ALE 미들웨어를 위한 이벤트 데이터 처리 방법

노영식* · 변영철** · 이동철***

A method of event data stream processing for ALE Middleware

Young-Sik Noh* · Dong-Cheol Lee** · Yung-Cheol Byun***

본 연구는 교육과학기술부와 한국산업기술재단의 지역혁신인력양성사업으로 수행된 연구결과임

요 약

RFID에 관한 관심이 높아짐에 따라 RFID 리더에서 수집된 데이터를 효율적으로 처리하기 위한 RFID 미들웨어에 대한 연구가 활발히 진행되고 있다. 한편 RFID 미들웨어의 사실상의 표준을 이끌고 있는 EPCglobal에서는 미들웨어 규격으로 ALE 미들웨어 스펙을 제안하고 있으나 표준 인터페이스 및 기능만을 정의하고 있을 뿐 상세적인 스펙 구현에 대한 내용을 다루고 있지 않다. 본 논문에서는 EPCglobal의 ALE 미들웨어 스펙에서 정의하고 있는 사실상의 국제 표준인 인터페이스를 기반으로 응용 및 서비스 사용자가 직접 조건을 정의하여, 정밀한 필터링, 변경 유무 검사, 전송할 집합 산출, 데이터 유무 검사, 세밀한 데이터 그룹핑, 그리고 다양한 RFID 포맷 변환 등을 바탕으로 스트림 형태의 EPC 이벤트 데이터를 효율적으로 처리할 수 있는 ALE 미들웨어의 이벤트 데이터 처리 방법을 제안한다.

ABSTRACT

As the interests on RFID technologies increase, a lot of research activities on RFID middleware systems to handle the data acquired by RFID readers are going on actively. Meanwhile, even though various kinds of RFID middleware methodologies and related techniques have been proposed, the common data type which is dealt with in those systems is an EPC code, mainly. Also, there are few researches on the implementation of collecting the stream data queued from RFID readers endlessly and without blocking, classifying the data into some groups according to usage, and sending the resulting data to specific applications. In this paper, we propose the method of data handling in RFID middleware to efficiently process an EPC event stream data using detail filtering, checking of data modification, creation of data set to transfer, data grouping, and various kinds of RFID data format transform. Our method is based on a de facto international standard interface defined in the ALE middleware specification by EPCglobal, and application and service users can directly set various kinds of conditions to handle the stream data.

키워드

RFID Middleware, ALE, EPC, Event, Data Stream Processing

* 제주대학교 대학원 컴퓨터공학과, 첨단기술연구소
** 제주대학교 통신컴퓨터공학부 교수, 교신저자
*** 제주대학교 경영정보학과 교수

접수일자 2008. 08. 14

Ⅰ. 서론

RFID(Radio Frequency Identification)는 유비쿼터스 패러다임 실현을 위하여 부각되는 기술 중 하나로서 무선 라디오파를 이용하여 특정 물체를 인식하는 기술이다. RFID에 관한 많은 연구가 진행되면서 RFID 리더 장치에서 수집된 데이터들을 효율적으로 사용하기 위한 RFID 미들웨어에 대한 관심이 높아지고 있다[1]. 현재 다양한 RFID 미들웨어 및 기술이 제안되고 있으나 주로 EPC 코드 등과 같은 간단한 형식의 데이터를 처리하며, 대량 데이터 처리 및 RFID 리더에서 끊임없이 들어오는 스트림 형태의 EPC 이벤트 데이터를 블록킹없이 수집하여 용도에 맞게 분류하고 해당 데이터를 원하는 곳에 전달하기 위한 방법에 대한 구체적인 방법에 대한 연구는 상대적으로 미미한 실정이다[2].

사실상 RFID 기술의 국제 표준을 이끌고 있는 EPCglobal에서는 미들웨어 규격으로 초기 MIT Auto-ID 센터의 Savant 기반의 참조 모델을 제안 했으나, 2004년부터는 인터페이스 중심의 ALE (Application Level Events)를 제안하여 2005년 7월 EPCglobal의 공식 표준으로 승인되었다[3]. 아직까지 Savant 기반의 미들웨어도 존재하나 최근의 RFID 미들웨어 연구는 대부분 ALE에 기반을 두고 있다.

가장 기본적인 ALE 미들웨어 기능은 다음과 같다. Edge 영역의 여러 RFID 리더 장치와 소프트웨어 시스템으로부터 센싱되어 전달된 RFID 태그 정보를 필터링, 요약, 그룹핑하여 이를 원하는 응용으로 이벤트를 실시간으로 전달하는 것이다[4]. 즉, ALE는 성능 최적화와 EPC 데이터의 실시간 처리를 위한 요구사항들을 만족해야 하며, 이를 따르는 미들웨어 역시 ALE의 요구사항을 만족하여야 한다.

본 논문에서는 EPCglobal의 ALE 스펙에서 정의하고 있는 사실상의 국제 표준으로 지정된 인터페이스를 기반으로 응용 사용자가 직접 조건을 정의하여, 정밀한 필터링, 변경유무 검사, 전송할 집합 산출, 데이터 유무 검사, 데이터 그룹핑, 그리고 다양한 RFID 포맷 변환 등을 바탕으로 EPC 이벤트 스트림 데이터를 효율적으로 처리 할 수 있는 ALE 기반 RFID 미들웨어의 이벤트 데이터 처리 방법을 제안한다.

