

## 이기종 RFID/센서 디바이스 동적관리를 위한 방법 설계 및 구현

김 휴 찬\*, 고 완 기\*\*, 고 석 용\*\*\*, 양 문 석\*\*\*\*

### Design and implementation of a dynamic management method for heterogeneous RFID/USN devices

Kim Hyu-Chan \*, Koh Seok-Yong \*\*, Koh Wan-Ki \*\*\*, Yang Moon-Seok \*\*\*\*

#### 요 약

최근 u-City 산업의 활성화 및 정부의 '저탄소 녹색성장' 정책의 추진으로 인한 '그린IT' 기술이 주목을 받고 있다. 이를 실현하기 위한 핵심적인 기반 기술인 유비쿼터스 컴퓨팅의 경우, 유비쿼터스 서비스를 제공하기 위해서는 주변 환경 정보 수집을 위한 RFID/USN 미들웨어를 필요로 한다. 또한, 미들웨어는 환경 정보 수집에 따른 다양한 이기종 RFID/센서 디바이스 처리 및 관리 기술이 필요하다. 이에 본 논문에서는 RFID/USN 미들웨어에서 다양한 이기종 RFID/센서 디바이스를 동적으로 관리하기 위한 방법을 제안한다. 제안하는 방법을 이용하여 새로운 RFID/센서 디바이스 추가 및 삭제 등, 미들웨어에서 동적으로 디바이스를 관리함으로써, 상황에 맞는 유비쿼터스 서비스를 제공할 수 있도록 한다.

#### Abstract

Recently "Green IT" technology has been attracting public attention that the Ministry of Environment has come up with new plans to make "low carbon and green growth" technology and to promote u-City industries. Making an offer, Uubiquitous Service needs RFID/USN Middleware to handle heterogeneous RFID/USN devices in order to gather existing informations. In this paper, we propose the design and implementation of a dynamic management method for heterogeneous RFID/USN devices. In compliance with the paper I will furnish Uubiquitous Service regarding existing circumstances by handling heterogeneous RFID/USN devices such as new RFID/Sensor devices addition and deletion and so on.

▶ Keyword : Uubiquitous Service, RFID/USN Middleware, Heterogeneous RFID/USN

• 제1저자 : 김휴찬

• 투고일 : 2009. 07. 10, 심사일 : 2009. 07. 13, 게재확정일 : 2009. 07. 27.

\* 제주한라대학 e-경영정보과 조교수 \*\* 제주한라대학 e-경영정보과 조교수 \*\*\* 제주한라대학 e-경영정보과 부교수  
\*\*\*\* 자바정보기술(주)

## I. 서론

최근 IT기술 동향은 IT기술과 건설이 결합하여 도시 공간 하나하나에 IT기술을 내장, 언제 어디서나 IT기술로 움직이는 미래형 도시인 유비쿼터스 도시(u-City)가 부각되고 있다. 유비쿼터스 도시건설은 IT기술의 새로운 도약과 발전의 의미이자 IT분야의 메가트렌드라고 할 수 있는 기술간 융합(convergence)이 가장 최고조에 도달해 구현될 수 있는 모델의 현실적 구현이라 할 수 있다[1].

또한, 기후 환경 변화가 국제적 이슈로 떠오르고 있는 가운데 전력 소모가 비교적 큰 IT분야가 주목을 받고 있는 상황이다. 최근 들어 '그린IT' 관련 세미나와 관련 보고서들이 증가하고 있으며 세계적으로 선진국을 중심으로 IT분야에 있어 '그린오션' 창출을 위한 움직임이 포착되고 있다. 우리나라 정부도 '저탄소 녹색성장'을 실현하기 위한 국가정보화 전략 하에 '그린IT'부분에만 향후 5년간 약 4조2000억원의 예산을 투입기로 했다[2].

u-City 및 그린IT 전략의 핵심 기술로 유비쿼터스 컴퓨팅이 주목받고 있으며, 유비쿼터스 컴퓨팅은 정보 생성 및 소비의 주체가 사람이었던 인간 중심의 정보화 사회가 사람과 사물뿐만 아니라, 사물 간에도 정보들이 유기적으로 결합되고 활용될 수 있게 변모하고 있다.

유비쿼터스 컴퓨팅은 마크 와이저에 의하여 주장된 이후 이를 실현하기 위한 기술로서 USN (Ubiquitous Sensor Networks)과 RFID (Radio Frequency Identification)에 관한 연구가 활발하게 이루어지고 있다[3]. USN은 필요한 모든 곳에 센서를 부착하고, 이를 통해 사물의 인식 정보를 기본으로 온도, 압력, 오염, 균열 등의 주변 환경 정보까지 각종 센서를 통하여 실시간 수집하여 관리, 통제할 수 있도록 구성된 네트워크이다[4]. RFID는 각종 물품에 전자 태그를 부착하여 사물의 정보와 주변 환경 정보를 무선 주파수로 전송·처리하는 비접촉식 인식 기술이다[5].

