

논문 2009-46CI-4-1

RFID 데이터 스트림에 대한 분산 연속질의 처리 기법

(Distributed Continuous Query Processing Scheme for RFID Data Stream)

안 성 우*, 홍 봉 희*, 정 동 규**

(Sungwoo Ahn, Bonghee Hong, and Donggyu Jung)

요 약

RFID를 적용한 기업의 활동이 점차 글로벌화 됨에 따라 RFID 네트워크에 흩어져 있는 제품의 정보를 효율적으로 수집하는 것이 필요하다. 특히, 공급망의 제품 현황을 파악하기 위해서는 해당 제품의 통계정보를 추출할 수 있는 질의를 제공해야 한다. 그러나 기존의 RFID 네트워크에서는 이러한 질의를 제공하지 않기 때문에 RFID 응용이 RFID 미들웨어에 직접 질의를 등록하고 수집된 결과를 분석해야 한다. 이러한 과정은 RFID 응용에게 높은 질의 처리 비용을 요구하는 문제가 발생된다. 이러한 문제를 해결하기 위해서 본 논문에서는 RFID 네트워크에 분산되어 있는 제품의 정보를 찾아내어 통계정보를 추출할 수 있는 분산 연속질의를 정의하고, 이를 효과적으로 처리하기 위한 분산 연속질의의 시스템을 제안한다. 제안된 분산 연속질의의 시스템은 여러 RFID 시스템 간의 제품의 이동을 실시간으로 탐지하기 위해서 Pedigree를 사용한다. 또한, Pedigree를 이용하여 동일 제품에 대한 중복 데이터가 수집되었을 때 이를 손쉽게 걸러냄으로써 질의 결과 생성에 대한 비용을 줄여주고 있다.

Abstract

An RFID application needs to collect product's information scattered over the RFID network efficiently according to the globalization of RFID applied enterprises. To be informed of the stock status of products promptly in the supply chain network, especially, it is necessary to support queries that retrieve statistical information of tagged products. Since existing RFID network does not provide these kinds of queries, however, an application should request a query to several RFID middleware systems and analyze collected data directly. This process makes an application do a heavy computation for retrieving statistical information. To solve this problem, we define a new *Distributed Continuous Query* that finds information of tagged products from the global RFID network and provides statistical information to RFID applications. We also propose a *Distributed Continuous Query System* to process the distributed continuous query efficiently. To find out the movement of products via multiple RFID systems in real time, our proposed system uses Pedigree which represents trade information of items. Our system can also reduce the cost of query processing for removing duplicated data from multiple middleware systems by using Pedigree.

Keywords: RFID, 미들웨어, 분산 연속질의, EPCglobal 네트워크, 데이터 스트림

I. 서 론

RFID(Radio Frequency Identification) 기술에 대한

사실상 표준을 담당하고 있는 EPCglobal^[1]은 RFID 정보에 대한 글로벌 네트워크 구축을 목표로 다양한 표준 명세를 제안하고 있다. 그 중 EPCglobal 아키텍처 프레임워크(EPCglobal Architecture Framework)^[2]는 RFID를 이용한 물류 환경의 개선 및 발전을 위한 핵심 서비스와 그와 관련된 하드웨어 및 소프트웨어 전반에 걸친 표준에 대해서 기술한다. 이를 통해 기업은 공급망에서 객체의 가시성, 추적성, 자동화, 보안성 등을 강화할 수 있게 되어, 재고 손실의 최소화뿐만 아니라 소비자 기호 변화에 따른 대응 능력을 향상할 수 있다^[3-5].

* 정희원, 부산대학교 컴퓨터공학과
(Department of Computer Engineering,
Pusan National University)

** 정희원, NHN(주) 뉴스서비스개발팀
(News Service Development Team, NHN Corp.)

※ 이 논문은 2009년 교육과학기술부로부터 지원받아 수행된 연구임 (지역거점연구단육성사업/차세대물류 IT기술연구사업단)

접수일자: 2009년5월29일, 수정완료일: 2009년7월3일

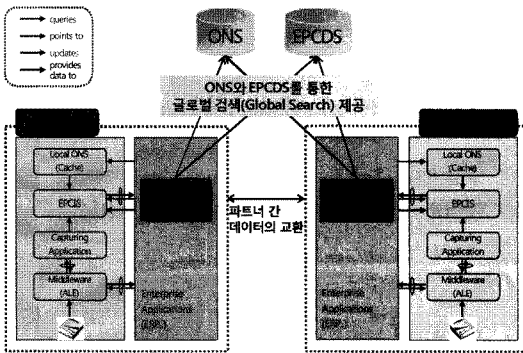


그림 1. EPCglobal 아키텍처 프레임워크
Fig. 1. EPCglobal Architecture Framework.

EPCglobal 아키텍처 프레임워크 기반의 EPCglobal 네트워크에서는 제품의 이동에 따라 발생하는 이력 정보를 공유함으로써 기업 간의 상호 협력에서도 이점을 제공한다. 그림 1과 같이 EPCglobal 아키텍처 프레임워크의 인터페이스에 따라 각 회사에서는 리더를 통해서 인식된 태그의 정보를 RFID 미들웨어인 ALE (Application Level Events)를 거쳐 EPCIS(EPC Information Services)에 저장한다. 만약 “제조사 A”와 “물류창고 B”를 통해서 이동한 “제품 C”의 이력정보를 추적하기 위해 RFID 응용은 각 EPCIS에 저장된 “제품 C”에 대한 정보를 접근해야 한다. 이런 경우, “제품 C”에 대한 글로벌 검색은 ONS(Object Name Service)와 EPCDS(EPC Discovery Service)를 통하여 “제품 C”의 정보를 저장한 EPCIS를 찾아냄으로써 가능하게 된다.

물류 서비스의 속도가 점점 증가하고 해당 기업들이 제공해야 할 서비스 범위가 글로벌화 됨에 따라, 다수의 기업에서 생산되는 정보를 통합하고 이를 기반으로 한 의사 결정의 필요성이 주요 현안으로 부상되고 있다. 이를 위해서 EPCglobal 네트워크는 전세계에 흩어져 있는 다수의 기업들이 관리하는 개별 정보를 수집하여 통계정보를 효율적으로 도출할 수 있는 질의를 제공해야 한다. 또한, 실시간으로 변화하는 기업들의 정보를 신속히 반영하기 위해서 이러한 질의는 RFID 스트림 데이터에 대한 분산 연속질의(Distributed Continuous Query)의 형태로 응용에게 제공되어야 한다^[6].

그러나 이러한 요구사항에 대처함에 있어 기존의 EPCglobal 네트워크에서 제공하는 인터페이스를 통한 단순 질의 처리는 크게 두 가지의 문제점이 발생한다. 첫째, EPCDS는 동기 전송 방식만을 지원하기 때문에 질의 결과 요청 시점의 정보만이 최신 정보가 된다. 따

라서 실시간으로 제품의 상태를 계속 추적하기 위해서는 EPCDS로 반복적으로 질의를 내려야 하는 문제가 있다. 둘째, EPCIS는 제품 정보에 대한 이벤트가 발생할 때마다 저장을 하기 때문에 제품의 수량, 비율 등의 변화에 대한 결과를 얻는 분산 연속질의의 수행 시 중복 데이터의 처리가 필요하다. 예를 들어, 특정 제품의 이전 수량과 현재 수량을 비교하여 증감률을 보고하기 위한 질의 수행 시 제품의 이동에 따라 다수의 EPCIS에 저장된 동일한 이벤트가 중복되어 응용에 전달될 수 있다. 이런 경우, 응용에서는 중복 이벤트를 제거한 후 질의를 처리해야 한다.

