

---

# 온톨로지 기반 EPC 코드 자동 변환 방법

노영식\* · 변영철\*\*

A method for automatic EPC code conversion based on ontology methodology

Young-Sik Noh\* · Yung-Cheol Byun\*\*

---

본 연구는 산업자원부와 한국기술재단의 지역혁신인력양성사업으로 수행된 연구결과임

---

## 요 약

ALE 기반 RFID 미들웨어는 리더 장치로부터 EPC 데이터를 입력받아 내부적으로 URN 형태로 변환하고 이를 필터링, 그룹핑 등을 수행한 후 응용으로 전송한다. 한편, EPC 데이터의 경우 유형이 다양할 뿐만 아니라 향후 또 다른 새로운 형식의 EPC 데이터가 제안될 수도 있으므로 RFID 미들웨어는 다양한 유형의 EPC 데이터를 효율적으로 처리할 수 있어야 한다. 본 논문에서는 EPCglobal의 ALE 표준 스펙을 기반으로 한 RFID 미들웨어에서 RFID 리더로부터 수집된 다양한 유형의 EPC 데이터를 효율적으로 처리하기 위하여 온톨로지 기반의 데이터 처리 방법을 제안한다. 즉, 다양한 유형의 EPC 데이터를 URN 형태로 효과적으로 변환하기 위하여 데이터 유형별 변환 규칙을 온톨로지로 구축한다. 그럼으로써 온톨로지의 재사용은 물론 새로운 유형의 EPC 데이터가 제안되어도 해당 EPC 데이터에 대한 온톨로지만 추가함으로써 미들웨어로 하여금 효과적으로 처리할 수 있도록 확장할 수 있다.

## ABSTRACT

ALE-compliant RFID middleware system receives EPC code data from reader devices and converts the data into URN format data internally. After filtering and grouping, the system sends the resulting URN code to application and(or) users. Meanwhile, not only the types of EPC code are very diverse, but also another new kinds of EPC code can be emerged in the future. Therefore, a method to process all kinds of EPC code effectively is required by RFID middleware. In this paper, a method to process various kinds of EPC code acquired from RFID reader devices in ALE-compliant RFID middleware is proposed. Especially, we propose an approach using ontology technology to process not only existing EPC code but also newly defined code in the future. That is, we build an ontology of conversion rules for each EPC data type to effectively convert EPC data into URL format data. In this case, we can easily extend RFID middleware to process a new EPC code data by adding a conversion rule ontology for the code.

## 키워드

ALE, RFID Middleware, Ontology, EPC, URN, Code Conversion

## I. 서 론

RFID(Radio Frequency Identification) 기술은 비접촉식

데이터 획득 기술로서 상황 인식 기술과 더불어 유비쿼터스 컴퓨팅 실현을 위하여 필요한 기술 중 하나로 여겨지고 있다[1,2]. 또한 데이터의 수집, 가공, 전달, 장치 제

---

\* 제주대학교 대학원 컴퓨터공학과

접수일자 : 2008. 2. 11

\*\* 첨단기술연, 제주대학교 컴퓨터공학과 교수, 교신저자

어 및 관리 등을 수행함으로써 응용 서비스 개발자로 하여금 RFID 기술을 이용한 유비쿼터스 응용을 쉽게 구축할 수 있도록 교량 역할을 하는 RFID 미들웨어 대한 연구 개발이 수행되고 있다.

사실상 RFID 기술의 국제 표준을 이끌고 있는 EPCglobal에서 제안하고 있는 EPCglobal Class 1 Gen. 2는 900 MHz 대역의 사실상의 표준일 뿐만 아니라 ISO 18000-6 C로서 국제 표준으로 채택되었다. 또한 EPCglobal은 EPC 네트워크 표준안을 제안하였고, RFID 미들웨어 플랫폼을 개발하였거나 개발하려는 많은 업체들이 이를 표준으로 수용하고 있다[3].

EPC 네트워크는 RFID 미들웨어와 관련하여 2002년에 구현 스펙 중심의 Savant Version 0.1을 제안하였으며 [4,5], 2004년에는 인터페이스 중심의 ALE(Application Level Event)를 제안하였다[6]. ALE는 EPC 정보의 필터링 및 수집 역할을 담당하는 미들웨어이며, 아키텍처의 구성 요소로서 리더와 태그의 에어(air) 인터페이스, EPCIS, EPC 검색 서비스, ONS 등으로 구성된다.

