

# EPCIS에서 MS-SQL과 ALTIBASE의 비교에 관한 연구

단 단\* · 송 영 근\* · 권 대 우\* · 이 두 용\* · 이 종 석\*\* · 이 창 호\*

\*인하대학교 산업공학과 · \*\*남서울대학교 산업경영공학과

## A Study on the comparison between MS-SQL and ALTIBASE in EPCIS

Da DAN\* · Young-Keun Song\* · Dae-Woo Kwon\* · Doo-Yong Lee\* · Zhong-Shi Li\*\* · Chang-Ho Lee\*

\*Department of Industrial Engineering, INHA University

\*\*Department of Industrial & Management Engineering, NAMSEOUL University

### Abstract

EPC Information Services (EPCIS) is an EPCglobal standard designed to enable EPC-related data sharing within and across supply chain. The EPCIS standard defines standard interfaces to enable EPC-related data to be captured and subsequently to be queried using a set of service operations and an associated data model.

There are two kinds of EPCIS data: event data and master data. Event data is created in the process of carrying out business processes. Traceability of goods across supply chain is based on event data. Therefore, each company must have an event data. This study compared the difference between MS-SQL(DRDBMS) and ALTIBASE(MMDBMS) for data storage. We compared the difference between two database management in many respects such as insert time and select time. We come to a conclusion that ALTIBASE is more efficient than MS-SQL.

**Keywords :** EPCIS, MMDBMS, DRDBMS, Hybrid DBMS, RFID

### 1. 서 론

세계적으로 점점 더 치열해지는 기업 간의 경쟁에서 살아남기 위해 기업들은 고객 개개인의 욕구에 맞춘 물품·서비스를 제공해야 하는 필요성이 증가하고 있다. 기업들이 고객 개개인에 맞춘 고객화 서비스를 제공하기 위해 수많은 물품들이 각기 다른 프로세스를 거치게 되고, 그만큼 다양화된 물품들로 인해 기업들도 물품 하나하나를 관리해야 하는 필요성을 인지하고 있다. 하지만 회사에서 그러한 물품을 관리하기 위해 부여한 제품관리코드의 구조가 모두 다르기 때문에 결국 기업 간에는 정보의 공유가 되지 않아 공급사슬의 가시성을 파악하기 힘들고, 중복업무가 발생하는 등의 업무 비효율이 발생하고 있다[1][2].

이러한 애로사항을 해결하기 위해 EPCglobal Network을 활용하게 되면 공급자에서부터 소비자에게 이르기까지 공급망 전체에 걸쳐 실시간 정보처리, 객체의 가시

성 확보, 제품 이력 추적 등의 이점을 얻을 수 있다[7].

EPCglobal Network에서는 현재 4개의 데이터 표준과, 8개의 인터페이스 표준을 발표한 상태이며, 이 중에서 정보처리의 게이트웨이 역할과 정보저장소의 기능을 하는 EPCIS에 대한 표준도 제정되어 있으나, EPCIS 표준에는 EPCIS Repository를 다양한 형태를 취할 수 있는 어플리케이션으로 정의하며 EPCIS Repository에 대한 세부적인 명세는 제공하지 않는다[3]. 이는 EPCIS Repository가 기능적인 측면에서 EPCIS Event Capture Interface를 통해 전달된 Events data를 저장하고 관리하기 위해 관계형 database, XML database, File system 혹은 다른 영구적인 저장 시스템이 될 수 있으며, EPCIS Query Interface를 통해 저장된 Events data를 다른 어플리케이션에게 제공할 수 있음을 의미한다[9]. 따라서 인터페이스에 대한 규정은 존재하되 EPCIS Repository를 구축하는 것은 당사자의 자율에 따르므로 EPCIS Repository를 효율적으로

† 본 연구는 한국연구재단 지원에 의하여 연구되었음.