이와 같은 문제를 해결하기 위해, 본 논문에서는 제품의 거래에 대한 이력 정보를 제공하는 Pedigree를 이용한 분산 연속질의의 시스템을 제안한다. 먼저 RFID 태그를 부착한 제품에 대한 분산 연속질의를 정의하고, 분산 연속질의를 효율적으로 처리하기 위한 기법을 제시한다. 새로운 분산 연속질의의 처리 기법은 Pedigree의 발생 여부를 실시간으로 감시하여 제품의 이동을 인지하며, 이를 통한 동적 질의 분배 방식을 통해 EPCDS에 대한 접근을 최소로 하고 있다. 또한, Pedigree에 기록된 제품 정보를 이용하여 각 EPCIS에서 전달되는 질의 결과들 중에서 중복 데이터를 효과적으로 파악하여 제거할 수 있다. 마지막으로 본 논문에서 제시하는 기법의 우수성을 검증하기 위해서 Pedigree 기반 분산 연속질의의 시스템을 설계하고 구현하여 기존의 시스템과 분산 연속질의를 수행했을 때의 성능을 비교한다.

본 논문의 구성은 다음과 같다. II에서는 관련 연구와 RFID 정보에 대한 분산 연속질의의 처리 시의 문제점을 분석한다. III장에서는 문제 정의를 기반으로 하여 이를 해결한 분산 연속질의의 처리 기법을 기술한다. IV장에서는 분산 연속질의의 시스템의 구조를 설계한 후 이에 대한 구현 내용 및 성능 평가 결과를 보인다. 마지막으로 V장에서는 결론 및 향후 연구를 기술한다.

II. 관련 연구

1. 다중 데이터 스트림 처리

기존의 DSMS(Data Stream Management System)에서는 단일 데이터 스트림뿐만 아니라 다중 데이터 스트림에 대해서도 처리가 가능하며 사용자가 각 데이터 스트림에 질의를 수행하는 것이 가능하도록 되어 있다. 이를 위해 일반적으로 각 서브 DSMS로부터 전달되는

데이터 스트림을 중앙 DSMS에서 일괄 처리하는 중앙 집중형 방식을 채택하고 있다. 이러한 방식을 채택한 DSMS로는 CACQ(Continuously Adaptive Continuous Query)^[7], STREAM(STanford stREam datA Manager)^[8], AdaptiveCQ^[9] 등이 있다.

이들 DSMS가 다중 데이터 스트림을 처리하는 방식은 차이가 있지만 공통적으로 다음과 같은 특징을 가지고 있다. 첫째, 연산자의 공유 및 분배이다. 즉, 하나의 DSMS로 전달되는 다중 데이터 스트림을 먼저 개별적인 연산자를 이용해서 처리한 후 생성된 결과들을 다중 데이터 스트림 연산자를 이용하여 통합하여 처리하는 형태이다. 둘째, 다중 데이터 스트림 처리를 위한 특수한 연산자를 지원한다. 예를 들어, STREAM에서 사용하는 CQL^[10]이라는 연속질의 처리 언어에서는 다중 스트림 처리 연산자가 존재하며, 내부적으로는 시놉시스(Synopsis)로 불리는 구조체를 이용하여 다중 스트림을 효과적으로 처리할 수 있게 하였다.

그러나 중앙 집중형 DSMS는 다중 스트림의 처리에 있어 다음의 문제점이 발생한다. 첫째, 데이터 스트림의 처리량과 등록된 질의의 개수가 증가함에 따라 과도한 조인 연산 등으로 인한 비용 증가로 성능의 한계점을 보인다. EPCglobal 네트워크와 같이 글로벌 환경에서 많은 양의 데이터 스트림의 생산이 발생할 때 중앙 집중형 접근 방식은 성능상의 문제를 발생시킬 수 있다. 둘째, 중앙 집중형 DSMS에서 지원하는 다중 스트림 처리 연산은 결과에 맞는 모든 데이터가 포함된 구조체를 사용자에게 전달한다. 이로 인해, 결과로 전송되는 데이터가 증가할 경우 응용에서 결과를 판독하는 데에도 많은 시간이 걸린다.

다중 데이터 스트림 처리에 있어 보인 중앙 집중형 DSMS의 문제를 해결하기 위한 방법으로 최근에 분산 연속질의 시스템에 관한 연구가 진행되었다. 센서 네트워크의 발전으로 인해 이러한 시스템의 사용 빈도는 더욱 늘어났고, 다중 데이터 스트림을 대상으로 하는 분산 연속질의에 대한 연구가 가속화되었다.

분산 연속질의를 처리하는 가장 대표적인 방법으로서 제시되는 것은 P2P(Peer-to-Peer)를 이용하는 것이며 CQ-Buddy^[11]와 같은 시스템이 있다. CQ-Buddy는 다수의 DSMS로 구성되며 DSMS 간의 P2P 통신을 이용해 질의를 분배하고 수행된 질의 결과를 서버 노드에서 수집하여 사용자에게 전달하는 방식을 사용한다. 서버 노드는 기본적으로 BestNode P2P 엔진^[12]을 통해 주어

진 질의의 대상이 되는 스트림 데이터를 처리하는 DSMS를 찾는다. 이후 P2P를 이용해 질의 수행 대상이 되는 DSMS에 질의를 분배하고, 결과를 전달받아 CQL에서 주어진 조건에 따라 사용자에게 결과를 전달한다.

이러한 분산 연속질의 처리 기법을 EPCglobal 네트워크에 적용하고자 할 때 크게 두 가지의 문제점을 해결해야 한다. 첫째, EPCglobal 네트워크는 기본적으로 P2P를 이용하지 않는다. 물론, P2P를 이용하여 공급망의 각 EPCIS와 통신을 할 수 있지만 EPCDS가 존재하기 때문에 P2P의 효율성이 높지 않다. 둘째, EPCglobal 네트워크에서 모든 EPCIS의 위치는 EPCDS에 등록되어 있기 때문에 새로 추가되는 EPCIS에 대한 질의 분배는 무조건 EPCDS를 이용하여야 한다. 따라서 EPCglobal 네트워크에 P2P 환경을 사용하기 위해서는 CQ-Buddy의 BestNode P2P 엔진의 역할을 하는 새로운 시스템이 추가 되어야 한다.

