RFID, 배터리가 없이 리더 쪽에서 에너지를 받아 구동하는 것을 패시브 RFID라 한다. RFID는 중장파를 사용하는 것과 마이크로파를 사용하는 것이 있으며, 중장파용 RFID의 주파수범위는 100kHz에서 1.5MHz 이고 데이터 율은 마이크로파 방식에 비해 상대적으로 낮다. 마이크로파 RFID는 900MHz에서 2.4GHz까지 사용되며 데이터 율은 중장파 방식에 비해 높다. 가격은 상대적으로 저주파용에 비해 비싸게 되지만 동작범위는 저주파를 사용하는 RFID에 비해 넓다는 장점이 있다. 중 장파 방식 RFID는 신호 및 에너지 전달 방식으로 코일의 유도결합을 이용하며 마이크로파 방식은 전자파로 신호와 에너지를 전달하고 안테나로 송수신하게 된다. 태그에 전지를 내장하는 것은 통신범위가 넓은 반면 태그의 수명에 제한을 받는다. 반면 태그에 전지를 내장하지 않는 시스템은 통신범위가 좁은 반면 태그의 수명이 반영구적이 된다.

이러한 기본적인 시스템 구성을 바탕으로 글로벌한 RFID 시스템을 구축하기 위해서 ONS, EPCIS Discovery Service, EPCIS, Savant와 같은 추가적인 시스템이 필요하며 네트워크 시스템은 아래 (그림 1)과 같이 구성된다.



(그림 1) EPC 네트워크 시스템 구성

EPC 네트워크 시스템은 주로 ONS, EPCIS, Savant, RFID 리더(RFID Reader), 태그(Tag) 등 5개의 구성요소로 구성된다. EPC 네트워크의 흐름은 RFID 태그에 저장된 EPC 정보를 RFID 리더로 읽어서 Savant가 EPC에 해당하는 관련 정보를 제공하는 PML(Physical Markup Language) 서버의 위치를 알려주는 ONS 서버에 요청하여 PML 서버의 IP주소를 제공 받아 EPCIS에 저장한다. 그리고 정보를 검색하기 위해 서비스 정보 시스템에 접속한다. 검색 서비스는 특정 EPC에 대한 EPCIS 포인터 리스트를 가지고 있으며 EPC 정보를 가지고 있다.

2.2 EPCIS

RFID 시스템은 트랜스폰더를 식별하기 위한 유일한 값으로 EPC를 사용한다. EPC 코드는 Auto-ID 시스템에서 기본이 되는 코드 체계로서 거의 무한한 자원이다.

EPC는 일종의 인터넷 IP 주소와 비슷한 64 비트 또는 128 비트 길이의 ID 숫자로 되어있는 상품 코드를 인식하기 위한 번호 체계이다. EPC는 4개의 필드로 구성되어 있다. 첫 번째 필드는 헤더이며 코드의 버전 넘버이다. 첫 번째 필드는 이름 체계를 구별해 주고 미래에 필요하게 될지 모르는 확장을 위해 고안 되었다. EPC 관리자라고 불리는 두 번째 필드는 코드에 번호를 매기기 원하는 회사나 기관을 식별해 준다. 상품 그룹은 객체 클래스라고 불리는 세 번째 필드에 의해 인식된다. 이미 존재하는 바코드 시스템의 확장인 마지막 필드는 각 상품을 구별해 주는 일련번호로 이루어진 필드이다. EPC의 첫 번째 버전에는, 코드가 각각 24 비트, 36 비트로 총 96 비트로 이루어져 있다.

<표> EPC 구조

필드명	Header	EPC Manager	Object class	Serial number
Bit 위치	0~7	8~35	36~59	60~95

리더가 전송 가능한 범위의 트랜스폰더에게 에너지와 정보를 주게 되면 트랜스폰더는 자신의 EPC 코드를 리더에게 전송하므로 자신의 정보를 전송하게 된다.