† 교신저자: 이창호, 인천시 남구 용현동 253, 인하대학교 산업공학과

M · P: 010-3761-2995, E-mail: lch5601@inha.ac.kr

2010년 4월 20일 접수; 2010년 6월 7일 수정본 접수; 2010년 6월 8일 게재확정

구성하는 것은 기업이나 기관의 몫이라 할 수 있다.

본 연구에서는 물류과정에서 발생하는 EPC data를 저장하는 저장소인 EPCIS Repository 효율화 방안으로 디스크 기반의 DBMS와 디스크 및 메모리 기반의 Hybrid DBMS의 특성과 성능을 비교하고자 한다. 신뢰할 수 있는 결과를 얻기 위해 실제 물류 센터에서 발생하는 비즈니스 프로세스를 바탕으로 시나리오를 작성하고, 이에 따라 발생하는 EPC data를 DRDBMS와 Hybrid DBMS에 각각 저장하고 검색하여 두 DBMS의 성능을 비교하였다. 본 연구에서는 DRDBMS는 MS-SQL을 사용하였고, Hybrid DBMS는 ALTIBASE를 사용하였다.

## 2. 이론적 배경

### 2.1 EPCIS

Electronic Product Code Information Services (EPCIS)는 EPCglobal에서 기업내부와 기업 간의 EPC와 관련된 data를 공유하기 위해 설계한 표준이다. 이 data 공유는 EPCglobal Network에 참여한 기업들끼리 비즈니스 단계 내에 있는 EPC와 관련된 물품의 가시성을 확보하기 위해 사용된다. 하지만 강제가 아닌 권고사항 정도의 표준으로 적용되며, 최소한의 유용한 집합으로써 식별이 필요한 상황에 기본적인 기능을 제공하고, 물류산업이 아닌 다른 산업에도 적용이 가능하다.

EPCIS 표준은 EPC와 관련된 data를 수집하고, 이후 서비스 과정과 data 모델과 관련된 집합에 질의할 때의 표준을 정의한다. EPC와 관련된 수집과 질의 인터페이스는 database와 그에 접근하는 어플리케이션을 포함한다[1][2].

### 2.2 DBMS

DBMS(Database Management System)는 database를 사용, 생성, 유지보수하기 위한 컴퓨터 프로그램의 집합체이다.

DBMS는 DRDBMS(Disk Resident DBMS), MMDBMS(Main Memory DBMS), Hybrid DBMS의 세 가지로 나타낼 수 있다. DRDBMS는 보조기억장치인 하드디스크에 정보를 저장한다. 디스크에 직접 정보를 저장하기에 다른 DBMS보다 느린 속도를 지니고 있으나 MMDBMS에 비해 안정적인 database구축이 보장되며 보조기억장치의 발달로 인해 테라바이트 이상의 무제한적인 database공간의 확장이 가능하다.

MMDBMS는 주기억장치에 정보를 저장하기 때문에 빠른 처리속도를 지니고 있으나 전원공급이 중단되면 저장된 data가 지워지는 휘발성 메모리에 데이터를 저장하기 때문에 data의 안정성에 문제가 나타나고 점차 주기억장치의 용량이 증가하고는 있으나 시스템적으로 제한된 확장성을 지니고 있다[8].

최근에는 DRDBMS와 MMDBMS의 장점만을 보완한 Hybrid DBMS가 출시되고 있다. Hybrid DBMS는 주기억장치와 하드디스크 모두에 정보를 저장할 수 있는 구조를 지니고 있으며, 서로 data를 교환할 수 있는 기능을 지니고 있다. 자주 select되는 data들은 주기억장치부분에 저장하고 장기보관이 요구되는 data들은 보조기억장치에 저장하는 방식으로 사용할 경우 MMDBMS만큼의 빠른 처리속도와 DRDBMS의 확장성 및 안정성을 지닐 수 있다.