3. RFID 데이터 스트림에 대한 분산 연속질의의 처리

분산 연속질의를 수행할 때 최신의 정보를 유지하기 위해서 제품의 위치를 실시간으로 파악하여야 한다. EPCglobal 네트워크에서는 이를 위해서 그림 2와 같이 분산 연속질의의 처리를 담당하기 위한 분산 연속질의 시스템을 추가할 수 있으며 이 시스템은 특정 제품의 위치를 파악하기 위해서 EPCDS를 이용해야 한다. 그러나 EPCDS는 질의 요청에 대한 결과를 전송할 때 동기 전송 방식만을 지원하기 때문에 분산 연속질의의 시스템의 요청이 있을 때에만 현재 제품의 위치를 알려준

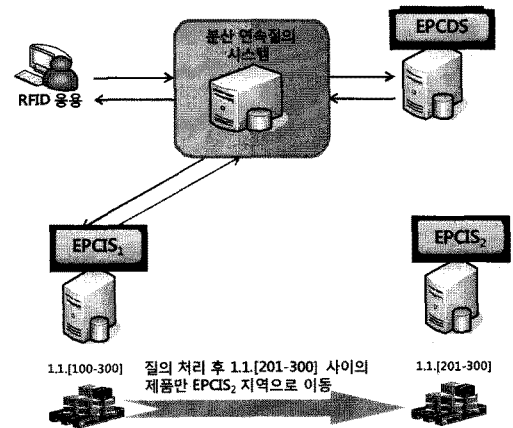


그림 2. EPCDS를 통한 분산 연속질의의 처리 시 문제점
Fig. 2. Problem of processing a distributed continuous query using EPCDS.

다. 따라서 분산 연속질의 시스템은 제품 정보 추출을 위해 EPCIS를 접근한 후에 변경된 해당 제품의 이력 정보를 즉각적으로 파악할 수 없다.

이러한 상황은 그림 2의 예를 통해서 설명될 수 있다. 만약 1.1.100~1.1.300 사이의 EPC에 해당하는 제품이 EPCIS₁에 머무르다 1.1.201~1.1.300 사이의 제품이 EPCIS₂로 이동하였다면, 제품의 이동은 EPCDS에 반영되어 기록이 된다. 그러나 RFID 응용이 1.1.250 제품에 대한 정보를 제품의 이동 이전에 요청을 하였다면, 분산 연속질의 시스템은 EPCDS를 통하여 다시 제품의 정보를 요청할 때까지 1.1.250이 EPCIS₁에 있는 것으로 판단을 한다. 이러한 경우, RFID 응용이 다시 제품의 이동 정보를 요청할 때까지 정확한 분산 연속질의의 결과를 생산할 수가 없다.

이러한 문제를 해결하기 위한 가장 간단한 방법은 RFID 응용이 요청한 제품의 정보를 분산 연속질의의 시스템에서 반복적으로 EPCDS에 정보를 요청하는 것이다. 그러나 이러한 접근법은 언제 제품이 이동할지 모르는 상황에서 지속적으로 EPCDS에 접근하여 제품의 현재 정보를 요청하는 것이기 때문에 질의 처리 비용을 증가시켜 매우 비효율적이다. 따라서 EPCDS로의 접근을 최소화하면서 제품의 이동을 효과적으로 감지할 수 있는 방법이 제시되어야 한다.

다른 문제로 동일 제품에 대한 중복 데이터가 분산 연속질의의 시스템으로 전달되는 것도 해결되어야 한다. ALE를 통해 수집되어 EPCIS에 저장되는 제품 정보는 현재 해당 제품이 대상 EPCIS가 관리하는 지역 내에 존재하지 않더라도 사라지지 않고 계속해서 저장되어 있다. 따라서 여러 EPCIS를 통해 이동하는 제품의 경우 각 EPCIS에 저장된 제품의 정보가 모두 전달되어 정확한 통계정보를 추출하는 데 어려움이 따르게 된다.

정확한 통계정보의 계산을 위해서는 EPCIS에서 전달되는 모든 질의 결과를 일일이 비교하여 중복 데이터를 제거해야만 한다. 이는 EPCIS로부터 전달되는 모든 질의 결과를 저장하고 있어야 가능한 방법이며, 비교 연산을 위하여 분산 연속질의의 시스템에 많은 부하가 발생하고 중복 데이터의 저장을 위한 공간이 증가하는 문제가 있다. 따라서 EPCglobal 네트워크에서 분산 연속질의의 시스템이 중복 데이터 제거를 통하여 RFID 응용이 요청한 질의에 대한 정확한 결과를 효과적으로 전달할 수 있는 방법이 필요하다.

III. EPCglobal 네트워크 기반 분산 연속질의 처리 기법

1. 대상 환경

그림 3은 EPCglobal 아키텍처 네트워크에서 RFID 응용의 분산 연속질의의 요청을 처리하기 위한 시스템 구성을 보여주고 있다. RFID 응용이 “현재 부산 지역의 매장 내에 있는 삼성 카메라 중 EPC가 1.1.1~1.1.1000인 제품 수량의 증감률을 보고하라”와 같은 형태의 질의를 요청할 경우 질의를 수신한 분산 연속질의의 시스템은 검색이 필요한 제품의 EPC를 저장하고 있는 EPCIS의 위치를 EPCDS에게 요청하여 EPCIS의 주소를 전달받는다. 분산 연속질의의 시스템은 전달받은 주소 리스트를 이용하여 각 EPCIS에 질의를 등록하고 질의 수행의 결과로 EPCIS로부터 전달받은 제품의 수량 정보를 분석하여 증감률을 계산한다. 마지막으로 분산 연속질의의 등록한 RFID 응용에 증감률 정보를 전달한다.

분산 연속질의의는 주로 RFID 태그를 부착한 제품에 대한 수량 및 비율 정보를 요청하거나 이러한 정보의 증감률을 파악하고자 한다. 물론 대상이 되는 RFID 환경에 따라 더욱 다양한 질의의 형태가 필요할 수 있지만 대부분의 기업에서 일반적으로 필요로 하는 질의는 수량 질의, 비율 질의, 증감률 질의의 형태이다^[13~14].

세 가지 종류의 분산 연속질의의를 RFID 응용이 개별적인 질의 형태로 분산 연속질의의 시스템을 통해서 처리하는 것보다 공통적으로 사용할 수 있는 질의 인터

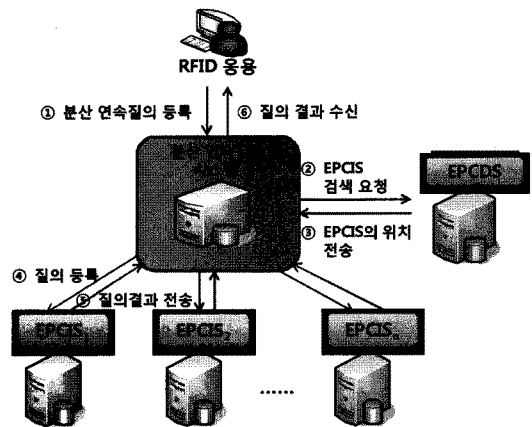


그림 3. 분산 연속질의의 처리를 위한 EPCglobal 네트워크

Fig. 3. EPCglobal network to process a distributed continuous query.

GCSpec	GCSpecName: String		
	QueryCondition	ExecuteCondition	ClassificationType
	QueryType: Integer EPC: EPC-Pattern Time: Timestamp Region: List<String> Action: Integer	Duration: Long NotificationURI: URI ReportFUchanged: Boolean	EPC: Boolean Region: Boolean

그림 4. GCSpec의 필드 정의
Fig. 4. Fields description of GCSpec.

GCReportSpec	ReportName: String		
	ResultBody: <List> QueryResult	QueryResult	ProductName: String NumOfObject: Integer Ratio: Integer Variation: String Time: Timestamp EPCISInfo
	SumOfResult: Integer		CompanyName: String Region: String WebEndPoint: URI

그림 5. GCReportSpec의 필드 정의
Fig. 5. Fields description of GCReportSpec.