RFID 미들웨어는 리더 장치로부터 EPC 데이터를 입력받을 경우 이를 처리하기 위하여 내부적으로 URN 형태의 데이터로 변환하고 이를 이용하여 필터링과 그룹핑 등을 수행한 후 그 결과를 응용으로 전송한다. 한편, EPC 데이터의 경우 유형이 다양할 뿐만 아니라 향후 또 다른 새로운 형태의 EPC 데이터 유형이 나타날 수도 있다. 따라서 RFID 미들웨어는 다양한 유형의 EPC 데이터를 효율적으로 처리할 수 있어야 한다.

본 논문에서는 EPCglobal의 ALE 표준 스펙을 기반으로 한 RFID 미들웨어에서 RFID 리더로부터 수집된 다양한 유형의 RFID EPC 데이터를 효율적으로 처리하기 위한 방법을 제안한다. 현재 존재하는 EPC 데이터뿐만 아니라 새롭게 정의되는 EPC 데이터를 효과적으로 처리하기 위하여 온톨로지 기반 방법을 제안한다.

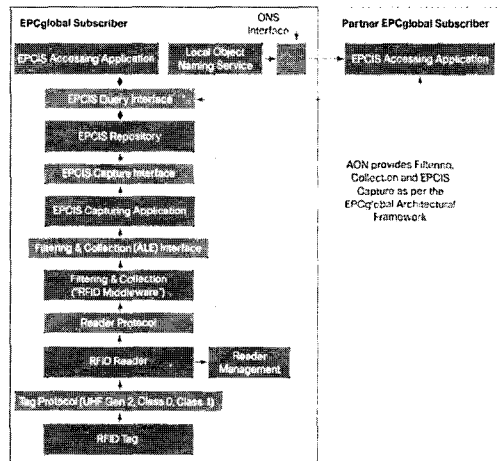
본 논문의 구성은 다음과 같다. 우선 II장에서 RFID EPC 코드 체계, RFID 미들웨어, 메타 데이터 기술을 분석하고, III장에서는 RFID 미들웨어가 온톨로지를 이용하여 EPC 데이터를 변환 처리하는 방법에 대하여 설명한다. IV장에서는 제안하는 방법을 구현하여 실험한 후 결과를 분석하고, 마지막 V장에서는 본 연구에 대한 결론을 맺는다.

## II. 관련 연구 및 기술

### 2.1. EPCglobal 아키텍처

EPCglobal에서는 개별 물체의 유일 식별자인 EPC 기반의 'Internet of Physical Objects'를 구성하기 위한 기술 집합을 EPC 네트워크라 정의하고 있다. 구성 요소로는 리더와 태그 사이의 정보 교환과 관련한 리더 프로토콜, 리더 제어 및 관리를 위한 표준 기능 목록 정의, 인식된 태그 정보로부터 정제 및 요약된 데이터를 얻기 위하여 필요한 데이터 표현 방식과 정보 제공을 위한 표준 규격을 정의하는 필터링과 콜렉션(collection), 외부 인터페이스를 정의하는 ALE, EPC와 관련된 정보를 획득하고 관리하며 이를 공유하기 위한 외부 인터페이스를 정의하는 EPCIS 등이 있다.

또한 EPC와 관련한 EPCIS 위치 정보 검색 서비스를 제공하는 ONS, EPC 인코딩/디코딩 규칙을 표현하고, 그 규칙에 근거하여 EPC 코드 변환 작업을 수행하는데 필요한 규격을 정의하는 태그 데이터 변환, 사용자 정보 보호, 데이터 암호화 등 EPC 네트워크 전반에 걸친 보안 프레임워크를 제공하기 위한 일종의 가이드라인을 정의하는 내용을 중심으로 표준 제정을 진행하고 있다[3,7]. 그림 1은 EPCglobal 네트워크 아키텍처 프레임워크 구조이다.



ALE = application level events  
EPCIS = electronic product code information system

그림 1. EPCglobal의 아키텍처 프레임워크  
Fig 1. The framework of EPCglobal architecture

EPCglobal에서 제안하고 있는 아키텍처는 RFID 데이터를 관리함에 있어 체계적인 구조를 갖고 있으나, RFID 태그와 RFID 리더 사이의 태그 데이터를 관리함에 있어 현존하는 다양한 유형의 EPC 데이터뿐만 아니라 새롭게 나타날 수 있는 EPC 데이터를 처리하기 위한 방법에 대해서는 기술하고 있지 않다.

### 2.2. EPC 코드 체계

EPCglobal에서는 태그 데이터에 해당하는 EPC 태그 데이터 스펙 버전 1.24, 1.27[8], 1.3[9]에서 클래스 0, 클래스 1 Gen1, Gen2 등을 제안하고 있으며, 향후 발전하는 RFID 산업에 맞춰 계속적으로 발전된 방식의 추가 코드가 제안 될 것으로 예상된다.