본 논문에서는 DRDBMS로 대표적인 상용 모델인 Oracle, MS-SQL, Sybase, Informix 중에서 연구실에서 보유하고 있는 MS-SQL2005 버전을 사용하였고, Hybrid DBMS로는 상용 모델 중에서 국내외에서 그 성능을 검증받은 국산 Hybrid DBMS인 ALTIBASE v.5를 사용하였다.

## 3. EPCIS Repository 구축

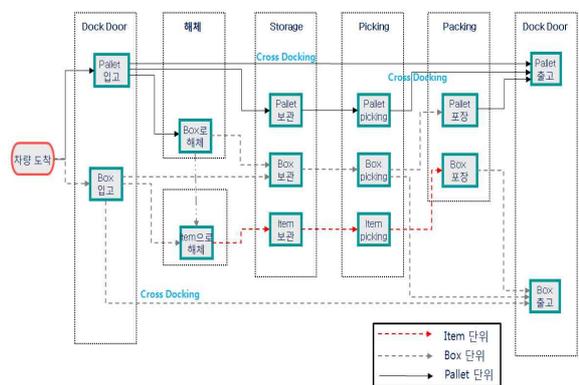
### 3.1 물류센터

물류센터의 업무 시나리오 설정을 위해 국내에 있는 도 단위 총판 의약품 물류센터인 J 의약품 물류센터를 모델로 하여 시나리오 작성하였다. [그림 1]은 참고한 물류센터의 비즈니스 프로세스를 간소화하여 작성한 것이다. 시물레이션 물동량은 통합물류센터를 가정하여 설정하였다.

물류센터에서 각 단계별 비즈니스 프로세스는 <표 1>과 같이 정의하였다.

### 3.2 Schema

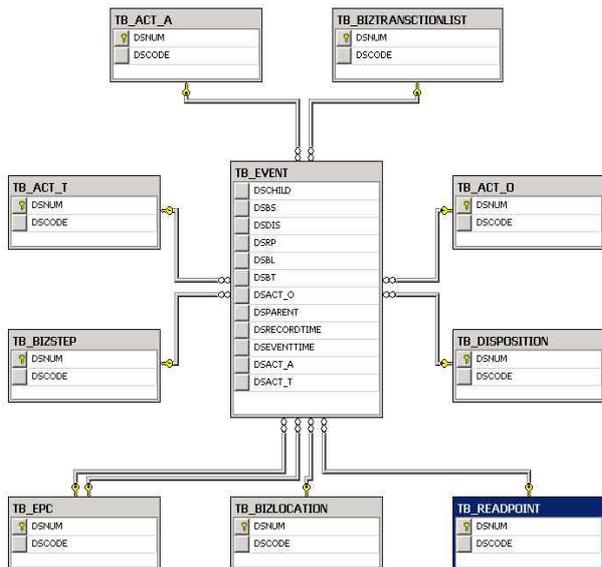
다음 [그림 2]와 [그림 3]은 사용한 Schema를 나타내고 있다. EPC, ReadPoint, Disposition, BizLocation, BizTransactionList, 각 Action 정보들은 해당 테이블에 저장되고 그 값을 정수형으로 TB\_EVENT, TB\_QUANTITY 테이블에서 참조한다. TB\_EPC와 TB\_BizTransactionList 테이블을 제외한 나머지 테이블에는 적은 양의 data만