본 논문의 구성은 다음과 같다. 우선 Ⅱ장에서 ALE 기반 RFID 미들웨어의 요구사항을 분석하고, Ⅲ장에서는 제안하는 RFID 미들웨어의 이벤트 데이터 처리 방법에 대해 기술한다. Ⅳ장에서는 제안하는 방법을 구현하여 실험한 결과를 분석하고, 마지막 Ⅴ장에서는 본 연구의 결론에 대하여 설명한다.

Ⅱ. RFID 미들웨어 요구사항

RFID 미들웨어의 기본적인 요구사항은 다음과 같다. 첫째, 시장에 존재하는 다수의 이기종 멀티 프로토콜 기반 RFID 리더 사이에 존재하는 이질성을 상위 계층에 숨기기 위하여 공통의 인터페이스를 정의하고, 이를 통하여 태그 데이터 수집, 리더 설정, 개별 리더의 모니터링 및 원격제어 등의 관리가 이루어 질수 있도록 기능을 제공하여야 한다.

둘째, 일반적으로 RFID 태그 데이터는 리더로부터 초당 적게는 수십 회부터 많게는 수백 회의 대용량 데이터가 빠른 속도로 미들웨어로 유입되며, 동일한 태그 데이터가 중복되어 인식된다. 일반적으로 응용은 이렇게 중복되어 인식된 태그 정보 전부를 필요로 하지 않기 때문에 중복 데이터 혹은 불필요한 데이터를 선별하여 제거하는 필터링이 필요하다.

셋째, RFID 리더로부터 수집된 태그 데이터를 데이터 수요자인 WMS (창고관리 시스템), ERP (전사적 자원 관리 시스템) 등의 기존 응용 시스템으로 전달하는 기능을 지원하여야 한다. 또한 사실상 RFID 기술의 국제 표준을 이끌고 있는 EPCglobal에서는 미들웨어 인터페이스 표준으로 ALE을 제시하고 있어 이를 바탕으로 구현할 필요가 있다[5, 6].

한편, ALE 기반 미들웨어 및 기술 역시 제안되고 있다[7]. LGCNS의 RFID 미들웨어는 ALE를 기반으로 컴포넌트 기반 구조의 자바를 이용하여 높은 유연성과 확장성을 갖도록 하였으며[8], ETRI는 UbiCore라는 XML 기반의 미들웨어를 발표하였고[9], 한울은 센서 데이터 네트워크 미들웨어인 CrossOVER를 발표하였는데[10], 이는 다수의 센서를 통해 입력된 이벤트에 따라 연계된 정보 및 서비스를 자동으로 제공, 연동할 수 있다. 이와 함께 최근에는 센서 및 RFID 정보에 기반한 상황인지형 미들웨어 기술도 발표되었다[12]. 하지만 기존 연구의 경우 ALE 표준 스펙을 충실히 반영하지 않거나 EPC 데이터 스트림을 효율적으로 처리하기 위한 구체적인 방

법에 대한 제시가 미미하며, 따라서 이에 대한 연구가 필요하다.

Ⅲ. 제안하는 방법

3.1. 이벤트 스트림 데이터 처리 방법 개요

특정 RFID 응용 및 사용자는 RFID 리더기를 관리하기 쉽도록 추상화시킨 논리(logical) 리더기로부터 어떤 RFID 데이터를 선별하여 수집할 것인지, 수집한 원시 데이터를 어떠한 형태로 가공 및 정제하여 응용으로 전송할 것인지 등에 대한 조건을 명시하는 EC 스펙이라는 문서를 정의하여 ALE 기반 RFID 미들웨어로 전송한다. 미들웨어는 사용자의 요청에 따라 처리한 결과를 EC 레포트라는 문서로 만들어 응용으로 전송하고, 응용은 이를 이용하여 사용자에게 서비스를 제공할 수 있다.

ALE 미들웨어는 응용이 요청하는 데이터를 전송하려면 결과 데이터를 생성해야 하는데, 이를 위한 일련의 과정을 이벤트 사이클이라고 한다[6]. 이벤트 사이클은 ALE 인터페이스와 클라이언트 사이의 가장 작은 단위의 동작으로서 응용 관점에서 볼 경우 데이터 처리를 위한 단위 동작에 해당된다.

그림 1은 RFID 미들웨어의 단일 이벤트 사이클 내에서 이벤트 데이터 처리를 위한 흐름도이다. RFID 리더기를 통하여 획득한 원시 데이터를 클라이언트가 원하는 데이터 형태로 필터링, 그룹핑, 코드 변환 등과 같이 데이터를 처리하여 응용으로 전송한다. 참고로, RFID 태그 데이터는 이벤트 사이클 단위로 처리되며, 클라이언트 응용은 처리 조건을 EC 스펙에 정의하여 EPCglobal의 ALE 스펙에서 정의하고 있는 표준 인터페이스를 호출하여 ALE 미들웨어로 전송한다.

세부적으로, ALE 미들웨어는 이벤트 데이터 처리를 위하여 논리 리더기로부터 RFID 태그 데이터를 수집하는 원시 데이터 수집 단계, 수집한 원시 데이터 중 필터링 연산자인 EPC 패턴 형태의 필터링 패턴을 이용하여 응용이 요청하는 RFID 데이터만을 추출하는 필터링 단계, 현재 이벤트 사이클에서 필터링한 데이터를 이전 이벤트 사이클에서 필터링한 데이터와 비교하여 다른지를 검사하는 변경 유무 검사 단계가 있다.

