

무선 환경을 지원하는 데이터베이스 통합 미들웨어 설계 및 구현

이상윤*, 성준호**, 이미영*, 김명준*

*한국전자통신연구원 컴퓨터소프트웨어연구소 컴퓨터시스템연구부

**한양대학교 정보통신대학원 정보통신공학과

e-mail : syllee@etri.re.kr

Design and Implementation of Database Integration Middleware Supporting Wireless Environments

Sang-Yun Lee*, Jun-Ho Seong**, Mi Young Lee*, Myung-Joon Kim*

*Computer System Dept., CSRL, ETRI

** Graduate School of Information & Communication, Hanyang Univ.

요 약

이질적인 정보 저장소를 통합하여 질의하고 검색하는 시스템들에 대한 연구가 최근 들어 XML 을 기반으로 활발히 연구되고 있다. 본 논문에서는 XML 에 기반한 DB 통합 미들웨어로서 DataBlender 시스템을 소개하고 이 시스템의 한 블록으로서 무선 환경을 지원하기 위해 개발된 DataBlender.s 의 설계 및 구현에 대해 기술한다. DataBlender.s 는 동기화의 표준 언어인 SyncML 1.0 을 기반으로 개발되었으며 본 논문에서는 DataBlender.s 의 모듈 구조, 타 모듈과의 연동, 동기화 처리 과정, 동기화 정보 관리 등을 기술한다.

1. 서론

최근 이질적인 정보 저장소를 통합하여 질의하고 검색하는 시스템에 관한 연구가 활발히 이루어지고 있다. 특히 차세대 인터넷 표준 언어인 XML 이 널리 보급되고 활용됨에 따라 XML 을 매개로 한 정보 저장소 통합이 대두되고 있다. 일반적으로 정보 저장소 통합 시스템은 각 정보 저장소의 정보를 추출할 수 있는 Wrapper 와 여러 Wrapper 로부터 추출한 정보를 통합하는 Mediator 로 구성되어 있다. 본 논문에서 소개하는 DataBlender 시스템은 무선 환경에서도 통합 검색이 가능한 데이터베이스 통합 시스템으로 메타 정보를 단말기에 복제하여 검색시 활용한다. 무선 환경에서 데이터를 단말기에 복제하여 검색하는 이동 DBMS 에 관한 연구는 [1], [2]등에서 보인 바 있다. 최근에는 동기화의 표준 프로토콜인 SyncML 이 제정되어 이를 기반으로 한 동기화 연구가 활발히 이루어지고 있다. 본 논문에서는 무선 환경에서 스카마 복제 및 동기화를 위해 SyncML 을 이용한 동기화 표현 및

동기화 과정과 내부 처리 방법에 대해서 제안한다. 본 논문의 구성은 다음과 같다. 2 장에서는 무선 환경을 위해 고려할 사항을 기술하고 3 장에서는 DataBlender 시스템을 소개하고 4 장에서는 메타 정보 동기화 설계를 기술하고 5 장에서 결론을 맺는다.

2. 무선 환경을 위한 고려

무선 환경은 유선 환경과는 달리 네트워크 연결을 보장할 수 없다. 접속 단절이 자주 일어나거나 아예 접속이 안 되는 상황 등 무선 네트워킹 기술의 한계도 문제가 되겠지만 무선 인터넷 사용자 부담도 무시할 수 없어 사용자들은 수시로 필요할 때만 네트워크에 접속하여 이용한다.

DataBlender 에서는 이러한 무선 환경을 고려하여 설계되었다. 무선 환경을 지원하기 위해서는 비동기 질의가 가능하여야 한다. 사용자는 질의를 서버에 보낸 후 질의 수행이 완료되기까지 기다리지 말고 다른 일을 할 수 있어야 한다. 연결하기 전에 질의 처리는 구문 검사, 권한 검사, 질의 정당성 검사 등의 전처리

를 먼저 클라이언트에서 수행하고 이 과정이 끝나면 연결을 하여 질의를 수행함으로써 무선 환경의 단점을 극복할 수 있다.