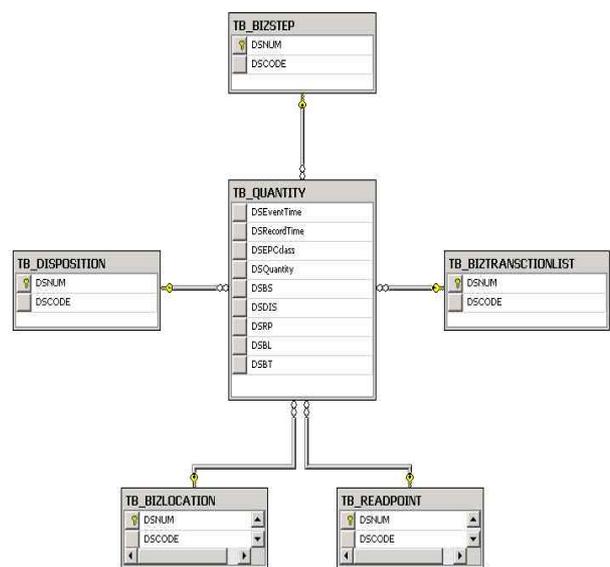
[그림 1] 물류센터 시나리오

<표 1> 물류센터의 단계별 비즈니스 프로세스

업무단계	RFID 적용 후 프로세스
입고 Dock Door (2시간/1일)	- 1~8번 입고 Gate : 300초에 한 번씩 박스당 30개의 Item이 포장된 20개의 박스가 파렛트에 실려서 입고 - 9~10번 입고 Gate : 300초에 한 번씩 박스당 20개의 Item이 포장된 박스 입고
Pallet 및 Box 해체 (Break down) (2시간/1일)	- 1번 Pallet 해체: 450초에 한번씩 박스당 30개의 Item이 포장된 20개의 박스가 실린 한 개의 파렛트가 박스로 해체 - 2번 Box 해체: 300초에 한번씩 박스당 30개의 Item이 포장된 박스가 아이템 단위로 해체 - 3번 Box 해체: 300초에 한번씩 박스당 20개의 Item이 포장된 박스가 아이템 단위로 해체
보관 Zone (1시간/1일)	- 1~2번 보관구역: 300초에 한번씩 박스당 30개의 Item이 포장된 20개의 박스를 실은 1개의 파렛트가 지게차에 실려서 보관 Zone으로 이동 - 3~4번 보관구역: 180초에 한번씩 박스당 20개의 Item이 포장된 박스가 컨베이어 벨트를 통해 보관 Zone으로 이동 - 5~6번 보관구역: 180초에 한번씩 박스당 30개의 Item이 포장된 박스가 보관 Zone으로 이동 - 7~8번 보관구역: 300초에 한번씩 100개의 Item이 보관 Zone으로 이동
Picking(Case 포장) (3시간/1일)	- 1~2번 Pallet Picking : 450초에 한번씩 박스당 30개의 Item이 포장된 20개의 박스를 실은 1개의 파렛트 Picking - 3번 Box Picking Zone: 120초에 한번씩 박스당 30개의 Item이 포장된 1개의 박스 Picking - 4번 Box Picking Zone: 120초에 한번씩 박스당 20개의 Item이 포장된 1개의 박스 Picking - 5~6번 Item Picking Zone: 300초에 한번씩 100개의 Item Picking
Pallet 및 Box 포장 (2시간/1일)	- 1~4번 Pallet Packing: 60초에 한번씩 박스당 20개의 Item이 포장된 박스가 Packing 위해 컨베이어 벨트로 이동 - 5~6번 Box Packing: 60초에 한번씩 30개의 Item이 박스 Packing 위해 이동
출고 Dock Door (2시간/1일)	- 1~3번 출고 Gate : 120초에 한 번씩 박스당 30개의 Item이 포장된 20개의 박스가 한 개의 파렛트에 실려 출고 - 4~6번 출고 Gate : 30초에 한 번씩 박스당 20개의 Item이 포장된 박스가 컨베이어 벨트를 통해 출고 - 7~9번 출고 Gate : 30초에 한 번씩 박스당 30개의 Item이 포장된 1개의 박스가 수작업을 통해 출고



[그림 2] TB\_EVENT 테이블



[그림 3] TB\_QUANTITY 테이블

입력되어있고, TB\_EPC와 TB\_BizTransactionList 테이블에는 인덱스를 설정하여 커다란 성능저하 없이 처리할 수 있도록 schema를 구성하였다.