페이지를 정의하는 것이 효율적이다. 그림 4와 그림 5는 분산 연속질의 시스템에서 질의 등록 및 결과 수신을 위해서 사용하기 위한 질의 인터페이스로 각각 GCSpec (Global Continuous Query Specification)과 GCReportSpec(Global Continuous Query Report Specification)을 보여주고 있다.

GCSpec은 RFID 응용이 분산 연속질의 시스템에 등록하는 질의 인터페이스이다. GCSpec은 다음과 같이 질의 조건(QueryCondition), 질의 수행 조건(ExecuteCondition), 분류 형태(ClassificationType)의 세 가지 부분으로 나누어진다.

- QueryCondition: 각 EPCIS로부터 제품의 정보를 수신하기 위한 조건을 명시한다.
 - QueryType: 수행하고자 하는 분산 연속질의 종류 (수량 질의, 비율 질의, 증감률 질의)
 - EPC: 질의 대상이 되는 제품의 EPC (EPC 범위나 패턴의 형태로 조건부여 가능)
 - Time: 질의 수행 대상이 되는 시간
 - Region: 질의 수행 지역 (대상 EPCIS들을 포함하는 지역을 명시)
 - Action: 제품의 상태 (판매완료, 재고, 구입)
- ExecuteCondition: 분산 연속질의 수행을 위해 대상 EPCIS에 제공할 질의 조건을 명시한다.
 - Duration: 질의 등록 후 각 EPCIS에서 수행될 질의의 반복 주기
 - NotificationURI: 질의 수행 후 결과를 받을 RFID 응용의 주소
 - ReportFUchanged: 직전에 전달받은 결과와 비교하여 값의 변동이 없을 때에도 결과를 전달받을

것인지에 대한 여부

- ClassificationType: 각 EPCIS에서 전달받은 정보를 이용하여 QueryType에 따라 제품에 대한 통계 정보를 추출한 뒤 결과를 정렬하기 위한 조건을 명시한다(옵션 필드).
 - EPC: 제품별로 결과를 분류할 지에 대한 여부
 - Region: 지역별(결과를 수집한 EPCIS별)로 결과를 분류할 지에 대한 여부

GCReportSpec은 RFID 응용이 등록된 GSSpec에 대한 결과를 분산 연속질의 시스템을 통해서 수신하기 위한 인터페이스이다. GCReportSpec은 질의 결과(QueryResult)의 리스트를 담고 있는 ResultBody와 결과의 개수를 담고 있는 SumOfResult로 크게 나누어진다. QueryResult는 각 EPCIS에서의 질의의 수행 결과를 포함하고 있으며 제품명(ProductName), 수량(NumOfObject), 비율(Ratio), 증감률(Variation), 시간(Time), EPCIS 정보(EPCISInfo) 필드를 포함하고 있다. 이 중에서 NumOfObject, Ratio, Variation 필드는 GCReportSpec에 대응되는 GCSpec의 QueryType에 따라서 선택적으로 값이 채워진다. EPCISInfo는 해당 QueryResult를 수행하여 전달하는 EPCIS의 위치 정보를 나타낸다. 이를 위해 대상 EPCIS를 운영하는 회사(CompanyName), 지역(Region), 접근 주소(WebEndPoint) 필드를 포함하고 있다.

2. 동적 질의 관리를 통한 분산 연속질의 처리
가. 기본 아이디어

EPCglobal 네트워크에서 분산 연속질의 시 2.3절에서 언급한 것과 같은 두 가지의 문제를 해결하기 위해 본 논문에서는 Pedigree^[15-16] 기법을 이용한다. Pedigree는 제품의 생산 및 금전적인 이력 정보를 나타내는 제품의 거래 정보를 의미하며, 주로 기업 간에 비즈니스 트랜잭션이 일어날 때 유통되는 제품에 대한 신뢰성을 보장하기 위해 사용이 된다. 이를 위해 Pedigree는 제품 정보, 전자 서명, 수신자 정보, 송신자 정보 등을 저장한다. 그림 6은 비즈니스 트랜잭션이 발생하여 제품이 유통될 때의 Pedigree 교환 예를 보여준다.

그림 6과 같이 “제조사 A”는 “제품 AAA”의 생산 후 “도매상 B”에게 제품을 보낼 때 제품의 Pedigree 정보를 함께 보낸다. “도매상 B”에서는 “제품 AAA”가 입고된 후 수신된 Pedigree에 대해서 다시 “제조사 A”로 회

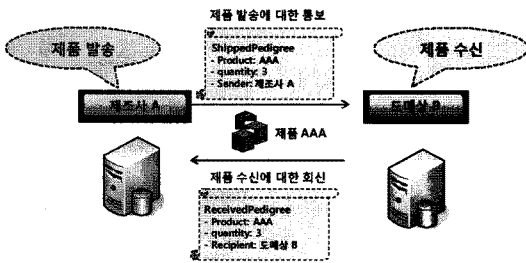


그림 6. 제품 거래 시 생성되는 Pedigree 정보
Fig. 6. Pedigree information at exchanging products.

신한다. 이렇게 제품을 주고받을 때 제품의 수신을 확인하는 Pedigree 정보를 남김으로써 해당 제품의 유통 경로를 파악하는 것이 가능하며 Pedigree를 통하여 인증된 제품만 유통되도록 할 수 있다.

본 논문에서는 Pedigree와 EPCIS의 질의 인터페이스를 이용하여 EPCIS 간의 제품 이동 시 생성되는 Pedigree의 발생 즉시 분산 연속질의 시스템으로 전달하는 것을 기본 아이디어로 사용한다. 이를 통하여 각 파트너 간 생성된 비즈니스 트랜잭션에 의한 제품 이동을 즉시 파악하는 것이 가능함으로써 분산 연속질의 처리를 위해 주기적으로 EPCDS에 접근하지 않아도 된다. 또한, Pedigree에 기록된 제품정보를 이용하여 질의를 동적으로 생성 및 분배함으로써 여러 EPCIS에서 전송되는 중복 데이터를 손쉽게 제거할 수 있다. 마지막으로, Pedigree 정보는 기존의 EPCglobal 네트워크에서 사용되는 EPCIS 이벤트의 확장점(Extension Point)^[17]을 통하여 전달할 수 있기 때문에 EPCIS의 질의 인터페이스를 그대로 이용할 수 있으며, EPCDS를 거치지 않더라도 Pedigree를 통해서 저장된 파트너 EPCIS의 웹 주소를 통하여 바로 접근할 수 있는 장점이 있다.