기본 코드를 살펴보면 GID(General Identifier)는 기존에 존재하던 어떠한 식별 체계나 표준에 관계없이 새롭게 독립적으로 제안된 식별 체계이며, SGTIN(Serialized Global Trade Identification Number) 코드는 EAN.UCC에서 제정된 GTIN(Global Trade Item Number) 코드를 기반으로 개개의 물체에 유일한 식별자를 할당하기 위하여 제안된 코드 체계이다.

SSCC(Serial Shipping Container Code) 코드 체계는 제조, 물류 또는 상품 공급 업체들이 상품을 주문한 업체로 배송하는 배송 단위(예를 들어 박스, 팔레트, 컨테이너 등)를 식별하기 위하여 제안되었고, SGLN(Serialized Global Location Number)은 현재 GLN(Global Location Number)의 물리적 위치만을 지원할 수 있다. 개개의 선반(slot)과 같은 개별적인 단위에서 창고 전체와 같은 집합적인 단위까지 포괄적으로 나타낼 수 있으며, GRAI(Global Returnable Asset Identifier) 코드 체계는 회수 자산 관리를 위해서 사용된다. 회수 자산이란, 운송 장비나 재사용 가능한 물건을 일컫는다. 즉, 팔레트, 통, 가스 실린더, 맥주 통, 철도 차량 등 운송과 무역 과정에서 사용될 수 있는 물건이나 혹은 차량을 일컫는다.

GIAI(Global Individual Asset Identifier) 코드 체계는 조직의 고정된 자산을 식별하는 데에 사용한다. 고정된 자산은 무역이나 비즈니스를 하는데 있어서 소모되지 않는 자산을 말하며, DoD 코드는 미 국방성에서 예하 부대로 입하되는 물자를 단품, 팔레트, 케이스 등으로 나누어 RFID 태그를 붙이고 이를 식별하기 위해 정의한 코드 체계이다. 상업 및 정부 사업자(CAGE : Commercial and Government Entity) 코드를 사용하는 것이 특징이며, CAGE

코드를 입력하여 물자를 공급하는 업체를 식별하고 공급 업체 내에서 사용되는 일련번호의 고유성을 보장한다.

일반적인 RFID EPC 코드 형태로는 GID-96이 있고, EAN.UCC 시스템 식별 타입으로 SGTIN-64, 96, 198, SSCC-64, 96, SGLN-64, 96, 195 등이 있다. 또한 GRAI-64, 96, 170, GIAI-64, 96, 202, DoD-64, 96이 있다. 표 1은 이러한 RFID EPC 코드 체계를 구성하는 필드 구조를 보여준다 [8,9].

표 1. EPC 코드의 필드 구조  
Table 1. The field structures of EPC codes

GID-96	Header		General Manager Number		Object Class	Serial Number	
	8	3	28	24	36		
SGTIN-64	Header		Filter Value	Company Prefix	Item Reference	Serial Number	
	2	3	14	20	25		
SSCC-64	Header		Filter		Company Prefix	Serial Number	
	8	3	14		39		
SGLN-64	Header		Filter Value	Company Prefix	Location Reference	Serial Number	
	8	3	14	20	19		
SGTIN-96	Header		Filter	Partition	Company Prefix	Item Reference	Serial Number
	8	3	3	20-40	24-4	38	
SGTIN-198	Header		Filter	Partition	Company Prefix	Item Reference	Serial Number
	8	3	3	20-40	24-4	140	
SGLN-96	Header		Filter	Partition	Company Prefix	Location Reference	Extension Component
	8	3	3	20-40	21-1	41	
GRAI-96	Header		Filter	Partition	Company Prefix	Asset Type	Serial Number
	8	3	3	20-40	24-4	38	
GIAI-96	Header		Filter	Partition	Company Prefix	Individual Asset Reference	
	8	3	3	20-40	62-42		

ALE 기반의 RFID 미들웨어는 RFID 리더 장치로부터 8000000040010000과 같은 16진 형식의 원시 코드 데이터를 입력받아 필드 구조에 따라 필드 값을 추출한 후 urn:epc:tag:sgtin-64:0.0.32.65536과 같은 형식의 URN 코드로 변환한다. URN 코드 데이터는 표준 스펙에 명시되어 있는 필터링, 그룹핑 조건에 의하여 처리되고 그 결과가 리포트로 작성되어 다양한 유비쿼터스 응용으로 전송된다.