우선 입고와 재포장(Packing)시 추가되는 EPC들은 모두 TB\_EPC테이블에 저장되고 이후의 모든 프로세

스에서 그 값을 참조하여 다른 정보들과 함께 TB\_EVENT 및 TB\_QUANTITY 테이블에 저장된다. 이벤트별로 ReadPoint, Disposition, BizLocation, Bizstep data 역시 이벤트가 발생할 때의 정보를 해당 테이블에서 찾아서 TB\_EVENT와 TB\_QUANTITY 테이블에 저장되는 구조이다.

## 4. DRDBMS / Hybrid DBMS의 비교

### 4.1 시뮬레이션 프로그램

시뮬레이션 프로그램은 Visual Basic 6.0으로 작성하였고, DB로의 연결은 신속한 database로의 작업을 위해 OLEDB를 사용하였다. 각 포장단위(팔레트, 박스)별로 하나의 객체로 취급되어 프로세스를 이동하는 것을 구현하기 위해 프로세스별로 배열을 생성하여 프로그램의 시작 전에 미리 배열에 모든 EPC를 저장해두고 프로세스 및 ReadPoint 별로 필요한 EPC들을 불러와서 사용할 수 있도록 개발하였다.

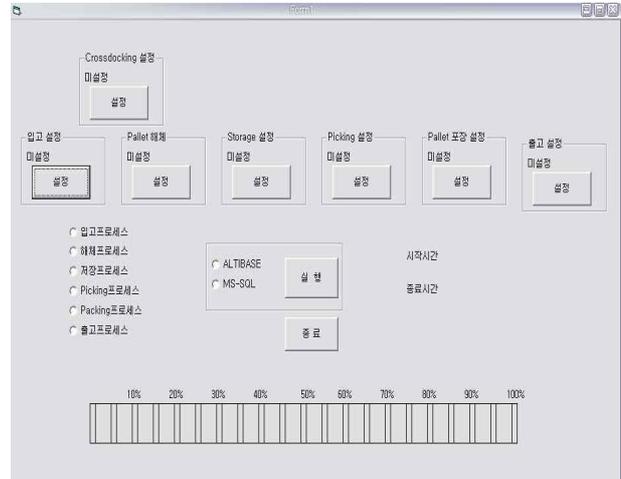
동일 프로세스라도 ReadPoint별로 Event가 발생하므로 ReadPoint별로 작동을 하기위해 Timer를 ReadPoint의 수만큼 구성하여 총 10개의 Timer가 각각의 ReadPoint를 대신하여 입고 Event를 발생하고, 발생된 EPC를 우선 TB\_EPC 테이블에 저장한 후, 그 EPC에 해당하는 DSNUM을 검색하여 그 값을 외래키로 사용하여 TB\_EVENT 테이블에 저장하는 방식으로 insert가 이루어지게 된다. 또한 실제상황에서는 각각의 ReadPoint가 다른 연결을 점유하기 때문에 10개의 ReadPoint에 각기 다른 ADODC객체를 부여하여 독립적인 작업이 이루어질 수 있도록 구성하였다.

각각의 프로세스에 대한 발생간격, 인식되는 EPC의 수는 설정버튼으로 조절이 가능하고, 각 프로세스의 설정을 마치게 되면 실행하고자 하는 프로세스와 접속하고자 하는 DBMS를 선택한 후 실행버튼을 누르게 되면 설정된 정보를 바탕으로 시뮬레이션이 시작된다. 기초 폼에는 후 실행4에서 볼 수 있듯이 시작시간과 종료시간 그리고 프로세스별 진행률을 파악할 수 있다.