리더는 EPC 코드와 부가 정보를 읽어 들여 레거시 시스템의 자료로 저장된다. EPC 코드의 값이 유일한 값이기 때문에 읽혀졌던 데이터를 서비스하면 태그가 부착된 제품의 이동 경로를 파악할 수 있다.

EPC 태그가 다른 위치로 이동할 때마다 리더가 태그를 읽으며 태그 데이터를 저장하기 위해서 사용자들에게 EPC에 대한 이벤트를 저장하고 있다가 클라이언트의 요청이 있으면 원하는 데이터를 검색하여 제공할 수 있는 EPCIS가 필요하다. 또한 EPCIS로 제공된 데이터는 웹 서비스를 통하여 사용 가능하다.

3. 실시간 서비스 정보 시스템 설계

3.1 실시간 서비스 정보 시스템 시나리오

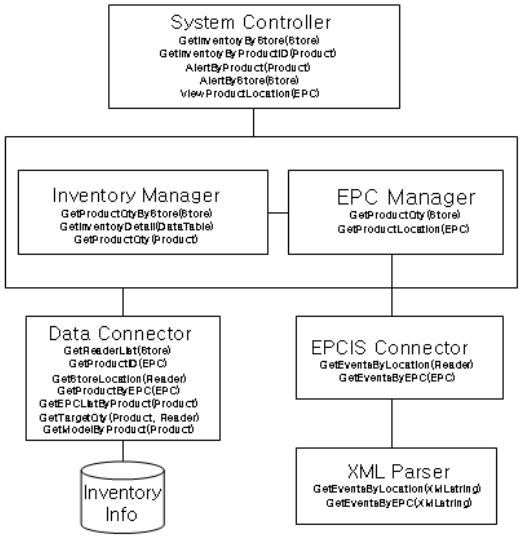
실시간 서비스 정보 시스템은 리더에 의해 읽혀진 EPC 코드를 저장하여 실시간으로 제품의 상태를 파악과 재고 관리를 제공 한다.

본 논문에서는 실시간 서비스 정보 시스템을 이용하여 제품 ID로 재고 상태를 검색하는 시나리오를 가지고 EPCIS를 이용한 실시간 서비스 정보 시스템을 설계한다. 또한 실시간 서비스 정보 시스템은 아래의 상황을 가정한다.

시스템 콘트롤러(System Controller)는 사용자가 선택한 제품을 모으기 위해 재고 관리자(Inventory Manager)에게 메시지를 보내며, 재고 관리자는 사용자가 선택한 저장소에 제품 양을 모으기 위해 EPC 관리자(EPC Manager)에게 메시지를 보낸다. 그리고 재고 관리자는 선택한 저장소에서 일치한 리더 리스트를 모으기 위해 데이터 컨넥터(Data Connector)로 메시지를 보내며, EPC 관리자는 선택된 저장소의 현재 EPC 리스트를 모으기 위해 EPCIS 컨넥터(EPCIS Connector)를 요청한다. 또한 재고 관리자는 제품과 저장소에서 타겟 양을 수집하기 위해 데이터 컨넥터를 요청하며, 저장소 위치로 제품 양을 모은다. 마지막으로 시스템 콘트롤러는 스크린에 보고서를 디스플레이 한다.

3.2 실시간 서비스 정보 시스템 설계

본 논문에서 (그림 2)는 EPCIS를 이용한 실시간 서비스 정보 시스템 설계도이며, 실시간 서비스 정보 시스템을 위한 함수를 정의한다.



(그림 2) 실시간 서비스 정보 시스템 설계도

시스템 콘트롤러는 GUI와 다른 구성요소 사이의 미들 레이어를 생성하는 모든 기능성을 캡슐화하며 특별한 사용자 인터페이스로부터 독립적인 시스템을 만든다. 호출되는 GetInventoryByStore(Store) 함수는 현재 양과 타겟 양을 리턴하며 GetInventoryByProductID(Product)는 제품 ID, 제품 이름을 포함하는 보고서로써 각각의 저장소에서 현재 양과 타겟 양을 리턴한다. AlertByProduct(Product)는 요청한 제품에 대한 타겟 양의 위치와 리스트로 사용자에게 경고하며, AlertByStore(Store)는 현재의 양이 타겟 양으로 요청된 저장소에서 제품 리스트를 사용자에게 경고 한다. 또한 ViewProductLocation(EPC)는 제품이 위치한 곳에서 디스플레이한다.