나. 동적 질의 관리를 통한 분산 연속질의 처리

분산 연속질의의 처리 과정은 크게 1. EPCDS에 문의하여 질의를 등록할 EPCIS의 선정, 2. EPCIS에 등록할 질의의 생성, 3. 각 EPCIS로의 질의 분배 및 질의 결과 생성의 단계를 거친다. 이러한 각 동작들은 순차적으로 이루어지며, GCSpec을 통하여 명시된 질의 실행 주기마다 매번 같은 동작을 반복 수행함으로써 실시간으로 정보를 업데이트한 후 RFID 응용에게 결과로 생성된 GCReportSpec을 전달할 수 있다. 이러한 단순 질의 관리 기법을 이용하여 분산 연속질의를 처리할 때 질의 수행 대상이 되는 EPCIS를 선정하는 과정을 매

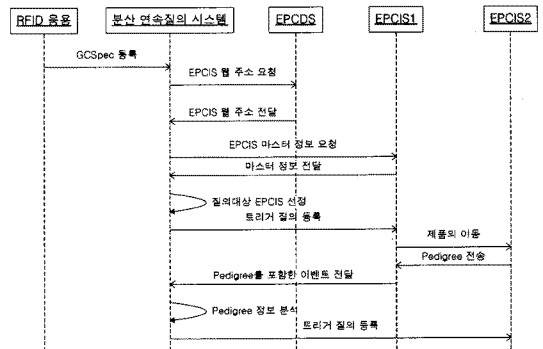


그림 7. 대상 EPCIS 선정 과정
Fig. 7. Sequence for the step of selecting target EPCISs.

질의 수행 주기마다 반복을 해야 하므로 EPCDS로의 질의 처리 비용이 증가하는 문제가 있다.

분산 연속질의의 처리를 위해 본 논문에서 제시하는 동적 질의 관리 기법 중 첫 단계인 질의 대상 선정은 그림 7과 같은 순서로 진행된다. EPCIS의 선정 단계에서 수행하는 작업에 대한 설명은 다음과 같다.

- 1) RFID 응용이 GCSpec을 생성 후 분산 연속질의의 시스템에 질의를 등록한다.
- 2) 분산 연속질의의 시스템은 GCSpec에서 정의된 EPC 패턴을 추출한 후 EPCDS에 해당 EPC 패턴을 가진 제품이 존재하는 EPCIS의 웹 주소를 요청한다.
- 3) 대상 EPC 패턴을 가진 제품을 가진 EPCIS가 존재하면 EPCDS는 EPCIS의 웹 주소를 분산 연속질의의 시스템에 전달한다.
- 4) 분산 연속질의의 시스템은 EPCIS 웹 주소를 기반으로 대상 EPCIS를 선정한다. 이를 위해, 분산 연속질의의 시스템은 각 EPCIS에 접근하여 EPCIS의 지역 정보 (Region)를 요청하기 위한 마스터 데이터 질의를 수행한다. 결과로 전달받은 EPCIS의 지역 정보와 GCSpec에서 명시한 지역 정보를 비교하여 질의 수행 대상 EPCIS를 선정한다.
- 5) 분산 연속질의의 시스템은 선정된 EPCIS에게 Pedigree 정보를 요청하기 위하여 트리거(Trigger) 질의를 등록하여 Pedigree 발생 시 분산 연속질의의 시스템에 전달되도록 한다.
- 6) GCSpec을 통해서 선택된 제품의 이동이 발생하면 제품이 도착한 EPCIS에서는 Pedigree를 발생시켜 제품이 출발한 EPCIS에 전송하며, Pedigree 정보는

등록된 트리거 질의에 의해서 분산 연속질의 시스템에 전달된다.

- 7) 분산 연속질의 시스템은 Pedigree 정보를 통하여 제품이 도착한 EPCIS에 접근하여 해당 EPCIS의 지역 정보와 GCSpec에서 명시한 지역 정보를 비교한다. 만약, GCSpec의 지역 정보에 포함된다면 해당 EPCIS에 대해서 5), 6)의 과정을 반복한다.

EPCIS는 RFID 응용이 질의를 등록하여 결과를 수신하기 위한 표준 질의 인터페이스를 제공하고 있다^[17]. 따라서 분산 연속질의 시스템이 대상 EPCIS를 선정 한 후 각 EPCIS에서 결과를 수신하기 위해서는 EPCIS 표준 질의 인터페이스에 따라서 질의를 생성하는 것이 필요하다. 그림 8과 그림 9는 RFID 응용이 등록된 GCSpec과 제품의 이동으로 발생한 Pedigree 정보를 기반으로 EPCIS 질의를 생성하는 예제를 보여주고 있다.

먼저, 그림 8과 같이 분산 연속질의 시스템은 등록된 GCSpec의 각 필드의 값을 EPCIS 질의 인터페이스에 대응되는 필드에 입력한다. 그러나, EPCIS 질의에서 Match_EPC와 eventTIME에 대한 정보를 GCSpec에서 모두 얻어내지는 못하며, 이는 제품의 이동에 의해서 발생한 Pedigree를 수신한 후 결정할 수 있다. 그림 9는 이러한 과정의 예를 보여주는 것으로 EPCIS 선정 과정

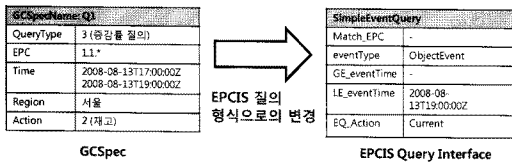


그림 8. GCSpec을 통한 EPCIS 질의 생성
Fig. 8. Generation of EPCIS query by GCSpec.

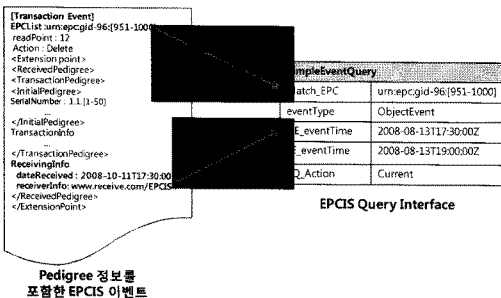


그림 9. Pedigree를 이용한 EPCIS 질의 조건 변경
Fig. 9. Changing query conditions of EPCIS query by Pedigree.

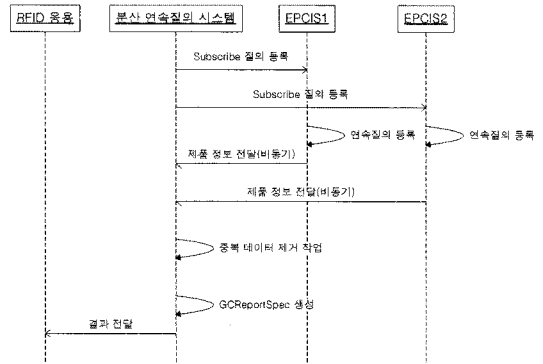


그림 10. 질의 분배 및 질의 결과 생성 과정
Fig. 10. Sequence for the step of query distribution and result generation.

에서 등록된 트리거 질의를 통하여 Pedigree가 포함된 EPCIS 이벤트를 수신한 후 그림 8의 단계에서 결정하지 못했던 두 필드의 값을 결정할 수 있다.

동적 질의 관리를 통한 분산 연속질의 처리의 마지막 단계로 생성된 EPCIS 질의를 각 EPCIS에 등록을 한 후, 전송된 결과를 모아서 RFID 응용에 전달할 GCRReportSpec을 만들어야 한다. 그림 10은 이에 대한 작업 순서를 보여주고 있으며, 질의 분배 및 질의 결과 생성 과정에 대한 설명은 다음과 같다.