또한 EC 스펙에 정의한 조건에 따라 이전 이벤트 사이클에서 생성한 결과 데이터와 비교하여 새롭게 추가된 데이터 집합이나 제외된 데이터 집합을 추출하거나, 혹은 현재 이벤트 사이클의 데이터 집합 전체를 추출하는 전송 집합 산출 단계, 그리고 응용에게 전달할 결과 데이터가 존재할 경우에만 EC 레포트로 만들어 응용에게 전송하는 데이터 유무 검사 단계, 현재 이벤트 사이클의 데이터를 보관하는 데이터 보관 단계, 필터링된 여러 EPC 태그 데이터들을 회사 이름이나 상품명 등에 따라 그룹화하는 데이터 그룹핑 단계, 그룹화된 데이터를 응용이 요구하는 형태, 즉, EPC, Tag, RawHex, RawDecimal 형태로 변환 및 카운팅하는 포맷 변환 단계, 그리고 최종적으로, 생성된 데이터를 실제 응용에게 전송할 EC 레포트를 생성하는 단계 및 전송하는 단계로 구성된다.

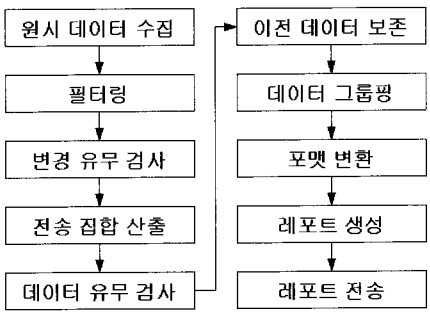

그림 1. 이벤트 데이터 스트림 처리 개요
Fig 1. The overall processing flow for event data stream

3.2. 필터링

ALE 스펙에는 사용자가 요구하는 조건에 맞는 데이터를 선별하는 기능인 필터링 연산이 기술되어 있다. 이를 위하여 그림 2와 같이 필터링 관리기, 필터링 패턴 관리기, 필터링 처리기로 필터링 기능을 설계하고 구현하였다.

필터링 관리기는 사용자가 정의한 처리 조건이 명시되어 있는 EC 스펙, 이벤트 사이클 동안 동시에 처리해야 하는 레포트 카운트 정보, 원시 데이터 수집 단계로부터 수집된 RFID 태그 원시 데이터를 입력받는다. 이후 EC 스펙에서 필터링 조건을 추출하고, 추출된 조건을 매개변수로 하여 필터링 패턴 관리기를 호출한다.

필터링 패턴 관리기는 필터링 조건을 필터링 처리기가 인지할 수 있도록 필터링 연산자에 해당하는 EPC 패턴을 분석한다. 구체적으로, RFID 태그 원시 데이터가 EPC 패턴의 조건에 부합되는 집합으로 필터링할 것인

지, 아니면 그렇지 않은 집합으로 필터링할 것인지 결정할 수 있는 조건을 추출하여 조건 패턴 리스트라는 곳에 저장한다.

필터링 처리기는 필터링 조건을 표현하는 패턴 리스트를 바탕으로 RFID 태그 원시 데이터를 필터링 하여 필터링된 데이터 집합을 생성한다. 생성된 결과 데이터는 필터링 관리기를 통하여 다음 단계에서 활용된다.

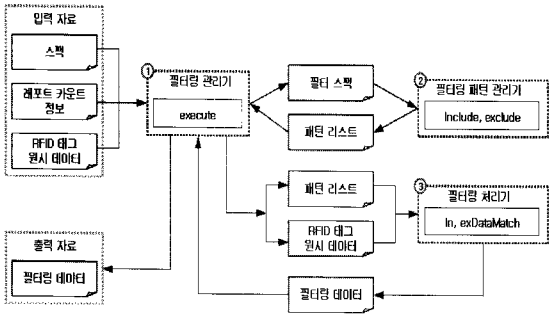

그림 2. EPC 태그 데이터 필터링
Fig 2. Filtering process of EPC tag data

3.3. 결과 데이터 변경 유무 검사

RFID 태그가 부착된 상품의 종류 및 수 등에서 변동이 발생할 경우 이를 ALE 미들웨어가 클라이언트 응용으로 적절히 알릴 필요가 있다. RFID 미들웨어에서 이를 검사하기 위한 연산이 변경 유무 검사 연산으로서, 이를 위한 모듈은 그림 3과 같이 변경 유무 관리기, 변경 유무 처리기로 구현하였다.

변경 유무 관리기는 EC 스펙, 레포트 카운트 정보, 앞 단계에서 생성된 필터링된 데이터, 이전 이벤트 사이클에서 처리되어 보관되어 있는 이전 결과 데이터를 입력받는다. 변경 유무 관리기는 입력 자료의 EC 스펙에서 변경 유무 검사 처리 조건을 추출하고, 이와 함께 필터링된 데이터, 이전 이벤트 사이클 결과 데이터를 매개변수로 변경 유무 처리기를 호출한다. 변경 유무 처리기는 변경 유무 검사 처리 조건을 바탕으로 필터링된 데이터와 이전 이벤트 사이클의 데이터를 비교하여 변경 유무에 따라 True 혹은 False를 반환한다. 변경 유무 처리기에서 생성된 결과 데이터는 변경 유무 관리기를 통하여 다음 단계에서 활용된다.