유비쿼터스 컴퓨팅의 핵심 기술인 RFID/USN 기술을 이용하여 행정, 의료, 교통, 물류, 정보가전, 환경, 재난 방재 분야에 적용함으로써 생활의 편의 증대, 삶의 질 향상, 복지 향상, 그리고 안전보장을 추구하고 있다[6].

이와 같이 유비쿼터스 컴퓨팅 기술을 적용하여 u-City 및 그린IT 등 여러 분야에 맞추어 유비쿼터스 서비스를 제공하기 위해서는 서비스에 맞는 다양한 주변 환경 정보의 수집이 필요하다.

주변 환경 정보를 획득하기 위해서는 센서 데이터 수집용

미들웨어가 필요하며, 미들웨어는 다양한 센서 데이터 처리, 센서 디바이스를 관리할 수 있는 기능이 있어야 한다.

본 논문에서는 이와 같은 다양한 센서 데이터 처리 및 센서 디바이스 관리를 위하여 미들웨어에서 이기종의 센서 데이터 처리 및 센서 디바이스 관리를 위한 방법을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련연구 및 기술로 RFID/USN 소프트웨어 기술 및 RFID/USN 미들웨어를 분석하고, 3장에서는 제안하는 방법을 적용한 RFID/USN 미들웨어 및 RFID/센서 디바이스 관리 방법에 대해 설명한다. 4장에서는 구현 및 실험 결과를 살펴보고, 마지막 5장에는 결론을 기술한다.

## II. 관련연구 및 기술

### 2.1 RFID/USN 소프트웨어 기술

유비쿼터스 응용 서비스를 쉽게 구축할 수 있으려면 RFID/USN 하드웨어와 응용 애플리케이션 또는 엔터프라이즈 시스템을 연결해 줄 수 있는 미들웨어 기능, 즉 RFID/USN 소프트웨어 기술을 필요로 한다[7].

RFID/USN 미들웨어는 대표적인 RFID/USN 소프트웨어 기술로서, 일반적으로 RFID 리더 장치 및 센서와 응용 애플리케이션을 연결하는 시스템 소프트웨어를 지칭한다.

RFID/USN 미들웨어는 다수의 이기종 RFID/센서 디바이스에서 수집된 대량의 데이터를 필터링하고 이벤트 데이터를 처리한 뒤, 이를 의미 있는 정보로 요약하여 서비스 시스템에 전달할 수 있어야 한다[8].

RFID/USN 기술이 응용 분야에 효과적으로 활용되기 위해서는 각각의 사물을 식별하기 위해 사물에 부착되는 RFID 태그와 이를 인식하기 위한 RFID 리더 장치, 그리고 센서 디바이스들로부터 수집된 데이터를 처리하여 응용 애플리케이션에게 유용한 정보를 서비스를 제공할 수 있는 RFID/USN 소프트웨어가 유기적으로 구축되어야 한다[9].

즉, RFID/USN 기술 도입을 통해 자동 데이터 수집에 소요되는 비용을 줄이고, 기존 업무에 대한 프로세스 개선 및 신규 서비스 창출 등의 목적을 효과적으로 달성하기 위해서는, 단지 우수한 성능의 리더 장치 및 센서 디바이스 설치 도입이나, 기존 엔터프라이즈 시스템의 확장만으로는 해결할 수 없다[10].

현재 RFID/USN 시스템 구축 및 기술 개발은 주로 물류, 유통서비스를 대상으로 한 수동형 RFID 기술을 중심으로,

특정 분야의 정보를 제한적으로 공유하는 폐쇄형 형태로 구축되고 있으며, 그 결과 RFID/USN 미들웨어 또한 특정 기업 및 일부 분야에 특화되어 매년 새롭게 설계 및 개발되고 있는 형편이다. 따라서 RFID/USN 디바이스를 통한 데이터 접근이나, 디바이스로부터 수집된 데이터의 저장 및 처리와 같은 기본 기능에만 집중되어 있고 특정 하드웨어 및 응용 애플리케이션에 매우 종속적이다.

이와 같은 문제를 해결하기 위하여 RFID/USN 미들웨어는 다양한 형태의 디바이스 인터페이스, 다양한 데이터 및 네트워크 연동, 여러 가지 응용 플랫폼에 대해서도 상호 운용성을 보장할 수 있어야 한다. 이를 위해 적어도 다음의 요건들이 만족되어야 한다[11].