### 2.3. RFID 미들웨어

ALE 스펙 기반의 RFID 미들웨어는 다음과 같은 요구 사항을 가진다. 우선, RFID 미들웨어는 다양한 형태의

리더 인터페이스, 다양한 코드 및 망 연동, 여러 가지 응용 플랫폼에 대해서도 상호 운용성을 보장할 수 있어야 한다. 두 번째로, 상호 운용성을 보장하는 미들웨어를 개방형 미들웨어라 하며, 이를 위하여 표준화된 코드, 정보 표현, 교환 규약을 준수하고 정보 교환을 위한 메시징 기술을 적용해야 한다. 세 번째로, 요구사항을 준수하기 위해서는 EPC 스펙, ISO 표준, 웹 서비스 등과 같은 참조 표현을 준용하고 멀티리더 프로토콜(EPC, ISO 15961, Alien)을 지원하는 인터페이스 설계 및 구현이 요구된다[10].

현재 개발된 RFID 미들웨어들의 필터링 관리 기법들에 있어서 콘텍스트 정의 및 처리는 독자 기술에 의한 정의된 필터(predefined filter)를 설계하여 적용하고 있으며, 최근에는 EPCglobal의 ALE 스펙에 기반한 미들웨어에 대한 연구 개발이 진행 중이다. 표 2는 현재 존재하는 미들웨어를 요약한 것이다[11, 12, 13].

표 2. RFID 미들웨어 비교  
Table 2. The comparison of RFID Middleware systems

분류	OAT System	Sun Java System RFID Software	Oracle Edge Server/Sensor Data Hub	CARIU
지원 리더기	Matrics, Alien, ThingMagic, SAMSys, AWID	Alien, Matrics, Sensormatic, ThingMagic	Alien, Intermec, lightstick	Alien, Intermec (PDA형), Matrics, 국내B사
데이터 형식	EPC	EPC	EPC	EPC
컨텍스트 처리	Predefined filter	Predefined filter	Predefined filter	Predefined filter
엔터프라이즈 연동	XML via File, JMS, http	File, JMS, XML/HTTP/ SOAP	Stream, JMS, Web Service, Http Post	HTTP, JMS, SOAP, XML

기존의 RFID 미들웨어의 경우 다양한 형태의 RFID EPC 태그 데이터를 지원함에 있어 EPCglobal에서 제안한 EPC Gen1, Gen2 등의 전체 코드 체계 중 일부만을 지원하는 경우가 대부분이다. 또한 지원하는 EPC 코드를 미들웨어에 종속적인 형태로 관리하고 있어서 새롭게 정의되는 RFID EPC 태그 데이터를 처리하도록 확장함에 있어서 효율적이지 않다. 또한, 기존 RFID 미들웨어의 구현상 문제점으로 컴포넌트 혹은 객체지향 기법으로 시스템을 구축했다 하더라도 프로그램 코드 내 존재하는 횡단 관심사(cross-cutting concern) 들로 인해 기능 변경 및 추가 등의 작업에 많은 시간과 노력이 필요하여 유지보수를 어렵게 한다[14, 15].

## 2.4. 온톨로지 기반 메타데이터

방대한 양의 정보에 대하여 의미를 이용하여 정보 이용의 효율성을 향상시키려는 연구로부터, XML의 의미 태그를 중심으로 한 메타 데이터 정보 모델링 등이 출현하였다. 또한 이를 개념 수준의 의미 처리로 추상화한 온톨로지(ontology) 기술이 등장하게 되었으며, 메타 데이터의 주석을 생성하고, 해석하고, 비교하기 위하여 온톨로지가 필요하다.

온톨로지는 대부분의 발전된 지식 표현 모델로서 보여질 수 있기 때문에 이질적인 환경에서 발생하는 모든 데이터 구조를 수용할 수 있다. 온톨로지는 고도로 확장이 가능하고 재사용이 가능하며, 사용자가 원하는 콘텐츠의 접근을 더욱 용이하게 한다. 이질적인 환경의 통합에서 구분적인 불일치는 SQL이나 XML과 같은 접근에 의해서 어느 정도 제거될 수 있다. 그러나 현대의 분산 환경에서 의미적인 정보의 액세스나 통합은 매우 중요하며, 이에 부응하기 위하여 온톨로지가 연구되어 왔다 [16].