이와 같이 변경 유무 검사는, 응용에서 설정한 처리 조건에 따라, 현재 이벤트 사이클의 결과 데이터와 보관되어있는 이전 이벤트 사이클의 결과 데이터를 비교하여 비록 변경되지 않더라도, 즉, 이전 사이클과 동일하더라도 결과 데이터를 응용으로 전송하거나 혹은 그렇지 않도록 할 수 있다.

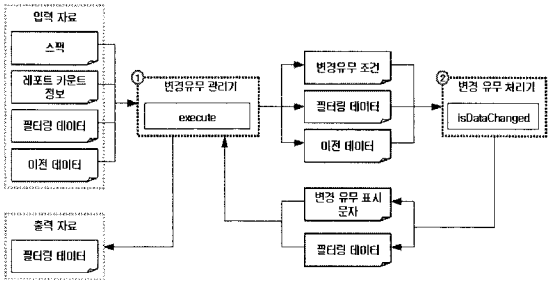

그림 3. 태그 데이터 변경 유무 검사
Fig 3. Modification detection of tag data

3.4. 전송 집합 산출

앞서 설명한 바와 같이 이전 전송 집합 산출 과정은 현 이벤트 사이클에서 생성한 결과 데이터와 이전 사이클의 결과 데이터를 비교하여 차이가 있을 경우 새롭게 추가된 데이터를 보낼 것인지, 혹은 제거된 데이터를 보낼 것인지를 결정하여 결과 데이터를 생성하는 과정이다.

전송 집합 산출 모듈은 그림 4와 같이 전송 집합 관리기, 전송 집합 처리기로 구성된다. 전송 집합 관리기는 EC 스펙, 레포트 카운트 정보, 앞 단계에서 생성된 데이터, 그리고 이전 사이클에서 보관된 결과 데이터를 입력받는다. 전송 집합 관리기는 EC 스펙에서 전송 집합의 유형에 대한 조건을 추출하고, 이를 필터링된 데이터, 이전 사이클 결과 데이터, 변경 유무 검사 결과(true, false)와 함께 전송 집합 처리기에 제공한다.

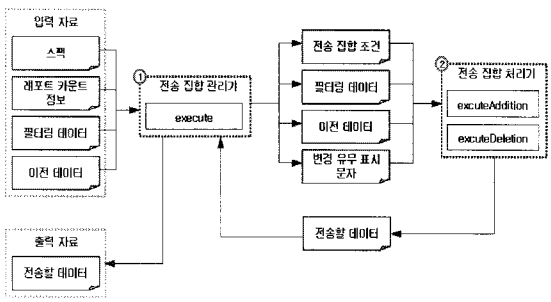

그림 4. 전송 데이터 집합 생성
Fig 4. Generation of sending data set

전송 집합 처리기는 변경 유무 결과에 따라 변경 여부를 확인하고, 처리 조건에 따라 현 단계의 결과 데이터와 이전 사이클의 결과 데이터를 비교하여 새롭게 추가된 데이터 혹은 제외된 데이터를 추출하여 응용으로 전송할 결과 데이터를 생성한다. 생성된 데이터는 전송 집합 관리기를 통하여 다음 단계에서 활용된다.

3.5. 전송 데이터 유무에 따른 처리

ALE 스펙에는 RFID 미들웨어가 클라이언트 응용으로 전송할 데이터를 가지고 있을 경우에만 결과 데이터를 전송하고, 없을 경우에는 이하의 처리를 수행하지 않고 새로운 이벤트 사이클을 수행하도록 하는 기능이 정의되어 있다. 클라이언트 응용의 경우 미들웨어가 전송할 데이터의 유무에 따른 처리가 가능하기 때문에 보다 유연한 응용 구현이 가능하다. 데이터 유무 검사는 그림 5와 같이 데이터 유무 관리기, 데이터 유무 처리기로 구성하였다.

데이터 유무 관리기는 EC 스펙, 레포트 카운트 정보, 앞 단계에서 생성된 결과 데이터를 입력받는다. 데이터 유무 관리기는 EC 스펙에 설정되어 있는 데이터 유무에 따른 처리 조건을 추출한 후 이를 결과 데이터와 함께 매개변수로 하여 데이터 유무 처리기를 호출한다.

데이터 유무 처리기는 전송 데이터 유무에 따른 처리 조건에 따라 데이터 유무를 판단하여 전송 유무를 True 혹은 False와 같이 생성하며 전송 데이터에 저장한다. 데이터 유무 처리기에서 생성된 전송 데이터는 데이터 유무 관리기를 통하여 다음 단계에서 활용된다.

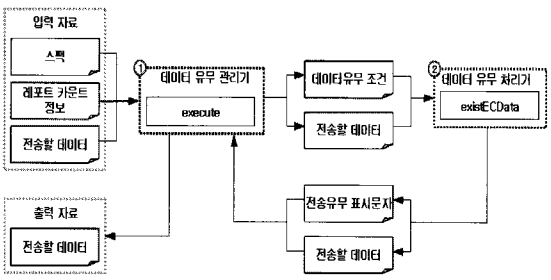

그림 5. 결과 데이터 유무 검사
Fig 5. Checking of the existence of resulting data

3.6. 이전 데이터 보관

이전 데이터 보관 과정은 현재 이벤트 사이클의 필터링, 전송 집합 산출, 데이터 유무 검사 등의 과정을 통하여 생성된 결과 데이터를 백업하여 다음 이벤트 사이클에서 활용하기 위한 것이다. 표 1은 백업을 위한 데이터베이스 테이블을 보여준다. 저장된 이전 데이터는 다음 이벤트 사이클에서 변경 유무 검사와 전송 집합 산출 과정에서 참조된다.