- 1) 질의 생성 과정에서 생성한 EPCIS 질의를 각 EPCIS에 전달하여 EPCIS의 Subscribe 연속질의 인터페이스를 통해 등록한다^[17].
- 2) 만약 질의 대상 선정 시 EPCIS에 등록된 트리거 질의를 통하여 Pedigree가 전송된 경우 이전에 등록된 질의는 삭제하고, 질의 생성 단계를 통하여 새로 생성된 질의를 다시 EPCIS의 Subscribe 연속질의 인터페이스를 통해서 등록한다.
- 3) 각 EPCIS는 등록된 질의에 따라서 해당 제품이 존재하는지 검색한 후 제품이 존재하면 제품에 대한 정보를 분산 연속질의 시스템에 전달한다. 만약, 제품이 존재하지 않으면 연속질의를 통하여 제품이 해당 EPCIS에 도착하는지를 탐지하여 도착 시 바로 분산 연속질의의 시스템에 전달한다.
- 4) 각 EPCIS로부터 전달되는 결과에 명시된 제품의 EPC와 트리거 질의를 통해 전달된 Pedigree에 명시된 EPC 패턴을 비교한다. 만약, 제품의 EPC가 Pedigree의 EPC 패턴에 포함된다면 해당 제품에 대해서 중복된 결과를 수신한 것이라 판단한다. 따라서 이는 결과로 포함하지 않고 제거한다.

- 5) 중복 데이터 제거 과정을 거쳐 생성된 결과에서 각 EPC의 수량 정보를 포함하고, 이를 이용하여 GCSpec에 대한 결과로 GCReportSpec을 만들어 다시 RFID 응용에게 전달한다.

IV. 분산 연속질의 시스템의 설계 및 구현

1. 분산 연속질의 시스템 블록 다이어그램

분산 연속질의 시스템의 구조를 블록 다이어그램으로 나타내면 그림 11과 같다.

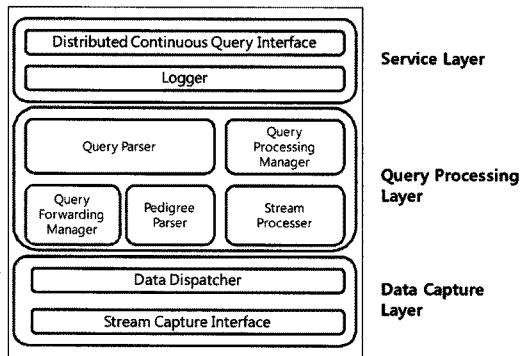


그림 11. 분산 연속질의 시스템 블록 다이어그램
Fig. 11. Block diagram of the distributed continuous query system.

분산 연속질의 시스템은 서비스 계층(Service Layer), 질의 처리 계층(Query Processing Layer), 데이터 획득 계층(Data Capture Layer)으로 구성되어 있다. 각 계층에 대한 상세 설명은 다음과 같다.

- 서비스 계층 (Service Layer): RFID 응용이 접근하여 질의를 요청할 수 있도록 질의 인터페이스를 제공해주는 역할을 한다. 서비스 계층의 API는 기본적으로 SOAP-RPC를 외부에 제공하기 위해 구현된 인터페이스로 구성되어 있다. Distributed Continuous Query Interface는 RFID 응용이 분산 연속질의 시스템에 질의를 등록하고 질의에 대한 결과를 받기 위한 외부 인터페이스를 제공해준다. Logger는 각 RFID 응용이 질의한 결과를 HTTP와 TCP 프로토콜을 이용해 전달하며, 비동기 전송 방식으로 결과의 제공이 가능하도록 해준다.
- 질의 처리 계층 (Query Processing Layer): 분산 연속질의를 수행하고 이 과정에서 연속질의 생성, 질의 분배 등을 처리하는 역할을 한다. Query

Parser는 동적 질의 관리를 통하여 각 EPCIS에 등록하기 위한 질의 또는 Pedigree 정보 요청을 위한 질의를 생성하며, 생성된 질의를 Query Processing Manager의 지시에 따라 Query Forwarding Manager로 전달하는 역할을 한다. Query Processing Manager는 RFID 응용이 등록된 GCSpec을 관리하고 등록된 GCSpec의 조건에 따라 분산 연속질의를 처리하고 제어하는 역할을 한다. Query Forwarding Manager는 응용이 요청한 질의의 전처리 및 질의 분배 대상 선정의 역할을 한다. Stream Processor는 분산 연속질의의 시스템으로 전달되는 질의 수행 대상들의 질의 결과를 서비스 계층에 전달하는 역할을 한다. Pedigree 정보를 이용해 중복 데이터를 제거하고, 질의의 결과를 GCReportSpec의 형태로 만들어 서비스 계층의 Logger로 전달한다. 마지막으로, Pedigree Parser는 데이터 획득 계층으로부터 전달받은 이벤트에서 Pedigree를 추출하여 정보를 분석하는 역할을 한다.

- 데이터 획득 계층 (Data Capture Layer): 데이터 획득 계층은 각 EPCIS로부터 전달되는 질의 결과와 Pedigree가 포함된 이벤트들을 획득하여 질의 처리 계층으로 넘겨주는 역할을 한다. Stream Capture Interface는 각 EPCIS로부터 전달되는 질의 결과와 Pedigree가 포함된 이벤트 정보를 획득하는 역할을 한다. Data Dispatcher는 Stream Capture Interface로부터 전달된 데이터를 질의 처리 계층으로 전달하는 역할을 한다.

2. 분산 연속질의 시스템의 구현

EPCglobal 아키텍처 프레임워크는 RFID 정보에 대한 글로벌 네트워크 구축을 목표로 하고 있으며 이를 실현하기 위해서는 네트워크를 구성하는 각 시스템은 다양한 환경에 적용이 가능해야 한다. 이를 위해서, 본 논문에서 제시하는 분산 연속질의 시스템은 Java 2 Platform Standard Edition 6.0을 사용하여 구현되었다. 또한, 서비스 계층에서의 SOAP-RPC를 지원하기 위한 웹 컨테이너로써 본 논문에서는 Jetty Web 6.0^[18]을 사용하였으며, 이를 동작시키는 웹 서버 역시 Jetty Web Server Platform 6.0을 사용하였다. 실험은 Windows XP Professional Service Pack 2 운영체제 기반의 2GB 메인 메모리, CPU Intel Core 2 Duo E6420을 장착한 PC를 이용하여 수행하였다.

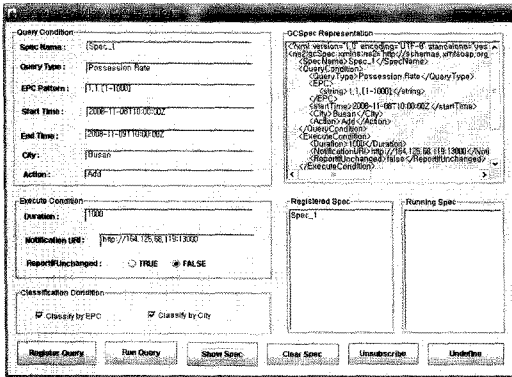


그림 12. 분산 연속질의 클라이언트 화면 구성
 Fig. 12. GUI of the distributed continuous query client.