프로그램이 종료된 후에는 전체 프로그램이 실행된 시간과 DB에 질의문을 실행하는 총 시간, 그리고 각 발생간격동안의 프로그램 실행시간 및 질의문 수행시간을 텍스트 파일로 출력하도록 하여 테스트결과를 파악할 수 있도록 하였다. 수행된 질의문은 다음과 같은 형태를 지니고 있다. 'insert into tb\_event(dseventime, dsrecordtime, ..., dsent\_a, dsent\_t) values(date, sysdate, ..., (select dsnum from tb\_ent\_t where dscode=ent\_t))의 형태로 질의문이 database에서 처리되게 되고, 하나의 insert문에는 9개의 select 질의가 포함되어있는 형태이다.

<표 2> Server Specification

모델명	HP G5 Storage Server
CPU	쿼드코어 Intel Xeon E5345 2.33GHz
램	32GB
OS	Windows server 2008 R2 Standard Edition



[그림 4] 시뮬레이션 프로그램

### 4.2 테스트 환경

<표 2>의 장비에 MS-SQL 2005와 ALTIBASE v.5를 설치하고 개인 PC에서 시뮬레이션 프로그램을 실행하였다. 시뮬레이션은 10회 진행하였고 각 횟수마다 5,010개의 EPC들이 54회씩 누적되어 1회당 총 270,540개의 EPC가 발생된다. 각 시뮬레이션은 동일한 조건을 유지하기 위해 한 번의 시뮬레이션이 종료되면 database를 삭제한 후 다시 생성하여 시뮬레이션을 진행하였다.

### 4.3 결과분석

#### 4.3.1 시뮬레이션 결과분석

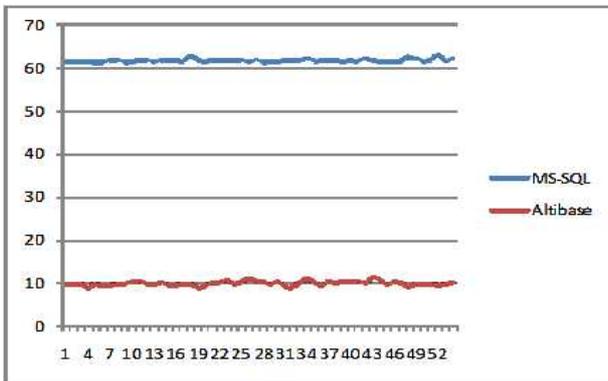
다음 <표 3>과 <표 4>는 10회의 시뮬레이션에서 발생한 select가 포함된 insert 질의를 입고 시나리오에 맞춰 진행한 결과의 평균값을 나타내었다. [그림 5]는 ALTIBASE와 MS-SQL의 결과를 비교하여 그래프로 표현한 것이다.

<표 3> 처리시간 (ALTIBASE)

횟수	시	분	초	1/1000 초	결린 시간	시간 차이	
1	0	0	9	837	9.837	9.837	
2	0	0	19	498	19.498	9.661	
3	0	0	29	190	29.19	9.692	
...	...	...	...	...	...	...	
52	0	8	36	169	516.169	9.424	
53	0	8	46	96	526.096	9.927	
54	0	8	56	78	536.078	9.982	
						MAX	11.207
						MIN	8.733
						AVERAGE	9.927

<표 4> 처리시간 (MS-SQL)

횟수	시	분	초	$\frac{1}{1000}$ 초	결린 시간	시간 차이
1	0	1	1	509	61.509	61.509
2	0	2	3	137	123.137	61.628
3	0	3	4	581	184.581	61.444
...	...	...	...	...	...	...
52	0	53	32	108	3212.108	62.98
53	0	54	33	569	3273.569	61.461
54	0	55	35	834	3335.834	62.265
MAX						62.980
MIN						61.189
AVERAGE						61.775



[그림 5] MS-SQL과 ALTIBASE의 비교(입고사나리오 적용)

본 논문에서는 EPCIS Repository에서 처리시간을 줄이기 위해 Hybrid DBMS를 제시하였으며, 구성된 시나리오를 바탕으로 하여 현재 보편적으로 사용되고 있는 DRDBMS와 Hybrid DBMS에서 EPCIS Event data를 생성하여 입고 프로세스의 처리시간을 비교하였다.