표 1. 이전 데이터 저장 테이블 스키마
Table 1. The table schema for backup of previous data

KIND	SPEC_NAME	URL	ECDATA
C/A/D	test_spec	#203.253.…,…	urn:epc:tag:……

3.7. 그룹핑

코드를 응용의 요청이 있을 경우 앞서 추출한 결과 EPC 데이터를 서로 그룹화할 수 있다. 가령 전자 제품, 식료품, 제조 회사 등과 같이 공통된 특성을 갖는 제품을 한데 묶을 수 있다. 그림 6은 이를 위한 그룹핑을 수행하기 위한 처리 흐름도로서 그룹핑 관리기, 그룹핑 패턴 관리기, 그룹 이름 관리기, 그룹핑 처리기 등으로 구현하였다.

그룹핑 관리기는 EC 스펙, 레포트 카운트 정보, 앞서 생성한 결과 데이터를 입력받는다. 그룹핑 관리기는 EC 스펙에서 그룹핑 조건을 추출하며, 추출된 그룹핑 조건을 그룹핑 패턴 관리기에 제공한다.

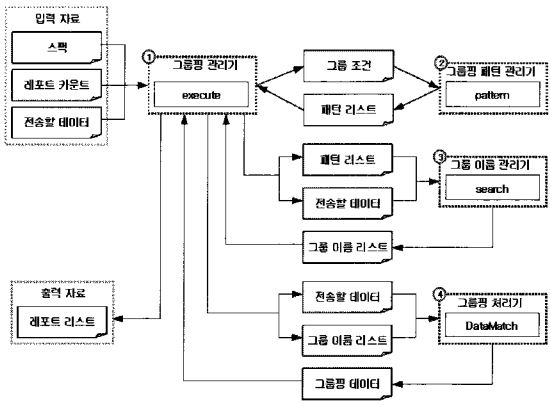

그림 6. 테그 데이터 그룹핑
Fig 6. Tag data grouping

그룹핑 패턴 관리기는 그룹 연산자인 그룹핑 패턴을 분석하여 그룹 조건을 추출하여 그룹 조건 패턴 리스트로 저장한다. 그룹 이름 관리기와 그룹핑 처리기는 이를 참조함으로써 그룹핑 조건을 알 수 있다.

그룹 이름 관리기는 그룹 조건 패턴 리스트와 전송할 데이터를 바탕으로 그룹 이름 리스트를 생성하며, 그룹핑 처리기는 생성된 그룹 이름 리스트와 전송할 데이터를 이용하여 클라이언트 응용이 원하는 형태로 그룹화를 수행한다. 또한 그룹핑 관리기는 그룹핑된 데이터를 바탕으로 응용으로 전송할 레포트 리스트를 생성한다.

3.8. 포맷 변환

ALE 미들웨어는 응용이 EC 스펙에 명시하여 요청하는 형태로 EPC 태그 데이터 포맷을 변환하여 전송해야 한다. 즉, EC 스펙에 정의하는 EPC, Tag, RawHex, RawDecimal 형태의 SGTIN, GIAI, SGLN, GRAI, SSCC, GID 코드체계로 RFID 태그 데이터를 변환하여 최종 결과 데이터를 생성한다. 그림 7의 포맷 변환은 포맷 변환 관리기, 그룹 카운팅 처리기, 그룹 리스트 관리기, 포맷 변환 처리기로 구성된다.

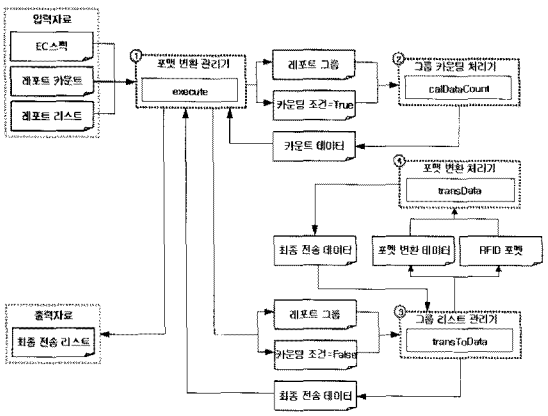

그림 7. 포맷 변환
Fig 7. Format conversion

포맷 변환 관리기 모듈은 EC 스펙, 레포트 카운트 정보, 앞서 생성한 레포트 리스트를 입력받는다. EC 스펙에 명시되어 있는 포맷 변환 조건 및 카운팅 조건에 따라 카운팅 처리를 하거나 포맷 변환을 수행한다. 카운팅 조건이 True이면 그룹 카운팅 처리기가 실행되어 입력 자료 중 레포트 리스트의 그룹을 바탕으로 카운트 데이터를 생성한다.

카운팅 조건이 False이면, 그룹 리스트 관리기가 실행되어 입력 자료 중 레포트 리스트의 그룹을 바탕으로 포맷 변환 조건과 포맷 변환할 데이터를 포맷 변환 처리기에 제공한다. 포맷 변환 관리기는 최종결과 데이터인 최종 전송 리스트를 생성하여 응용으로 전송할 수 있도록 한다.