재고 관리자는 재고 비즈니스 논리를 캡슐화함으로써 타겟 양을 모으며 타겟 항목에 대하여 경고한다. GetProductQtyByStore(Store)는 현재 제품 양, 제품 ID, 위치를 리턴하며 GetInventoryDetail(DataTable)는 데이터를 확보하고 현재 양과 타겟 양 사이의 차이를 계산한다. GetProductQty(Product)는 요청된 제품의 현재 제품 양을 리턴한다.

EPC 관리자는 EPCIS로부터 리턴된 데이터를 조작하며, 현재 재고 양과 저장소에서의 총 제품을 모은다. GetProductQty(Store) 함수는 저장된 현재 제품 양을 모으며 GetProductLocation(EPC)는 요청된 제품의 현재 위치를 얻는다.

또한 EPCIS 컨넥터는 EPCIS에 대하여 웹 서비스 호출을 요청하는 모든 함수를 캡슐화하며 XML 형식에서 모든 입출력은 스트링으로한다. GetEventsByLocation(Reader)는 리더의 위치를 확보하며 확보된 모든 이벤트를 리턴한다. 그리고 GetEventsByEPC(EPC)는 EPC를 확보하며 확보된 최근 이벤트를 리턴한다.

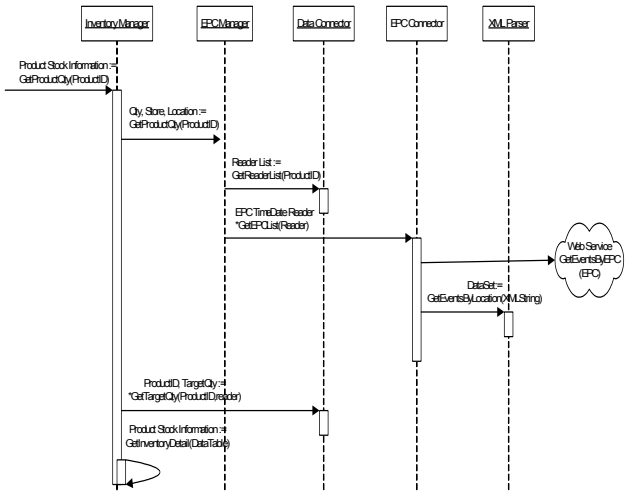
데이터 컨넥터는 데이터베이스 래퍼이며 데이타베이스와 직접적으로 상호 작용하는 모든 함수를 캡슐화한다. 이 함수는 데이타베이스에서 정보를 검색하는 저장된 프로시저를 호출한다. 그리고 리턴된 결과는 데이터 셋 형태이다. 함수로 GetReaderList(Store)는 저장소에 대해 표시된 리더 리스트를 리턴하며 GetProductID(EPC)는 제품 ID를 검색한다. 그리고 GetStoreLocation(Reader)는 리더 위치 명세를 리턴하고 GetProductByEPC(EPC)는 EPC의 제품 ID를 리턴한다. GetEPCListByProduct(Product)는 제품 ID로 EPC 리스트를 리턴하며 GetTargetQty(Product, Reader)는 리더가 제품의 타겟 양을 리턴한다. GetModelByProduct(Product)는 관련된 제품의 모델