가. 이기종 RFID/USN 시스템 지원 및 관리

RFID/USN 미들웨어는 다수의 이기종 RFID/USN 시스템간의 이질성(예를 들면, 지원하는 디바이스와 프로토콜의 상이함, 디바이스와 호스트 시스템간의 네트워크 인터페이스의 다양성 등)이 존재하는 환경 하에서, RFID/USN 하드웨어 시스템을 상위계층에서 일관되게 접근이 가능하도록 기능을 제공해야 한다. 데이터 정보 수집, 디바이스 설정 및 원격 제어, 디바이스 시스템 모니터링 등의 관리가 이뤄질 수 있도록 기능이 제공되어야 한다.

나. 데이터 처리

RFID/USN 데이터가 리더 및 센서로부터 반복적으로 대량의 정보가 유입됨에 따라서, RFID/USN 미들웨어는 이러한 중복된 정보 및 응용 시스템 계층에 불필요한 정보들을 필터링하고 요약하는 기능을 제공해야 한다.

다. 응용 시스템과의 연동

RFID/USN 미들웨어는 정제, 요약된 데이터를 데이터 수요자인 기존 응용 시스템에 신뢰성 있게 전송할 수 있는 기능을 제공해야 한다.

즉, RFID/USN 미들웨어는 리더 및 센서에서 계속적으로 발생하는 데이터를 수집, 제어, 관리하는 기능을 하며, 모든 구성요소와 연결되어 계층적으로 조직화되고 분산된 구조의 미들웨어 네트워크를 구성하여 서로 통신 하도록 한다.

2.2 RFID/USN 미들웨어

최근 RFID/USN 미들웨어 관련 연구는 다음과 같다.

Oracle의 Sensor Edge Server[12]는 데이터 수집, 이벤트 처리, 데이터 분배(dispatching) 등의 기능을 지원하는 센서 기반 서비스 통합 플랫폼이다. 미리 정의된 필터들을 이

용하여 응용이 원하는 데이터를 추출하며, 보다 복잡한 데이터를 처리하고자 할 경우 직접 로직 필터를 작성할 수 있다.

CapTech 사의 TagsWare[13]는 RFID 태그 데이터를 응용에 전달하기 위한 링크, RFID 장비로의 표준 인터페이스를 지원하는 드라이버, 그리고 응용에서 링크와 드라이버의 사용을 지원하는 응용 기반 요소들로 구성된다. 리더 장치로부터 수집된 원시 데이터에서 태그 데이터를 추출하여 평활화(smoothing)한 후 응용에 전달하기 위한 링크 컴포넌트가 체인으로 연결된 구조를 제공한다.

SUN 사의 Java System RFID 소프트웨어[14]는 다양한 센서로부터 오는 센서 데이터 스트림을 이벤트 관리기에서 처리하고, 리더 어댑터, 필터, 로거(logger), 엔터프라이즈 게이트웨이 등으로 구성된다. 델타(delta)와 평활화 질의를 할 수 있고, 필터들을 서로 연결하여 특정 마스크(mask) 조건을 만족하는 EPC 데이터 값을 얻을 수 있을 뿐만 아니라 사용자 정의 필터를 개발할 수 있는 도구를 지원한다.

ETRI의 UbiCore 시스템[15]은 XML 기반 미들웨어 시스템으로 다양한 센서를 관리하며, XQueryStream이라는 XQuery에 기반한 연속 질의 언어를 제공한다. 필터링과 중간 결과 재사용을 통하여 스트림 데이터에 대한 질의 처리 속도를 개선하였고, 연속적으로 생성되어 들어오는 실시간 데이터뿐만 아니라 저장된 이력 데이터에 대한 질의를 지원하며 컨텍스트와 서비스의 연계 정보를 표현하기 위한 마크업 언어인 CSML(Context-driven Service Markup Language)를 제공한다. UbiCore[16]의 새로운 버전에서는 다양한 센서를 관리하고, 센서에서 센싱한 데이터를 수집하고, 수집된 데이터를 처리하여 자동적으로 어플리케이션에 제공하는 방법을 제안하였다.

RFID 미들웨어 플랫폼인 REMS(RFID Event Management System)과 RBPTS(Real-time Business Process Triggering System)[17]에서는 ALE와 호환성이 있을 뿐만 아니라 수동적인 리더와 능동적인 리더 등 다양한 타입의 리더를 처리할 수 있다.

ESN(EPC Sensor Network) 아키텍처[18]는 EPCglobal 아키텍처에 기반하여 분산되어 있는 리더와 센서로부터 실시간으로 다양한 데이터를 처리하기 위해 CEP(Complex Event Processing)를 이용하여 RFID와 WSN(Wireless Sensor Network)을 통합하는 방법을 제안하였다.

이와 같은 연구에서는 리더 및 센서와의 인터페이스 연결 및 데이터 처리 기능 등 미들웨어 성능 개선 및 향상에 대한 연구가 대부분이고 리더 및 센서 디바이스를 효율적으로 관리하기 위한 방법에 관한 연구는 미미한 실정이다.