3.9. 레포트 생성 및 전송

레포트 생성은 포맷 변환에서 생성된 최종 자료인 전송 리스트를 바탕으로 응용이 수신하는 EC 레포트를 생성하며, 이는 XML을 기반으로 처리된다. 레포트 전송은 레포트 생성 과정에서 만들어 진 EC 레포트를 Socket, RMI, HTTP, SOAP 통신 인터페이스를 사용하여 클라이언트 응용으로 전송한다.

Ⅳ. 구현 및 실험

4.1. 구현 환경

ALE 스펙에 명시되어 있는 연산 및 기본 기능을 적절히 수행하는지를 확인하기 위하여 이클립스 환경에서 자바 언어, MySQL 4.0 데이터베이스를 이용하여 구현하였다. 또한 그림 1에서 설명한 이벤트 처리의 전 과정을 구현하여 테스트하였다.

4.2. 필터링(FilterMng)

그림 8은 필터링 시퀀스 다이어그램으로 그림 2의 처리 흐름도를 바탕으로 구현하였다.

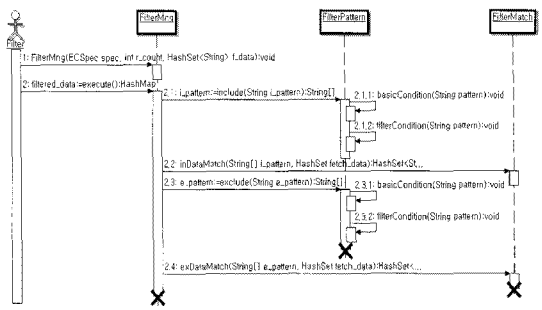

그림 8. 필터링 시퀀스 다이어그램
Fig 8. The sequence diagram for filtering

필터링 관리기에 해당하는 FilterMng 클래스는 필터링 패턴 관리기에 해당하는 FilterPattern 클래스의 include와 exclude 메소드를 호출하여 패턴을 분석하며, FilterPattern 클래스는 이를 basicCondition과 filter Condition 메소드에 의해 처리한다. 이후 FilterMng 클래스는 필터링 처리기에 해당하는 FilterMatch 클래스의 inDataMatch와 exDataMatch 메소드를 호출하여 필터링한다.

그림 9는 그림 8의 필터링을 구현하여 실험한 결과로 RFID 태그 원시 데이터(①)인 Fetched Data를 입력으로 받아 필터링 조건(②)을 바탕으로 필터링 처리 과정(③)을 통하여 Filtered Data를 생성한 결과를 보여준다(④).

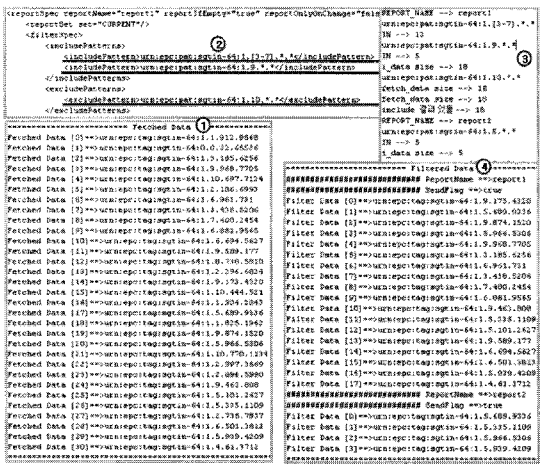

그림 9. 필터링 결과
Fig 9. The result of filtering

4.3. 변경 유무 검사(DataOnChangeMng)

그림 10은 변경 유무 검사 시퀀스 다이어그램으로 그림 3의 처리 흐름도를 바탕으로 구현하였다.

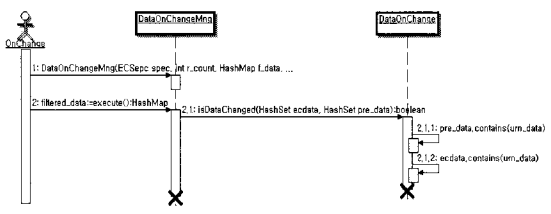

그림 10. 변경 유무 검사 시퀀스 다이어그램
Fig 10. The sequence diagram for modification dection

변경 유무 관리기에 해당하는 DataOnChangeMng 클래스는 변경 유무 처리기에 해당하는 DataOn- Change 클래스의 isDataChanged 메소드를 호출하여 변경 유무를 검사한다.

그림 11은 필터링 데이터인 Filtered Data(①)와 이전 데이터인 Pre Data(②)를 입력으로 받아 변경 유무 검사 조건(③)을 바탕으로 변경 유무 검사 처리 과정(④)을 통하여 OnChanged Data(⑤)를 생성한다.

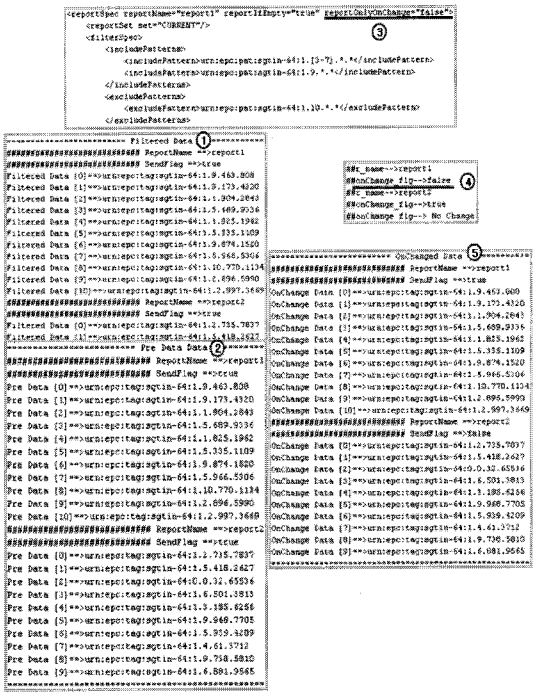

그림 11. 변경 유무 검사 결과
Fig 11. The result of modification detection

4.4. 전송 집합 산출(DataSetterMng)