리스트를 리턴한다. 또한 EPCIS 컨넥터가 EPC 정보 시스템에 접속하여 함수를 호출할 때, EPCIS에서 제공한 웹 서비스를 호출하여 EPC, 리더 ID, 확보한 이벤트 시간을 리턴한다. 그 다음에 확보한 리더 ID와 일치한 저장 위치를 검색하여 출력해주는데 다음에 나오는 (그림 3)은 EPCIS에서 출력 스트링을 문법적으로 분석할 샘플 소스를 보여준다.

```
"<observations>
<logEvent>
<location>
urn:epc:id:gln:900100.7296</location>
<observation>
<DateTime>2005-7-17T09:30:47</DateTime>
<Tag>
<ID>urn:epc:id:sgtin:900100.456.876</ID>
</Tag>
</observation>
</logEvent>
<logEvent>
<location>urn:epc:id:gln:900100.7297</location>
<observation>
<DateTime>2005-7-15T09:30:47</DateTime>
<Tag>
<ID>urn:epc:id:sgtin:900100.456.876</ID>
</Tag>
</observation>
</logEvent>
<logEvent>
<location>urn:epc:id:gln:900100.7298</location>
<observation>
<DateTime>2005-7-16T09:30:47</DateTime>
<Tag>
<ID>urn:epc:id:sgtin:900100.456.876</ID>
</Tag>
</observation>
</logEvent>
</observations>"
```

(그림 3) GetEventsByEPC(XMLstring) 소스

(그림 4)는 제품 ID로 재고 상태를 실시간으로 검색하는 위의 시나리오에 대한 순차도로 사용자는 저장소에 대하여 특정 제품 양의 검색을 요청한다. EPC 관리자는 요청하는 제품에 대하여 일치하는 EPC 리스트를 첫 번째로 파악한다. EPCIS 컨넥터는 EPCIS를 접속하고 EPC의 리스트에서 EventsByEPC를 호출한다. 웹 서비스는 EPC, 리더 ID, 그리고 확보된 이벤트 시간을 리턴한다. 데이터 컨넥터는 리더 ID를 확보하고 일치한 저장소 위치를 검색한다. 지금, 재고 관리자는 저장소에 있는 제품을 파악하기 위해서 EPC 리스트를 카운트이며 모든 저장소 위치에 있는 제품의 현재 양을 실시간으로 리턴한다.



(그림 4) 시나리오에 대한 순차도

4. 결 론

본 논문에서는 실시간으로 제품 상태 파악을 위해 서비스 정보 시스템을 설계하였다. 서비스 정보 시스템은 유통 과정의 기록, 유통 과정에서 발생하는 오류 상황 감지, 그리고 일반 사용자에 유통 정보를 제공하는 역할을 한다.

이에 RFID는 제품을 인식하는 새로운 방법이 되고 있으며 RFID 기반 시스템은 바코드 시스템 문제에 대한 솔루션을 제공한다. 각각의 유일한 EPC는 각 제품에 대한 정보를 가지고 전송되며 리더가 코드를 판독할 때, 디지털 형식으로 전환되어 EPCIS에 저장된다. 이를 테면 실시간 서비스 정보 시스템은 EPCIS가 정보를 검색하기 위해 접속한다. EPCIS에서 리턴된 결과로는 EPC, 시간, 위치를 출력함으로써 실시간으로 제품 상태와 재고 관리를 제공한다.

이와 같이 실시간 서비스 정보 시스템 설계를 통하여 사용자에게 웹 환경에서 실시간 정보를 확인함으로써 신뢰성과 업무의 효율성을 향상 시킬 수 있도록 하였다. 향후 연구 방향으로는 인증 시스템을 도입하고자 한다.

참고문헌

[1] Magnus Bang, Anders Larsson, Erik Berglund and Henrik Eriksson, "Distributed user interfaces for clinical ubiquitous computing applications", International journal of Medical Informatics, Vol.74, pp. 545-551, 2005.

[2] Willam P. Walsh, Research and application of RFID Technology to enhance aviation security, IEEE, 2000.

[3] *Mark Harrison, EPC Information Service, Cambridge Auto-ID Lab, Institute for Manufacturing, University of Cambridge*

[4] David L. Brock, "The Virtual Electronic Product Code", Auto-ID Center Whitepaper, 2002.