다양한 유비쿼터스 서비스 제공을 위해서 RFID/USN 미

들웨어는 다양한 형태의 리더 및 센서 디바이스 인터페이스, 다양한 데이터 및 망 연동, 여러 가지 응용 플랫폼에 대해서도 상호운영성을 보장할 수 있어야 한다.

본 논문에서는 상호운영성 및 디바이스 확장성을 고려하여 미들웨어를 수정하지 않고서도 이기종 RFID/센서 디바이스를 처리 및 관리할 수 있는 방법을 제안한다.

### III. RFID/센서 디바이스 동적관리 방법

#### 3.1 RFID/센서 동적 관리를 위한 미들웨어 개요

본 논문에서 제안하는 방법은 미들웨어에서 RFID/USN 디바이스를 동적으로 관리하기 위한 것으로, 유비쿼터스 서비스를 제공하기 위하여 상황에 맞는 RFID/USN 디바이스의 추가 및 삭제 등 디바이스 변경을 용이하게 관리하기 위한 방법이다. 미들웨어 관리자 및 사용자는 관리 도구를 이용하여 RFID/USN 디바이스 정보를 관리하고 변경된 정보는 동적으로 미들웨어로 통지하여 미들웨어에서 변경된 사항을 적용하도록 하는 시스템 및 방법을 제공하는데 있다.

본 논문에서 제안하는 RFID/센서 디바이스 동적관리 방법을 적용하는 미들웨어의 전체적인 구조는 다음과 같다.

그림 1은 미들웨어 구조로서, 미들웨어 주요 기능을 세분화하고 인터페이스를 정의하여 모듈화하여 구성한 미들웨어 프레임워크를 나타낸다.

미들웨어 프레임워크의 구성에 포함되는 주요 모듈로는 전체적인 미들웨어 주요 기능을 관리하는 Control Manager, 미들웨어 구동 시 필요한 기능 및 모듈을 실행하기 위하여 환경설정 정보를 읽어서 모듈을 실행하는 Configure Manager가 있다.

또한, 입력되는 태그 및 센서 데이터를 저장하는 큐를 생성 및 관리하는 Queue Manager, 데이터베이스 관리를 위한 Database Manager, 각 모듈의 기능에 필요한 유틸 등을 관리하는 Util Manager 등으로 구성된다.

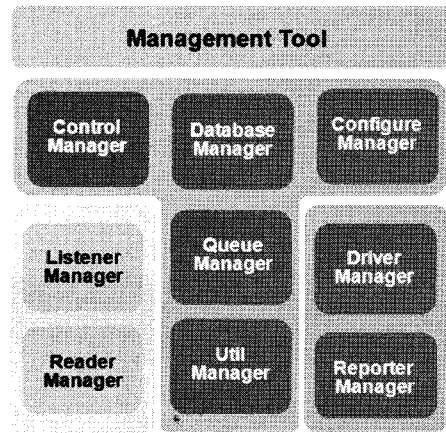


그림 1. 미들웨어 구성 모듈  
Fig. 1. Configuration of middleware system.

미들웨어에서 처리된 데이터 결과를 전송하기 위한 Reporter Manager와 미들웨어 관리도구와 연동하며 RFID/센서 디바이스와 연결을 위하여 제안하는 방법에서 사용하는 드라이버 등을 관리하는 Driver Manager를 포함한다.

마지막으로 디바이스와의 연결 및 데이터 수집을 위한 Reader Manager와 Listener Manager로 구성된다.

#### 3.2 RFID/센서 관리 방법

미들웨어의 디바이스 인터페이스 기술은 미들웨어 중요 요소 기술로서, RFID의 경우 이를 위하여 표준화 작업이 이루어지고 있으나, 현재까지는 리더 제조사에 의해 프로토콜이 정의되어 사용되어지고 있는 경우가 대부분이며, 센서의 경우에도 센서의 따라 다양한 프로토콜이 사용되어 지고 있다. 또한, RFID/센서 디바이스는 다양한 환경 정보 수집을 위하여 새로이 연구 및 개발되어지고 있다.

본 논문에서는 이와 같이 다양한 프로토콜을 사용하는 RFID/센서 디바이스 및 새로이 디바이스 추가 및 삭제 시 이를 관리하기 위한 방법으로 다음과 같은 방법을 제안한다.

디바이스 인터페이스를 관리하는 역할을 미들웨어 프레임워크의 하나의 모듈로 설계하고, 디바이스 인터페이스를 정의하여 새로이 추가되는 디바이스마다 인터페이스(드라이버)를 구현하여 미들웨어 관리 도구를 이용하여 등록하도록 한다.

미들웨어 관리 도구는 동적으로 디바이스를 관리 가능하도록 하고, 디바이스 정보의 변경 시 미들웨어에 통지하여 변경된 사항을 적용하도록 한다.

이를 위하여 디바이스 정보 관리를 위한 환경 설정에 필요한 정보를 XML 문서로 정의하고, 인터페이스를 구현한 클레

스(드라이버)를 관리하는 디바이스 관리 모듈을 설계한다.