그림 12는 전송 집합 산출 시퀀스 다이어그램이다. 전송 집합 관리기에 해당하는 DataSetterMng 클래스는 전송 집합 처리기인 DataSetter 클래스의 executeCurrent, executeAddition, executeDele- tion 메소드를 호출하여 현재 데이터(Current), 추가된 데이터(Addition), 제외된 데이터(Deletion) 전송 집합을 산출한다.

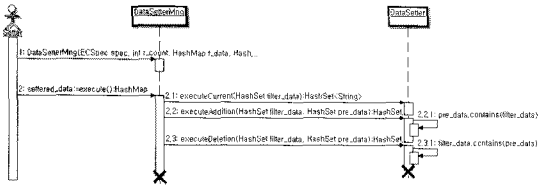

그림 12. 전송 집합 산출 시퀀스 다이어그램
Fig 12. The sequence diagram for sending data generation

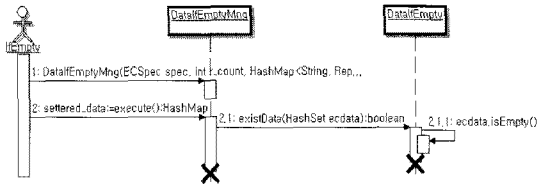

그림 14. 데이터 유무 검사 시퀀스 다이어그램
Fig 14. The diagram for existence checking of resulting data

그림 13은 필터링 데이터인 Filtered Data(①)와 이전 데이터인 Pre Data(②)를 입력으로 받아 전송 집합 산출 조건(③,⑤)을 바탕으로 Settered Data(④,⑥)를 생성한 모습이다.

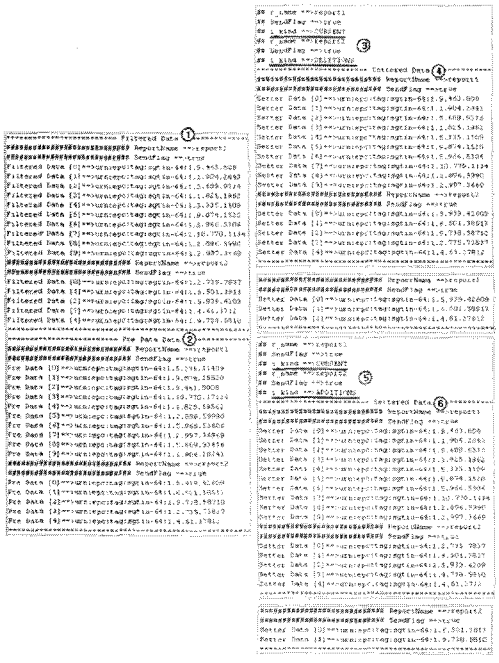

그림 13. 전송 집합 산출 실험 결과
Fig 13. The result of generation of sending data

그림 15는 전송 집합 산출 데이터인 Settered Data(①)를 입력으로 받아 데이터 유무 검사 조건(②)을 바탕으로 데이터 유무 검사 처리 과정(③)을 통하여 Settered Data(④)를 생성한 모습이다.

그림 15. 데이터 유무 검사 실험결과
Fig 15. The result of sending data checking

4.5. 데이터 유무 검사(DataIfEmptyMng)

데이터 유무를 검사하기 위하여 그림 5의 처리 흐름도를 바탕으로 구현하였다. 데이터 유무 관리기에 해당하는 DataIfEmptyMng 클래스는 데이터 유무 처리기에 해당하는 DataIfEmpty 클래스의 existData 메소드를 호출하여 데이터 유무를 검사한다.

4.6. 데이터 그룹핑(GroupMng)

그림 16은 데이터 그룹핑 시퀀스 다이어그램으로 그림 6의 처리 흐름도를 바탕으로 구현하였다. 그룹핑 관리기에 해당하는 GroupMng 클래스는 그룹핑 패턴 관리기에 해당하는 GroupPattern 클래스의 pattern 메소드를 호출

하여 패턴을 분석하며, GorupPattern 클래스는 이를 basic Condition와 groupCondition 메소드에 의해 처리한다.

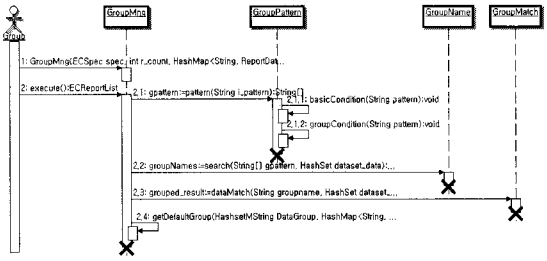

그림 16. 데이터 그룹핑 시퀀스 다이어그램
Fig 16. The sequence diagram for grouping

이후 FilterMng 클래스는 그룹 이름 관리기에 해당하는 GroupName 클래스의 search 메소드를 호출하여 그룹 이름을 산출하며, 필터링 처리기에 해당하는 FilterMatch 클래스의 dataMatch 메소드를 호출하여 데이터 그룹핑 처리를 한다. 끝으로 FilterMng 클래스는 어떠한 그룹에도 속하지 않는 데이터를 구하기 위하여 getDefault Group 메소드를 호출하여 디폴트 데이터를 그룹핑 한다.