## 2.5. 다중 EPC 처리를 위한 고려 사항

위와 같이 관련 연구 및 기술의 내용과 문제점들을 바탕으로 본 논문에서 고려해야 내용을 살펴보면 다음과 같다. 우선 EPCglobal의 EPC 태그 데이터 변환 규칙을 바탕으로 표준을 준수해야 한다. 두 번째로, 유연하면서 다양한 유형의 EPC 코드를 변환 처리할 수 있는 공통 방법이 필요하다. 즉, 현재까지 제안되어 있는 EPC 코드 체계를 모두 처리할 수 있어야 한다. 세 번째로, 확장할 수 있는 EPC 태그 데이터 변환 방법이어야 한다. 즉, 향후 RFID 기반의 유비쿼터스 컴퓨팅 환경은 계속적으로 발전할 것이며, 이에 따라 새로운 형태의 RFID EPC 태그 데이터가 제안될 수 있다. 따라서 새롭게 추가되는 데이터 형식을 효율적으로 처리할 수 있도록 쉽게 확장할 수 있어야 한다.

## III. 제안하는 방법

### 3.1. 개요

그림 2는 본 논문에서 제안하는 방법의 개요이다. 다수의 RFID 리더 장치로부터 SGTIN-64, SSCC-96, GIAI-202, SGTIN-198 등 다양한 유형의 RFID EPC 데이터를 입력받아 데이터 변환을 위한 온톨로지 메타 데이터를



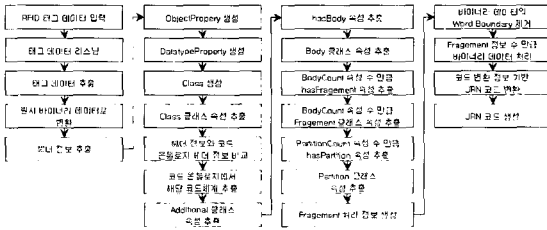


그림 5. URN 코드 변환 흐름도  
Fig 5. The flowchart for code conversion

다음으로 온톨로지에 Header 속성 정보가 있는 Class 클래스의 CodeHeader 속성 정보와 이진 데이터로 변환한 태그 데이터의 헤더 정보를 비교하여 온톨로지의 해당 코드 체계를 추출하고, 추출된 코드의 Addition 클래스 속성, Body 클래스 속성, Fragment 클래스 속성, Partition 클래스 속성을 바탕으로 변환된 이진 데이터를 URN 코드로 변환한다.

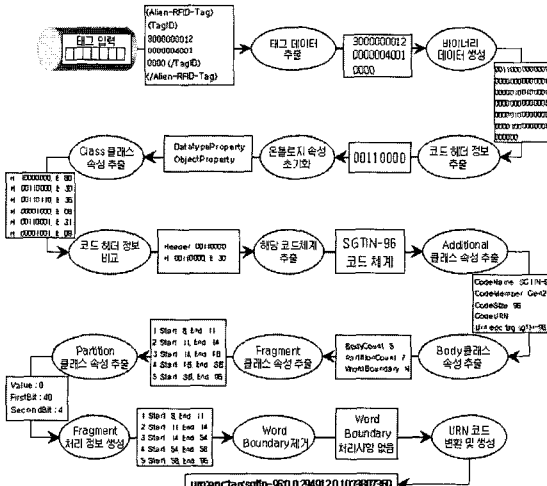


그림 6. 코드 변환의 예  
Fig 6. An example of code conversion

그림 6은 300000001200000040010000이라는 예제 데이터를 이용하여 코드 변환 규칙을 적용하여 urn:epc:tag:sgtin-96:0.0.294912.0.1073807360이라는 범용 URN 코드로 변환하는 과정을 상세하게 보여준다.

IV. 구현 및 실험

4.1. 구현 및 실험 환경

다양한 유형의 다중 EPC 코드 데이터를 RFID 미들웨어 내부에서 처리할 수 있는 URN 코드로 자동 변환하기 위한 시스템 구현 환경은 다음과 같다.

온톨로지 설계 및 구현을 위하여 OWL, Protege를 이용하였으며, Jena API를 이용하여 온톨로지 정보를 검색하고 추론하였다. 실험 데이터 집합으로는 Gen1의 코드 4종과 Gen2의 코드 8종을 이용하여 실험하였다. 코드 온톨로지 정보를 검색, 활용하여 RFID 리더로부터 수집된 RFID EPC 태그 데이터의 헤더 정보를 분석하여 이에 맞는 RFID EPC URN 코드로 자동 변환하는지에 대하여 실험하였다.

4.2. 모듈 구성

그림 7은 본 논문에서 제안하는 다중 EPC 데이터를 범용 URN 코드로 변환하기 위한 시스템 패키지 다이어그램이다.

SensorListener는 RFID 리더로부터 EPC 태그 데이터를 수집하고, TranslationApp와 CodeTransaction은 RFID EPC 데이터를 코드 온톨로지를 바탕으로 범용 URN으로 변환하기 위한 메인 클래스이다. epc 패키지는 EPC Gen1 코드를 변환하기 위한 ECTranslation 클래스와 EPC Gen2 코드를 변환하기 위한 ECTranslationExtension 클래스 등 EPC 코드를 변환하기 위한 클래스로 구성된다.