RFID/USN 디바이스 정보를 관리하기 위하여 다음 그림 2와 같이 XML을 정의하여 관리한다.

그림 2는 디바이스 관리를 위한 정보를 기술하며, 미들웨어에서는 디바이스 정보를 논리적 개념을 도입한 논리리더와 실질적인 리더정보인 물리리더의 개념을 사용한다. 이와 같은 방법을 사용하는 이유는 디바이스에 종속되지 않도록 사용하기 위한 방법으로 응용 애플리케이션에서는 사용되어 지고 있는 실질적인 디바이스 정보를 알지 않고서도 논리적 개념인 논리리더를 이용하여 데이터를 요청하여 사용한다.

```

<Caru-EM>
  <Logical-Reader>
    <Reader-Name>test</Reader-Name>
    <Queue-Size>10</Queue-Size>
    <Physical-Readers>
      <Physical-Reader-Name>aaa</Physical-Reader-Name>
    </Physical-Readers>
  </Logical-Reader>
  <Physical-Reader-List>
    <Physical-Reader>
      <Reader-Name>physicalReader1</Reader-Name>
      <Reader-Explanation>test</Reader-Explanation>
      <IPAddress>127.0.0.1</IPAddress>
      <Port>10000</Port>
      <Rssi>100</Rssi>
      <Driver-Class>KiscomReader</Driver-Class>
    </Physical-Reader>
  </Physical-Reader-List>
</Caru-EM>
    
```

그림2. 미들웨어 RFID/USN 디바이스 정보 XML 문서  
Fig. 2. XML of RFID/USN devices information

Physical-Reader는 실질 디바이스 정보를 나타내며, 실제 디바이스의 이름, 네트워크 연결을 위한 IP주소, 포트번호 및 디바이스 인터페이스를 구현한 드라이버(클래스 파일)를 정의한다.

Logical-Reader는 논리적 개념을 도입한 디바이스로 데이터를 전송하는 실제 디바이스 정보와 데이터를 저장할 큐 사이즈 정보를 정의한다.

이와 같이 디바이스 정보를 XML로 정의하여 미들웨어는 XML을 토대로 디바이스를 연결 및 통신 등을 한다. 또한 동적으로 변경되는 디바이스 정보는 미들웨어 관리 도구를 이용하여 미들웨어로 통지하고 변경된 정보를 XML에 업데이트 하도록 한다.

XML 정보는 미들웨어 관리 도구를 이용하여 등록 및 삭제 관리 메뉴를 이용하여 미들웨어 관리자가 작성 및 업데이트 한다.

미들웨어 관리자는 리더와의 연결을 위하여 드라이버 변경 및 업데이트 또는 새로이 드라이버를 웹 기반의 관리 도구를 이용하여 드라이버 및 리더 정보를 관리한다.

관리 도구는 변경되는 리더 정보 및 드라이버 등을 미들웨어 시스템에 통지하여 변경된 사항을 미들웨어에 적용하여 동작이 가능하도록 한다.

그림 3는 관리자 및 사용자가 미들웨어 관리 도구를 이용하여 리더 정보 및 드라이버를 관리하는 순서도를 나타낸다.

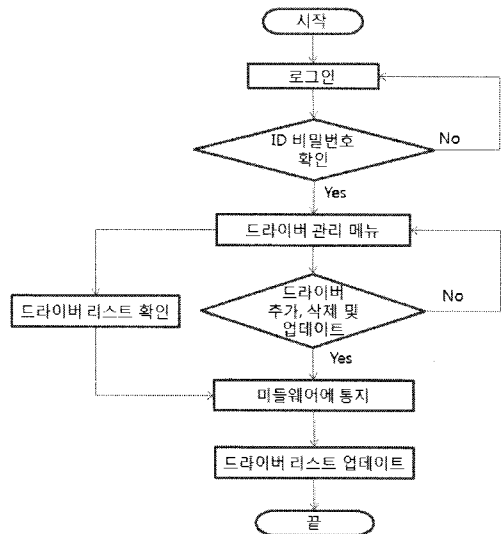


그림3. RFID/USN 디바이스 정보 확인 및 등록 과정  
Fig. 3. XML of RFID/USN devices information

RFID/USN 디바이스 관리를 위한 시스템 구성은 미들웨어 시스템과 미들웨어 관리 도구로 구성되며 미들웨어 관리자 및 사용자는 웹 기반의 미들웨어 관리 도구에 로그인하여 드라이버 관리 메뉴를 이용하여 현재 등록되어 있는 디바이스 정보 및 추가, 삭제, 업데이트를 할 수 있고, 관리 도구는 미들웨어 시스템에 변경된 정보를 전송한다. 미들웨어는 변경된 정보를 시스템에 적용하고 디바이스 정보를 XML로 기록 및 관리한다.