테스트의 결과를 살펴보면 Hybrid DBMS가 DRDBMS보다 빠른 처리시간을 나타내고 있다. 시뮬레이션 결과를 살펴보면 ALTIBASE를 사용했을 시에 MS-SQL보다 6배 정도 빠른 처리속도를 나타내고 있다. 이러한 결과가 발생하는 이유는 주기억장치에서 바로 data를 처리하는 방식의 Hybrid DBMS와 보조기억장치에서 저장공간을 찾고, 그 위치에 data를 처리하는 방식의 DRDBMS의 특성에서 나타난 결과이다.

### 4.3.2 정성적 기대효과

주기억장치에 기록되어있는 data들을 빈번하게 사용되는 Hot data와 자주 사용되지 않는 Cold data로 구분하여 Hot data는 빠른 처리를 위해 주기억장치에 저장하고, Cold data는 빠르게 처리할 필요가 없기 때문에 보조기억장치에 저장하는 구조로도 hybrid DBMS를 이용이 가능하다.

이미 발생되어 있는 Hot data는 일정기간이 지나면 새로 발생한 Hot data보다 덜 사용되게 되므로 특정 기간을 지정

하여 Cold data로 변환하여 ALTIBASE의 Move 질의를 이용하여 보조기억장치로 이전시키는 방법을 사용한다면 계속해서 주기억장치의 필요용량이 늘어나는 것을 방지할 수 있다.

## 4. 결론 및 향후 연구과제

본 연구에서는 EPCIS Repository를 구축하고자 하는 기업에게 작은 용량을 소모하면서도 안정적이고 빠른 응답시간을 가져다줄 수 있도록 Hybrid DBMS를 활용하는 선행 연구로써 시뮬레이션을 통해 Hybrid DBMS와 DRDBMS의 성능을 비교하였다. 시뮬레이션을 위해 실제 의약품 물류센터의 비즈니스 프로세스를 분석하였고, 물동량은 통합 물류센터의 비즈니스 시나리오를 가정하여 설정하였다. 테스트를 위해 Hybrid DBMS인 ALTIBASE와 DRDBMS인 MS-SQL을 사용하여 ALTIBASE가 MS-SQL보다 data insert 시간이 6배 가량 빠르다는 것을 확인할 수 있었다.

물류센터의 시나리오를 작성하고 시나리오대로 database에 event와 관련된 data들이 Hybrid DBMS와 DRDBMS에 저장되는 simulator를 개발하여 테스트를 진행하였고, 테스트 결과는 Hybrid DBMS인 ALTIBASE가 DRDBMS인 MS-SQL보다 6배정도 빠르게 data를 처리하는 것으로 나타났다. 또한 data를 Hot data, Cold data로 구분하여 주기억장치와 보조기억장치에서 각각 처리할 경우 주기억장치의 과도한 확장이 필요치 않게 된다. 이러한 분석으로 인해 Hybrid DBMS를 사용하는 것이 일반적인 DRDBMS를 사용하는 것보다 더 경제적이고, 실시간 혹은 빠르게 처리되어야 하는 data의 양이 많은 물류센터관리시스템에 적합하다.

본 논문에서는 database와 질의문을 거의 튜닝하지 않고 진행을 하였기에 실제 EPCIS Repository를 구축하기 위해서는 본 논문의 schema와 Hybrid DBMS를 이용하되 전문 DBA를 통해 database와 질의문의 튜닝이 이루어지게 된다면 더욱 나은 결과를 산출할 수 있을 것이다.

향후 연구로는 schema 최적화에 대한 연구가 필요하다. 최적화된 schema에 반드시 필요한 정보만 저장하여 data 용량을 줄일 수 있고 이것은 insert 및 select 시간에 직접적으로 영향을 미치기 때문이다[1].