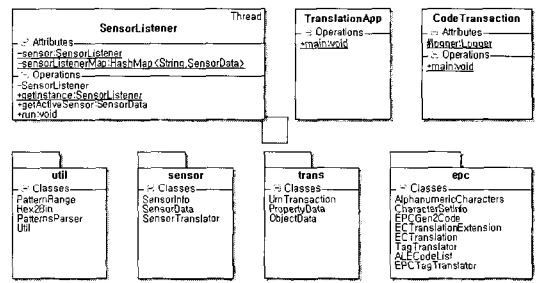


그림 7. 시스템을 위한 패키지 다이어그램  
Fig 7. Package diagram for the implementation

trans 패키지는 코드 온톨로지의 속성 자료를 처리하는 PropertyData 클래스와 코드 온톨로지의 객체 자료를 처리하는 ObjectData 클래스, 속성 자료와 객체 자료를 바탕으로 범용 URN으로 변환하는 UrnTransaction 클래스 등 코드 온톨로지 데이터를 바탕으로 범용 URN으로 변환하는 클래스로 구성된다.



그림 11은 EPC 클래스1 Gen2의 코드 중 SGTIN-96 코드를 바탕으로 실험한 내용을 보여준다. SGTIN-96 태그 데이터를 이용하여 300000001200000040010000이라는 태그 데이터가 입력되면 00110000이라는 헤더 정보를 추출하여 urn:epc:tag:sgtin-96:0.0.294912.0.1073807360이라는 URN 코드로 변경된다.

SGTIN-96 코드는 그림 10의 SGTIN-64 코드와 비슷한 형태를 갖고 있으나, 표 1에서와 같이 SGTIN-64 코드와 다른 점이 Partition 정보를 갖고 있어 Company Prefix와 Item Reference의 정보가 유동적으로 변하기 때문에 PartitionValue 정보 0을 추출하여 Company Prefix(40), Item Reference(4)를 추가로 생성한다.

그림 14는 EPC Class1 Gen2의 코드 중 SGTIN-198 코드를 바탕으로 실험한 내용을 보여준다. SGTIN-198 태그 데이터를 사용하여 360000000120000000000000400100000116912665000000000이라는 데이터가 입력되면 헤더 00110110을 추출하여 urn:epc:tag:sgtin-198:0.0.18432.0.2.0.ER3이라는 범용 URN 코드로 변환된다.

SGTIN-198 코드는 SGTIN-96 코드와 유사하게 처리되지만 워드 바운더리 처리를 추가적으로 수행한다. 워드 바운더리는 태그에 인코딩된 16진 데이터를 2진 데이터로 변경하는데 필요한 정보이다. 또한 URN 코드의 마지막 Fragment 정보인 Serial Number 정보를 SGTIN-64, SGTIN-96 코드에서 10진수로 표현하는 방식과는 다르게 7Bit 아스키 코드로 표현된다.

```

2007-11-14 16:53:14,576 INFO - ----- 온톨로지를 이용한 코드 변환 시작 -----
2007-11-14 16:53:14,576 INFO - Input CodeType = EPC
2007-11-14 16:53:14,576 INFO - Input CodeHeader = 00110110
2007-11-14 16:53:14,576 INFO - Input PartitionValue = 0
2007-11-14 16:53:14,655 INFO - Code Name = SGTIN96
2007-11-14 16:53:14,656 INFO - 코딩 방법 = 36
2007-11-14 16:53:14,656 INFO - 코드 크기 = 00110110
2007-11-14 16:53:14,656 INFO - 코드 크기 = 198
2007-11-14 16:53:14,656 INFO - 코드 URN = urn:epc:tag:sgtin-198:
2007-11-14 16:53:14,656 INFO - 코드 길이 = Gen2
2007-11-14 16:53:14,656 INFO - 바이트 갯수 = 5
2007-11-14 16:53:14,656 INFO - 비트 갯수 = 7
2007-11-14 16:53:14,656 INFO - 워드 바운더리 = 0000000000
2007-11-14 16:53:14,656 INFO - PartitionValue = 0
2007-11-14 16:53:14,697 INFO - Fragment = 40
2007-11-14 16:53:14,697 INFO - SecondBit = 4
2007-11-14 16:53:14,697 INFO - FragmentName = Filter
2007-11-14 16:53:14,697 INFO - StartLocation = 0
2007-11-14 16:53:14,697 INFO - EndLocation = 11
2007-11-14 16:53:14,697 INFO - FragmentName = Partition
2007-11-14 16:53:14,697 INFO - StartLocation = 11
2007-11-14 16:53:14,697 INFO - EndLocation = 14
2007-11-14 16:53:14,697 INFO - FragmentName = Company Prefix
2007-11-14 16:53:14,697 INFO - StartLocation = 14
2007-11-14 16:53:14,697 INFO - EndLocation = 54
2007-11-14 16:53:14,697 INFO - FragmentName = Item Reference
2007-11-14 16:53:14,697 INFO - StartLocation = 54
2007-11-14 16:53:14,697 INFO - EndLocation = 58
2007-11-14 16:53:14,697 INFO - FragmentName = Serial Number
2007-11-14 16:53:14,697 INFO - StartLocation = 58
2007-11-14 16:53:14,697 INFO - EndLocation = 198
2007-11-14 16:53:14,697 INFO - RFID TAG DATA = 360000000120000000000000400100000116912665000000000
2007-11-14 16:53:14,697 INFO - BIN - ( 20B ) 001101100000000000000000000000000000000000001001000000000000
2007-11-14 16:53:14,697 INFO - URN = urn:epc:tag:sgtin-198:0.0.18432.0.  ER 3
2007-11-14 16:53:14,703 INFO - ----- 온톨로지를 이용한 코드 변환 종료 -----
    
```