또한 디바이스 드라이버의 경우, 어댑터 모델을 적용하여 디바이스 접속을 위한 필요 기능(메서드)을 정의하여 인터페이스를 정의하여 드라이버 제작 시 인터페이스를 구현하도록 설계한다. 관리자는 관리 도구를 이용하여 각각의 디바이스에 맞는 드라이버 파일을 미들웨어에 전송하여 등록한 후, 사용 하도록 한다.

## IV. 구현 및 실험

### 4.1 RFID/센서 관리 모듈의 클래스 다이어그램

RFID/센서 디바이스 관리를 위한 주요 클래스들은 그림 4와 같다.

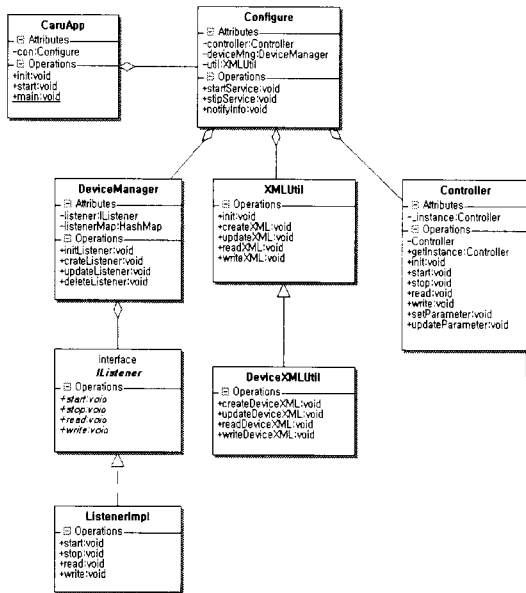


그림 4. 주요 클래스의 클래스 다이어그램  
Fig. 4. Class diagram

Configure Manager 클래스는 미들웨어 시스템을 구성하는 주요 모듈 및 필요한 기능을 관리하며 실행하는 메인 클래스로서 미들웨어 구동 시 미들웨어 시스템에 필요한 기능 및 모듈 정보를 XMLUtil 클래스의 객체를 생성하여 미들웨어 시스템 정보를 관리하는 XML 파일에서 모듈 정보를 읽어온 후, 주요 모듈을 실행한다.

Controller 클래스는 미들웨어 관리 도구와 연결되어 관리 도구에서 요청되는 사항을 미들웨어에 적용하기 위한 정보를 관리하는 싱글톤 패턴을 적용한 클래스로서, Configure Manager에 의해 생성되어 동적으로 변경되는 디바이스 정보 및 미들웨어 정보를 관리한다.

Device Manager는 미들웨어와 연결되어 사용되는 디바이스를 관리하는 클래스로서, 미들웨어 관리 도구를 이용하여 등록된 디바이스 정보와 연결된 디바이스를 맵핑하여 디바이

스와 미들웨어를 연결한다.

미들웨어와 통신을 위하여 이용되는 디바이스 드라이버의 경우, Listener 인터페이스를 구현한 클래스(ListenerImpl)를 지칭하며, 미들웨어 관리 도구를 이용하여 드라이버를 등록 및 삭제 등을 관리한다.

### 4.2 RFID/센서 관리 모듈 실행 및 결과

미들웨어에 새로이 디바이스를 등록하고 적용하는 과정은 다음과 같다.

미들웨어 관리자는 IListener 인터페이스를 구현한 클래스(드라이버)를 관리 도구의 드라이버 메뉴의 기능을 이용하여 드라이버를 등록하고, 미들웨어는 관리 도구의 디바이스 등록 요청을 받으면 Controller에 의해 디바이스 정보를 관리하는 Device Manager에 의해 동적으로 설정이 변경된 드라이버를 이용하여 리스너를 실행하여 디바이스와 통신을 하게 된다. 리스너를 실행한 후, XMLUtil 클래스를 이용하여 디바이스 정보 XML를 업데이트 한다.

다음 그림은 미들웨어 관리 도구를 이용하여 관리자에 의하여 디바이스 드라이버 추가하였을 경우, 미들웨어에서 디바이스를 연결, 실행하는 과정을 보여준다.

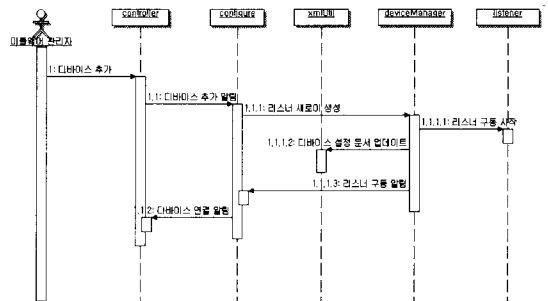


그림 5. 시퀀스 다이어그램  
Fig. 5. Sequence diagram

그림 5는 미들웨어 관리 도구를 이용하여 등록되어 있는 드라이버를 이용하여 새로이 디바이스를 추가하였을 경우를 나타내는 시퀀스 다이어그램이다.