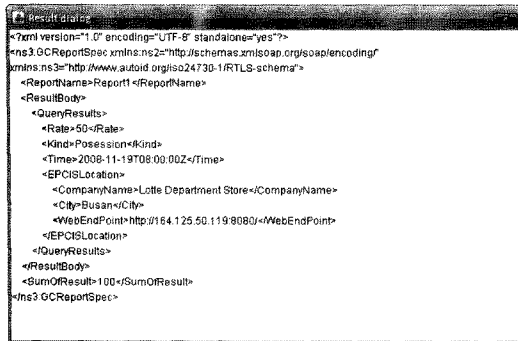


그림 13. GCSpec을 통한 질의 결과 출력
 Fig. 13. Output of the query result by GCSpec.

네트워크에서의 사용자는 분산 연속질의 시스템을 활용하는 다양한 RFID 응용을 통해서 질의 처리를 요구한다. 또한, 사용자가 직접 분산 연속질의 시스템에 접근하여 질의를 수행하고 그에 따른 결과를 전달 받을 수도 있다. 따라서 본 논문에서는 분산 연속질의 시스템의 접근과 수행된 질의 결과를 한 눈에 보는 것이 용이하도록 하기 위해 그림 12와 같이 분산 연속질의 클라이언트를 구현하였다.

분산 연속질의 클라이언트는 크게 GCSpec 등록과 GCSpec 출력 부분으로 나눌 수 있다. 그림 12에서 보듯이 사용자는 텍스트 창, 체크 박스, 라디오 버튼을 통해 질의 조건, 질의 수행 조건, 분류 조건을 입력할 수 있으며, 이를 통해 GCSpec을 생성하여 분산 연속질의 시스템에 등록하고 질의를 수행할 수 있다. 생성된 GCSpec은 XML 형태로 변환되어 분산 연속질의 시스템에 등록된다. GCSpec에 대한 질의 결과를 분산 연속질의 시스템으로부터 수신하면 그림 13과 같이 XML로 구성된 GCSpec을 출력한다.

3. 분산 연속질의 처리 성능 평가

이 절에서는 본 논문에서 제안하는 동적 질의 관리를 기반으로 한 분산 연속질의 처리 기법에 대한 성능을 평가한다. 이를 위하여 RFID 응용이 등록된 분산 연속질의를 처리하기 위해서 EPCDS에 주기적으로 대상 제품이 있는 EPCIS의 주소를 요청하는 단순 질의 관리 기법과 질의 분배 성능 및 질의 처리 성능을 비교한다. 두 기법의 성능 비교를 위해 일반적으로 공급망 내에서 일어날 수 있는 상황에 따라 제품이 이동하도록 설정한 후 이를 시뮬레이터를 통해서 실행하고 처리 시간을 측정하였다. 실험 환경에 대한 설정 내용은 표 1과 같다.

표 1에 표기한 실험 설정 값에 따라 제품은 그림 14와 같이 이동하도록 설정하였다. 각 제품은 10개의 단위로 이동을 하며, 3분 주기로 제조사에서 도매상, 도매상에서 소매상으로 이동을 한다. 따라서 제품의 이동이 완료되면 “제조사”에는 70개, “도매상 A”, “도매상 B”, “소매상”에는 각각 10개의 제품이 존재하게 된다.

단순 질의 관리 기법과 동적 질의 관리 기법은 두 가지 측면에서 차이가 난다. 첫 번째는 질의 분배 방식에서 차이가 나며, 두 번째는 질의 결과 처리 방식에서 차이가 난다. 따라서 본 절에서는 분산 연속질의 처리 성능을 질의 분배와 질의 처리로 구분하여 각 항목에 대해서 걸리는 소요 시간을 측정하여 비교하였다. 먼저, 각 질의 관리 기법에 대한 질의 분배 시의 총 처리 시간은 표 2와 같다.

표 2에서 볼 수 있듯이 총 질의 수행 시간을 비교했을 때 동적 질의 관리 기법은 약 3.3배 정도의 성능 향

표 1. 성능 평가를 위한 실험 변수 설정
 Table 1. Parameter configuration for experiments.

실험요소	내용
사용 질의	창원 지역 내에서 EPC 가 1.1.[1-100]인 제품의 각 매장별 재고 증감률을 매 분마다 보고하라.
질의 수행 횟수	12회
최대 대상 EPCIS	4개
지역 내의 제품 수	100개
제품 이동 주기	매 3분 단위로 이동

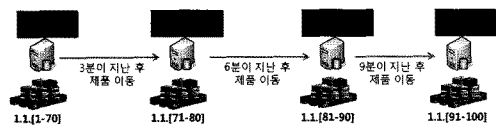


그림 14. 제품의 이동에 대한 설정
 Fig. 14. Configuration of product movements.

표 2. 질의 분배 성능에 대한 비교
Table 2. Performance comparison for query distribution.

측정항목	단순질의	
	관리기법 (ms)	동적질의 관리기법 (ms)
EPCDS 접근	6,756	563
질의 등록 확인	360	0
위치 정보 획득	628	628
질의 등록	412	412
Pedigree 정보 획득	-	882
총 질의 수행 시간	8,156	2,485



그림 15. 질의 처리 성능에 대한 비교
Fig. 15. Performance comparison for query processing.

상을 가져옴을 알 수 있다. 표의 항목을 살펴봤을 때 단순 질의 관리 기법에서 가장 많은 수행시간을 차지하는 것은 역시 EPCDS 접근으로 총 수행 시간의 대부분을 차지한다. 동적 질의 관리 기법에서는 EPCDS 접근을 질의 당 1회로 최소화하였기 때문에 EPCDS 접근에 대한 부하가 해소될 수 있었다. 비록 Pedigree의 정보를 분석하고 획득하기 위해서 추가적인 비용이 발생하지만 이에 대한 비용을 감안하더라도 단순 질의 관리 기법보다 동적 질의 관리 기법이 더 효율적임을 알 수 있다.

질의 처리 성능도 그림 15와 같이 질의 분배 성능과 마찬가지로 표 1의 시나리오에서 사용된 질의 처리에 소요되는 총 수행 시간을 측정하여 결과로 나타내었다. 단, 각 질의 수행에 대해서 생성되는 결과에 차이가 있으므로 이를 일반화하기 위하여 각 질의 수행 시마다 총 질의 결과 생성 시간을 측정하여 이를 비교하였다.

그림 15에서 볼 수 있듯이 대상 질의의 개수가 최종적으로 12개가 되는 시점에서의 질의 처리 시간을 비교했을 때 동적 관리 기법이 기존의 단순 질의 관리 기법에 비해 3배 정도 적게 드는 것으로 나타났다. 또한, 단순 질의 관리 기법에서는 질의의 개수가 하나 추가될 때마다 결과 생산을 위한 시간이 크게 증가하는데 비하여 동적 질의 관리 기법에서는 질의의 수가 증가하더라도 질의 처리 시간의 차이가 거의 없는 것을 알 수 있

다. 즉, 단순 질의 관리 기법에서는 중복 데이터 제거를 위해 모든 EPCIS에서 보내주는 질의 결과를 일일이 비교하기 위한 횟수의 증가로 인하여 처리할 질의 개수의 증가에 비례하여 질의 처리 시간이 크게 늘어난다. 그러나 동적 질의 관리 기법에서는 Pedigree를 이용하여 패턴 필터링을 하기 때문에 질의가 증가하더라도 비교하는 횟수가 거의 차이 나지 않는다. 따라서 질의 처리 시간은 거의 일정하게 나타나게 된다.