그림 17은 그룹핑 테스트 결과로서 전송 집합 산출 데이터인 Settered Data(①)를 입력으로 받아 그룹핑 조건(②)을 바탕으로 그룹 이름을 산출(③)하여 Grouped Data(④)를 생성한 모습이다.

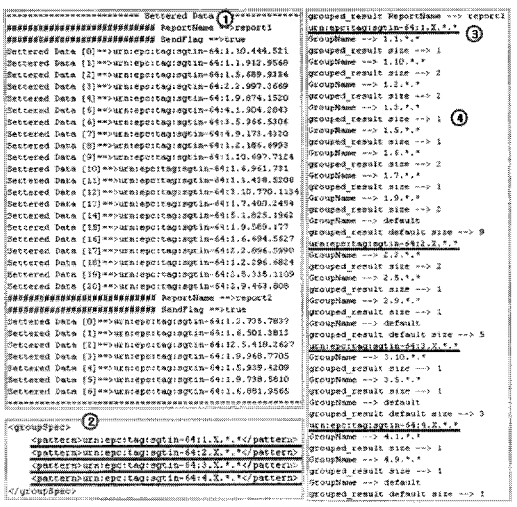

그림 17. 데이터 그룹핑 실험결과
Fig 17. The result of grouping operation

V. 결 론

본 논문은 여러 유비쿼터스 컴퓨팅의 주요 요소 기술 중 하나로 인식되고 있는 RFID 미들웨어 기술에서 실시간 테그 데이터 스트림을 처리하기 위한 방법에 관한 것이다. 사실상의 국제 표준으로 인식되고 있는 RFID 미들웨어 스펙인 EPCglobal의 ALE 스펙에서는 미들웨어의 인터페이스 및 주요 기능만을 정의하고 있을 뿐 이를 구현하기 위한 자세한 방법 혹은 알고리즘을 기술하고 있지 않을 뿐만 아니라 관련 논문 및 상세한 알고리즘에 관한 연구가 미미하다. 따라서 본 논문에서는 실시간으로 발생하는 대량의 EPC 데이터 스트림을 국제 표준을 따르는 방법으로 처리하기 위한 방법 및 알고리즘을 제안하였다. EPCglobal의 ALE 스펙에 명시되어 있는 인터페이스를 기반으로 응용 시스템 사용자가 직접 조건을 정의할 경우 이에 따라 ALE 미들웨어 측면에서 필터링, 변경유무 검사, 전송할 집합 산출, 데이터 유무 검사, 세밀한 데이터 그룹핑, 그리고 다양한 RFID 포맷 변환 등의 연산을 수행하기 위한 방법 및 알고리즘을 설계하고 구현하였다. 테스트 결과 실시간 스트림 형태의 EPC 이벤트 데이터를 처리하여 해당 클라이언트 응용에게 제공함은 물론 ALE 미들웨어 스펙에서 규정하는 기능을 효과적으로 수행함을 알 수 있었다.

참고문헌

[1] 한수, 박상현, 최용식, 전영준, 신승호, "USN/RFID 환경에서 상황인식이 가능한 통합 미들웨어 System 설계", 한국정보과학회, 2006 가을 학술발표논문집 제33권 제2호(D), pp.174-178, 2006.

[2] 박병섭, "대용량 데이터처리를 위한 XML기반의 RFID 미들웨어시스템", 한국콘텐츠학회, 한국콘텐츠학회논문지, pp.31-38, 2007.

[3] "Extending the EPC Network- The Potential of RFID in Anti-Counterfeiting," In Proceedings of ACM Symposium on Applied Computing, 2005.

[4] 황희정, 최진탁, "RFID 미들웨어의 동적 이벤트 확장을 위한 애스팩트 설계", 한국정보기술학회, 한국정보기술학회논문지 제4권 제6호, pp.31-40, 2006.

[5] 정태수, 김영일, 이용준, "RFID 미들웨어 플랫폼 기

술," Telecommunications Review 제15권 2호, 2005. 4
[6] K. Traub, S. Bent, T. Osinski, S. N. Perertz, S.Rehlinh, S. Rosenthat, and B. Tracey, The Application Level Event(ALE) Specification, ver1.0, 2005.
[7] 홍연미, 변영철, "ALE 기반 RFID 미들웨어 설계 및 구현," 한국해양정보통신학회논문집, 제11권, 제4호, pp.648-655, 2007.
[8] http://www.lgcns.com
[9] 이훈순, 최현화, 김병섭, 미명철, 박재홍, 이미영, 김명준, 진성일, "UbiCore : XML 기반 RFID 미들웨어 시스템", 한국정보과학회논문지:데이타베이스, 제33권, 제6호, pp.578-589, 2006.
[10] http://www.hanwol.co.kr
[11] http://www.repia.com
[12] T. S. Lopez and D. Y. Kim, "A Context Middleware Based on Sensor and RFID Information," Proc. of IEEE Percom' 2007, pp.331-336, 2007.

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

이동철(Dong-Cheol Lee)

2002년 성균관대학교 산업공학과 박사
2003년~현 제주대학교 경영정보학과 조교수
※관심분야 : Agent, EC, MIS 응용