## 5. 참고 문헌

- [1] Alexander Ilic, Thomas Andersen, Florian Michahelles, "EPCIS-based Supply Chain Visualization Tool", Business Processes & Applications, 2009.
- [2] Alexander Ilic, Thomas Andersen, Florian Michahelles, "increasing Supply-Chain Visibility with Rule-Based RFID Data Analysis", IEEE Internet Computing, 2009.
- [3] EPCglobal, "EPC Information Services (EPCIS) Version 1.0.1 Specification", EPCglobal, September 2007.

- [4] EPCglobal, "The EPCglobal Architecture Framework Version 1.3", EPCglobal, September 2009.
- [5] Seshadri, A. Silberschatz, and S. Sudarshan, "Distributed Multi-Level Recovery in Main-Memory Databases", Proc. of the International Conference on Parallel and Distributed Information Systems, 1996.
- [6] Verisign, Inc., "The EPCglobal Network: Enhancing the Supply Chain", 2005.
- [7] 안재명, 이종태, 오해석, (주)리테일테크 기술 연구소 공저, "EPCglobal Network 기반의 RFID 기술 및 활용", 글로벌, 2007. 2.
- [8] 이규용, "고가용성을 위한 주기억장치 DBMS ALTIBASE의 이중화 관리 기법", 한국인터넷 정보학회, 6권, 1호, 2005, 2.
- [9] 조용철, "RFID기반의 통합물류센터를 위한 효율적인 EPCIS Repository 구축에 관한 연구", 인하대학교 박사학위논문, 2009.
- [10] 조이현, "RFID 기반 물류환경에서의 실시간 데이터 처리를 위한 EPCIS의 설계 및 구현", 부산대학교 석사학위논문, 2007.

### 저 자 소 개

#### 단 단



중국 남경공업대학교 산업공학과 학사 취득. 현재 인하대학교 대학원 산업공학과 석사과정 중.  
관심분야 : SCM, RFID 관련 물류관리 시스템 개발, RFID를 활용한 항공물류 정보시스템 등.

주소: 인천광역시 남구 용현동 253, 인하대학교 산업공학과

#### 송영근



인하대학교 산업공학과 공학사 취득. 현재 인하대학교 대학원 산업공학과 석사과정 중.  
관심분야 : SCM, RFID 관련 물류 관리 시스템 개발, EPCglobal Network 시뮬레이션 등

주소: 인천광역시 남구 용현동 253, 인하대학교 산업공학과

#### 권대우



인하대학교 산업공학과 공학사 취득. 현재 인하대학교 대학원 산업공학과 석사과정 중.  
관심분야 : SCM, RFID 관련 물류 관리 시스템 개발, 항공물류 RFID 시스템 개발, LBS 등

주소: 인천광역시 남구 용현동 253, 인하대학교 산업공학과

#### 이두용



인하대학교 대학원 산업공학과 석사 취득. 현재 인하대학교 대학원 산업공학과 박사과정 중.  
관심분야 : RFID 관련 물류 관리 시스템 개발, 항공물류 RFID 시스템 개발, SCM, LBS 등

주소: 인천광역시 남구 용현동 253, 인하대학교 산업공학과

#### 이종석



인하대학교 대학원 산업공학과 공학박사취득. 현재 남서울대학교 산업경영공학과 교수로 재직 중.  
관심분야 : 항공물류 정보시스템, 시뮬레이션, EPCglobal Network, SCM, ERP 등.

주소: 충남 천안시 성환읍 매주리 21 남서울대학교 산업경영공학과

#### 이창호



인하대학교 산업공학과에서 학사 취득. 한국과학기술원에서 산업공학과 석사, 경영과학과 공학박사 취득. 현재 인하대학교 교수로 재직 중.  
관심분야 : 물류, RFID, SCM 등.

주소: 인천광역시 남구 용현동 253, 인하대학교 산업공학과