그림 12. SGTIN-198 코드 변환 실험 결과

Fig 12. The result of code conversion for SGTIN-198 code

표 3은 본 논문에서 제안한 다중 EPC 변환 방법을 적용하여 구현한 시스템을 바탕으로 12 종류의 다중 RFID EPC 코드를 URN 코드로 변환한 실험 결과로서, 총 12 종류의 코드 데이터를 URN 코드로 변환한 결과이다. 코드 당 평균 처리 시간은 0.130초이다.

표 3. 코드 변환 실험 결과  
Table 3. Experimental results of code conversion

코드명	실험데이터	헤더	변환결과	시간
SGTIN-64	8000000040010000	10	urn:epc:tag:sgtin-64:0.0.32.65536	0.125
SGTIN-96	300000001200000040010000	00110000	urn:epc:tag:sgtin-96:0.0.294912.0.1073807360	0.125
SGTIN-198	360000000120000000000000400100000116912665000000000	00110110	urn:epc:tag:sgtin-198:0.0.18432.0.2.0.ER3	0.125
SGLN-64	0600000040010000	00001001	urn:epc:tag:sgtin-64:0.0.2048.65536	0.125
SGLN-96	320000001200000040010000	00110010	urn:epc:tag:sgtin-96:0.0.294912.0.1073807360	0.125
SGLN-195	360000000120000000000000400100000116912665000000000	00111001	urn:epc:tag:sgtin-195:0.0.18432.0.2.0.2.ER2	0.141
GRAI-64	0A00000040010000	00001010	urn:epc:tag:grai-64:0.0.2048.65536	0.141
GRAI-96	330000001200000040010000	00110011	urn:epc:tag:grai-96:0.0.294912.0.1073807360	0.125
GRAI-170	3700000120000000004001000116912665000000000	00110111	urn:epc:tag:grai-170:0.0.4718592.0.ER3	0.109
GIAI-64	0600000040010000	00001011	urn:epc:tag:gai-64:0.0.1073807360	0.156
GIAI-96	340000001200000040010000	00110100	urn:epc:tag:gai-96:0.0.294912.1073807360	0.141
GIAI-202	360000000120000000000000400100000116912665000000000	00111000	urn:epc:tag:gai-202:0.0.18432.-	0.125

이와 같이 본 연구에서는 RFID EPC 태그 에뮬레이터에 의해 각 코드별로 10개의 랜덤 데이터를 생성하여 총 120개의 EPC 코드에 대하여 변환 실험을 수행한 결과 모든 코드에 대하여 성공적으로 URN 코드로 변환함을 알 수 있었다.

### V. 결 론

본 논문은 현재 활발하게 진행되고 있는 유비쿼터스 컴퓨팅의 핵심 요소 기술인 EPCglobal의 ALE 표준 스펙을 기반으로 한 RFID 미들웨어에서 다수의 RFID 리더로부터 수집된 다양한 유형의 RFID EPC 데이터를 온톨로지 기반의 메타 데이터 정의를 통하여 표준 형식의 URN 데이터로 효율적으로 변환할 수 있는 다중 EPC 변환 처리 방법을 제안하였다. 제안하는 방법을 구현한 후 12가지 유형의 데이터 120개를 이용하여 실험한 결과 데이



터를 성공적으로 변환할 수 있음을 알 수 있었다.