관리 도구에서 디바이스 추가 요청이 들어오면 controller는 요청 정보를 적용하기 위하여 디바이스가 추가되었음을 configure에 알리고 configure는 device manager에게 통지하면 device manager는 요청에 맞는 드라이버를 맵핑하여 디바이스와 통신을 위한 리스너를 실행한다. 리스너가 실행된 이후, 디바이스 관리 문서를 업데이트하고 관리자에게 디바이스와 연결됨을 통지한다.

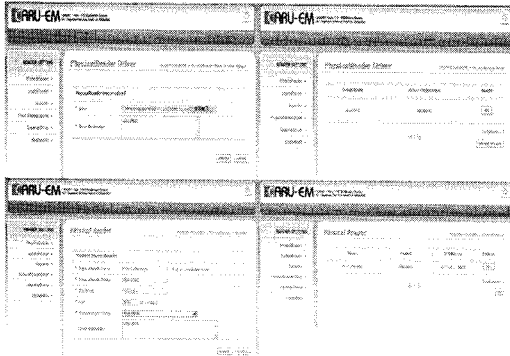


그림 6 미들웨어 관리 도구

Fig. 6. Middleware management tool of web based for managing RFID/USN devices

그림 6은 미들웨어 관리 도구에서 디바이스 드라이버 및 디바이스 등록 과정의 모습을 보여준다.

좌측 상단의 화면은 새로운 디바이스를 사용하기 위하여 드라이버를 등록하는 과정을 나타내며, 우측 상단은 등록되어 있는 디바이스 드라이버 리스트를 나타낸다.

좌측 하단의 화면은 등록된 드라이버를 이용하여 디바이스와 연결하기 위한 디바이스를 등록하는 모습을 나타내며, 우측 하단의 화면은 등록된 디바이스 리스트를 나타낸다.

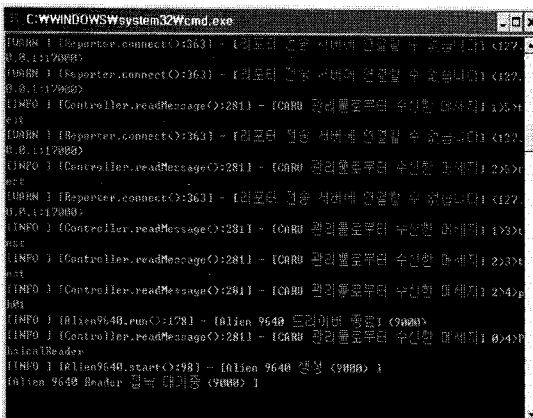


그림 7. 미들웨어 실행 화면

Fig. 7.

그림 7은 이제까지 설명한 디바이스 등록 절차를 마쳤을 경우, 미들웨어에서 변경된 디바이스 정보를 시스템에 적용하여 연결된 디바이스와의 통신을 위해 대기하는 모습을 보여준다. 이와 같이 본 논문에서 제안하는 방법은 미들웨어 시스템의 디바이스 관리 모듈과 미들웨어 관리 도구를 이용하여

RFID/USN 디바이스를 관리한다. RFID/USN 디바이스를 동적으로 관리하여, 변경된 사항을 미들웨어 수정 없이 적용할 수 있도록 함으로써 향후 다양한 디바이스를 적용할 수 있도록 확장성을 제공한다.

## V. 결론

본 논문에서는 상호운영성 및 디바이스 확장성을 고려하여 미들웨어를 수정하지 않고서도 이기종 RFID/센서 디바이스를 처리 및 관리할 수 있는 방법을 제안한다.

유비쿼터스 서비스를 성공적으로 구현하기 위해서는 사용자와 응용 환경에 맞는 사용자 중심의 적절한 기술 적용이 필요하며, 특히 u-City 건설 및 유비쿼터스를 이용한 비즈니스 모델 등 서비스들이 활성화되기 위해서는 다양한 비즈니스를 제공하기 위해 다양한 이기종 RFID/센서의 활용은 필수적이다.

이러한 서비스를 제공하기 위하여 미들웨어 단에서 다양한 형태의 리더 및 센서 디바이스 인터페이스, 다양한 데이터 및 망 연동, 여러 가지 응용 플랫폼에 대해서도 상호운영성을 보장할 수 있어야 한다.

본 논문에서 제안하는 방법과 같이 디바이스를 동적으로 관리할 수 있는 방법 이용하여 새로운 RFID/센서가 추가적으로 운영되어도 미들웨어 수정 없이 사용함으로써, 이기종 RFID/센서를 쉽게 연결할 수 있는 범용성 및 비용 절감의 효과를 가질 수 있다.