V. 결론

본 논문에서는 EPCglobal 네트워크를 구성하고 있는 EPCDS와 EPCIS를 이용하여 분산 연속질의를 수행할 때 발생하는 문제점을 분석하고 그에 대한 해결책을 제시하고 있다. 최근 RFID 시스템을 적용한 기업들의 서비스 범위가 글로벌화 됨에 따라 다수의 EPCIS를 대상으로 제품의 이력을 추적하여 이에 대한 통계정보를 얻기 위한 분산 연속질의의 필요성이 증가하고 있다. 그러나 기존의 EPCglobal 네트워크에서 이러한 분산 연속질의를 처리할 때 두 가지의 문제가 발생한다. 첫째, EPCDS는 동기 전송방식만을 지원하기에 분산 연속질의의 처리를 위해 주기적으로 접근을 해야 한다. 둘째, 분산 연속질의의 처리를 위해서 EPCIS에 등록된 질의의 결과로 다수의 EPCIS에 걸쳐서 동일한 이벤트가 중복되어 수집되는 문제가 있다.

본 논문에서는 이를 해결하기 위한 새로운 분산 연속질의 처리 기법을 제시하고 EPCglobal 네트워크 기반의 분산 연속질의의 시스템을 설계하고 구현하였다. 새로운 분산 연속질의의 처리 기법은 기업 간 비즈니스 트랜잭션이 일어날 때 주고받는 문서인 Pedigree를 이용하여 문제를 해결하였다. 즉, 물류의 흐름을 감지하고 기업 간 비즈니스 트랜잭션의 발생 및 종료 시점을 정확히 파악함으로써 EPCDS로의 지속적인 접근을 줄이고 있다. 또한, Pedigree에 기록된 제품정보를 이용하여 질의를 동적으로 생성하여 분배하고, 중복 데이터를 제거함으로써 분산 연속질의의 처리비용을 감소시켰다.

본 논문에서 제시한 질의 처리 기법을 적용한 분산 연속질의의 시스템은 EPCglobal 아키텍처 프레임워크의 기본 요소만을 이용하여 분산 연속질의를 처리할 때보다 빠른 질의 처리 성능의 보장이 가능하다. 이는 두 방법에 대해서 질의 분배 단계와 질의 처리 단계에서의 성능 비교 평가를 통해서 확인할 수 있었다. 또한, 새로

운 분산 연속질의 시스템은 EPCglobal 아키텍처 프레임워크의 별다른 수정 없이 바로 적용함으로써 새로운 글로벌 서비스의 제공이 가능한 장점을 가지고 있다. 이를 통하여 기업은 RFID 기술에 대한 표준을 준수하면서 적은 비용으로 효과적인 분산 연속질의 처리 시스템을 도입하는 것이 가능하다.

본 논문에서 제시하는 분산 연속질의 시스템은 EPCglobal 네트워크만을 대상 환경으로 삼고 있다. 따라서 분산 연속질의 시스템의 활용성을 높이기 위해서는 EPCglobal 네트워크뿐만 아니라 이기종 분산 데이터베이스 시스템에도 적용할 수 있도록 환경을 일반화시키는 것이 필요하다. 또한, 분산 시스템에 질의를 안정적으로 처리하기 위해서 분산 연속질의 시스템의 신뢰성을 높일 수 있는 방안에 대한 연구도 필요하다.

참 고 문 헌

- [1] EPCglobal, <http://www.epcglobalinc.org>
- [2] EPCglobal, "The EPCglobal Architecture Framework," EPCglobal Inc., 2005
- [3] 백한진, "RTE 구현을 위한 전략," 삼성 SDS IT 건설팀, 2004.
- [4] 한국유통물류진흥원, "업종별 RFID/EPC 확산전략 로드맵," 한국유통물류진흥원, 2005년.
- [5] G. Deshpande and V. Chakravarthy, "IB Master Data Management," Technical Report, IBM, 2005.
- [6] J. K. Park, B. H. Hong, and C. H. Ban, "A Query Index for Continuous Queries on RFID Streaming Data," *Science in China Series F: Information Sciences*, 51(12), pp.2047-2061, 2008.
- [7] S. Madden, M. Shah, J. M. Hellerstein, and V. Raman, "Continuously Adaptive Continuous Queries over Streams," *ACM SIGMOD International Conference on Management of Data*, pp.49-60, 2002.
- [8] A. Arasu et al., "STREAM: The Stanford Data Stream Manager," *IEEE Data Engineering Bulletin*, 26(1), pp.19-26, 2003.
- [9] W. H. Tok and S. Bressan, "Efficient and Adaptive Processing of Multiple Continuous Queries," *EDBT, LNCS 2287*, pp.329-332, 2002.
- [10] A. Arasu, S. Babu, and J. Widom, "The CQL Continuous Query Language: Semantic Foundations and Query Execution," *VLDB Journal*, 15(2), pp.121-142, 2006.
- [11] N. W. Siong, S. Yanfeng, and T. W. Hyong, "CQ-Buddy: Harnessing Peers for Distributed Continuous Query Processing," Dspace MIT, <http://dspace.mit.edu/handle/1721.1/3855>, 2004.
- [12] W. Ng, B. Ooi, and K. Tan, "Bestpeer: A Self-Configurable Peer-to-Peer System," *ICDE*, pp.272, 2002.
- [13] 통계청, "2008년 통계행정 편람," 통계청 기획재정 담당관실, 2008년.
- [14] 관세청, "2007 수출입 물류 통계연보", 관세청 대변인실, 2008년.
- [15] EPCglobal, "Pedigree Specification Version 1.0 Ratified", EPCglobal Inc., 2007.
- [16] SupplyScape, "Combining EPCIS with the Pedigree Messaging Standard", GS1 US, 2008.
- [17] EPCglobal, "EPC Information Services (EPCIS) Specification," Version 1.0, EPCglobal Inc., 2006.
- [18] Jetty 6 - Jetty Web Server, <http://www.mortbay.org>

— 저 자 소 개 —



안 성 우(정회원)
 1999년 부산대학교 컴퓨터공학과
 학사 졸업.
 2001년 부산대학교 컴퓨터공학과
 석사 졸업.
 2009년 부산대학교 컴퓨터공학과
 박사 졸업.

2009년~현재 부산대학교 BK21사업단 박사후
 연구원.

<주관심분야 : RFID 미들웨어, 유비쿼터스 시스템, 이동체 색인, LBS>



홍 봉 회(정회원)
 1982년 서울대학교 컴퓨터공학과
 학사 졸업.
 1984년 서울대학교 컴퓨터공학과
 석사 졸업.
 1988년 서울대학교 컴퓨터공학과
 박사 졸업.

1987년~현재 부산대학교 컴퓨터공학과 교수.

<주관심분야 : RFID 미들웨어 데이터베이스, 실시간 위치정보 시스템, 유비쿼터스 미들웨어>



정 동 규(정회원)
 2007년 부산대학교 컴퓨터공학과
 학사 졸업.
 2009년 부산대학교 컴퓨터공학과
 석사 졸업.
 2009년~현재 NHN(주) 뉴스서비스
 스개발팀 사원.

<주관심분야 : RFID 미들웨어, EPCIS, RTLS 위치 측위>