이처럼 다양한 형식을 갖는 EPC 태그 데이터를 효과적으로 변환하기 위하여 데이터 유형별 변환 규칙을 온톨로지를 이용하여 구축함으로써 재사용은 물론 새로운 형식을 갖는 데이터가 추가되더라도 단지 온톨로지만 추가함으로써 쉽게 확장할 수 있다. 향후 현존하는 RFID 리더에서 지원되는 EPC 데이터를 적용한 온톨로지를 바탕으로, 사용자가 직접 정의한 RFID 코드도 효과적으로 처리할 수 있는 방법에 대한 연구를 수행하고자 한다.

### 참고문헌

[1] 이근호, 한호현, 강병권, 조영빈, "유비쿼터스 컴퓨팅의 핵심 RFID Hnabook," 영진닷컴, 2002.

[2] 노영식, 변영철, "상황인식기반 지능형 홈 서비스에 관한 연구," 한국해양정보통신학회논문집, 제11권, 제4호, pp.678-686, 2007.

[3] EPCglobal Inc, "The EPCglobal network: overview of design, benefits, and security," Available: <http://www.epcglobalinc.org>.

[4] Oat Systems & MIT Auto-ID Center, "The Savant Version 0.1(Alpha)," Auto-ID Center, 2002.

[5] Sean Clark, Ken Traub, Dipan Anarkat, Ted Osinski, "Auto-ID Savant Specification 1.0," Auto-ID Center, 2003.

[6] EPCglobal, "The Application Level Event(ALE) Specification, Version 1.0," 2005.

[7] 임용훈, 주성호, 최문석, 김영일, "RFID 미들웨어 기반의 Sensor Data 처리 방안 연구," 한국정보처리학회, 2006년 추계학술대회논문집, VOL.13, NO.02, pp.1141-1144, 2006.

[8] EPCglobal, "EPC Tag Data Standards Version 1.1 Rev.1.27," 2005.

[9] EPCglobal, "EPCglobal Tag Data Translation 1.3," 2006.

[10] 최용식, 전영준, 박상현, 한수, 신승후, "RFID 미들웨어 환경에서 센서 노드의 생존성 향상과 효율적인 프로토콜 설계를 위한 연구," 한국정보과학회, 제33권 제2호, pp.68-73, 2006.10.

[11] 홍연미, 변영철, "ALE 기반 RFID 미들웨어 설계 및 구현," 한국해양정보통신학회논문집, 제11권, 제4호, pp.648-655, 2007.

[12] 이훈순, 최현화, 김병섭, 이명철, 박재홍, 이미영, 김명준, 진성일, "UbiCore : XML 기반 RFID 미들웨어 시스템," 한국정보과학회논문지, 제33권, 제6호, pp.578-589, 2006.

[13] 석수옥, 박재관, 홍봉희, "RFID 미들웨어에서 이벤트 필터링을 위한 질의 색인 기법," 제32회 통신학회추계 학술발표회논문집, Vol.32, No.2, 2005.

[14] William Griswold, Macneil Shonle, Kevin Sullivan, Yuanyuan Song, Nishit Tewari, Yuanfang Cai, Hridesh Rajan, "Modular Software Design with Crosscutting Interfaces," Journal of IEEE SOFTWARE, 2006.

[15] Ivar Jacobson, Pan-Wei NG, "Aspect-Oriented Ooftware Development with Use Cases," Addison Wesley, 2005.

[16] 홍현술, Robert M. Colomb, "시맨틱 웹을 위한 이기종 시스템간의 온톨로지 매핑," 한국정보과학회논문지, 제1권, 제1호, pp.25-32, 2004.

### 저자소개

노영식(Young-Sik Noh)



2004년 탐라대학교 컴퓨터공학과 학사

2007년 제주대학교 컴퓨터공학과 석사

2007년~현 제주대학교 컴퓨터공학과 박사과정  
 ※ 관심분야: 지식기반시스템, 시맨틱 웹, 지능형 컴퓨팅, 유비쿼터스 미들웨어

변영철(Yung-Cheol Byun)



1993년 제주대학교 정보공학과 학사

1995년 연세대학교 컴퓨터과학과 석사

2001년 연세대학교 컴퓨터공학과 박사  
 2001년 한국전자통신연구원 선임 연구원  
 2002년~현 제주대학교 컴퓨터공학과 교수  
 ※ 관심분야: 패턴인식, 시맨틱 웹, 지능형 컴퓨팅, 유비쿼터스 미들웨어