향후, 다양한 유비쿼터스 서비스 제공을 위한 RFID/USN 디바이스 개발은 계속적으로 이루어질 전망이며, 이에 맞추어 RFID/센서 디바이스를 동적으로 관리하여 사용함으로써, 미들웨어는 다양한 디바이스를 사용할 수 있는 확장성 및 상호운영성을 제공할 수 있다.

## 참고문헌

- [1] 조병선, 정우수, 조항숙, "u-City 사업전개와 추진동향," ETRI, 전자통신동향분석 제21권 4호, 2006.08
- [2] '그린IT 활용,' 정책연구센터, 한국소프트웨어 진흥원, 2008.
- [3] M. Weiser, J. S. Brown, "The coming of age of calm technology," Xerox PARC, 1996
- [4] 송석현, 신상철, "RFID/USN 표준화 동향 및 이슈," 정보과학회지, 제22권 제2호, 2004.
- [5] 장병준, 안선일, 이윤덕, "RFID/USN 기술개발 동향,"

- 한국정보학회, 정보과학회지, 제23권 제2호, 2005.
- [6] 황재각, 표철실, "USN 미들웨어 기술 개발 동향," 한국전자통신연구원, 한국전자과학회지, Vol.19 제6호 2008.
- [7] 홍상균, 정태수, 김영일, 이용준, "RFID 미들웨어 기술 동향 및 응용," ETRI 전자통신동향분석 제20권 제3호, June 2005.
- [8] 황재각, 정태수, 김영일, 이용준, ETRI, "RFID 미들웨어 기술 동향 및 응용," 전자통신동향분석, 제20권 제3호, 2005.
- [9] 오세원, 박주상, 이용준, "RFID SW기술과 표준화 동향," 한국통신학회지, 제24권 제7호, 2007.
- [10] Nicholas D. Evans "Middleware Is the Key to RFID," RFID Journal, April 2004.
- [11] 성종우, 김대영, "RFID와 USN 통합 인프라스트럭처를 위한 EPC 센서 네트워크," 한국통신학회 논문지, 제23권 제12호, 2006.
- [12] Enterprise Information Architecture for RFID and Sensor-Based Services, Oracle White Paper, 2006.
- [13] TagsWare, Agile RFID Solution, <http://www.tagsware.com>.
- [14] A. Gupta, M. Srivastava, "Developing Auto-ID Solutions using Sun Java System RFID Software", 2004.
- [15] 이훈순, 최현화, 김병섭, 이명철, 박재홍, 이미영, 김명준, 진성일, "UbiCore : XML 기반 RFID 미들웨어 시스템," 정보과학회논문지, 제33권, 제6호, pp. 578-589, 2006.
- [16] H. S. Lee, S. I. Jin, "An Effective XML-Based Sensor Data Stream Processing Middleware for Ubiquitous Service," Lecture Note in Computer Science, Vol. 4704, Springer, Heidelberg, pp.844-857, 2007.
- [17] T. S. Cheong, Y. G. Kim, Y. J. Lee, "REMS and RBPTS: ALE-compliant RFID Middleware Software Platform," International Conference on Advanced Communication Technology, pp. 699-704, 2006.
- [18] W. Wang, J. W. Sung, D. Y. Kim, "Complex Event Processing in EPC Sensor Network Middleware for Both RFID and WSN," 11th IEEE Symposium on Object Oriented Real-Time Distributed Computing, pp. 165-169, 2008.

## 저자 소개



### 김휴찬

1993 : 제주대학교 에너지공학과(학사)  
 1997 : 제주대학교 대학원 에너지공학(공학석사)  
 2002 : 제주대학교 대학원 컴퓨터공학과(박사수료)  
 1999 ~ 현재 : 제주한라대학 e-경영정보과 조교수  
 관심분야 : RFID 응용, e-비즈니스 모델, 최적화



### 고석용

1993년 제주대학교 경영학과(경영학사)  
 1994년 한국의국어대학교 경영정보학과(경영학 석사)  
 1998 한국의국어대학교 경영정보학과(경영학 박사)  
 1997~현재 제주한라대학 e-경영정보학과 교수  
 2008~현재 제주한라대학 기획처장  
 관심분야 : 데이터베이스, RFID 비즈니스 모델



### 고완기

1989 동국대학교 전자계산학과(공학사)  
 1994 조선대학교 전자계산학과(공학석사)  
 2003 제주대학교 경영학과(경영학과(박사수료))  
 1997~ 현재 : 제주한라대학 e-경영정보과 교수  
 관심분야 : 데이터베이스, 생산이력추적시스템, 전자상거래



### 양문석

2003년 제주대학교 전자공학과 학사  
 2009년 제주대학교 컴퓨터공학과 석사  
 2009년~ 현재 자바정보기술(주)  
 관심분야 : 유비쿼터스 미들웨어, 센서 미들웨어, 지능형 컴퓨팅