3. DataBlender 개요

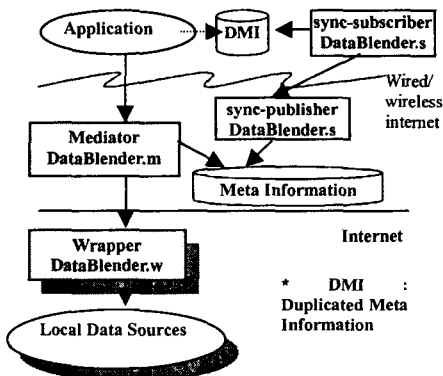
DataBlender 는 미디어이터 (DataBlender.m), 래퍼 (DataBlender.w), 메타 정보 동기화기 (DataBlender.s)로 구성되어 있다.

DataBlender.m 은 통합 스키마를 관리하고 통합 질의를 처리한다. DataBlender.m 은 통합 스키마에 기반한 통합 질의를 지역 스키마에 기반한 질의로 변환하고 변환된 질의를 하위 질의로 분해한 후 그들을 해당되는 래퍼로 전송한다. 그리고 최종적으로 각 래퍼로부터의 결과를 통합하여 사용자에게 통합된 결과를 제공한다.

DataBlender.w 은 미디어이터 서버와 지역 데이터 소스와의 연결을 담당한다. DataBlender.w 은 미디어이터 서버로부터 온 하위 질의를 지역 데이터 소스의 질의 언어로 변환한 후 질의를 실행하고, 지역 데이터 소스로부터의 질의 결과를 XML 문서 형식으로 변환한다. 그리고 결과 XML 문서를 미디어이터 서버로 전송한다.

DataBlender.s 는 메타 정보를 복제하고 복제된 메타 정보와 마스터 메타 정보와의 일치성을 유지한다. 이는 모바일 환경을 지원하기 위한 것이며 응용 프로그램과 미디어이터 서버와의 연결 시간을 축소시키고 가능한 한 연결이 단절되더라도 질의 처리를 계속 수행할 수 있도록 한다[3].

[그림 1]은 DataBlender 의 구조를 나타낸다.



[그림 1] DataBlender 의 구조

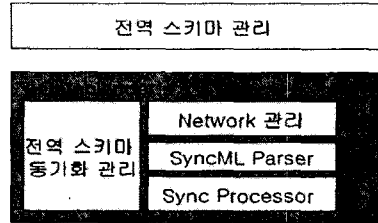
4. 메타 정보 동기화기

메타 정보 동기화기(DataBlender.s)는 메타 정보의 복제를 무선 단말기에 저장하고 관리하며 무선 환경에서 마스터 메타 정보와 복제 메타 정보 간의 동기화를 수행한다.

4.1 모듈 구조

메타 정보 동기화기는 [그림 2]와 같이 전역 스키마 관리 모듈, SyncML Parser 모듈, Sync Processor 모듈,

Network 관리 모듈, 전역 스키마 동기화 관리 모듈로 구성되어 있다.



[그림 2] 메타 정보 동기화기 블록 구조

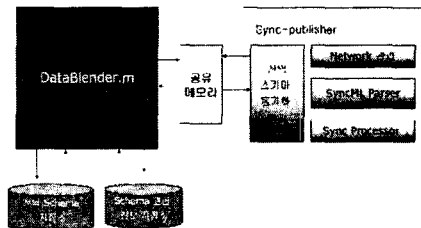
- 전역 스키마 관리
DataBlender.m 이 전역 스키마 정보를 관리하는 모듈로서 메타 정보 동기화기와 공유한다.
- SyncML Parser
동기화 정보를 담고 있는 SyncML 메시지를 분석하고 파싱한 후 내부 자료 모델로 변환한다. 파서는 무선 단말기에서는 SAX 1.0 을 지원하는 파서를 사용하고 서버에서는 DOM 1.0 core level 을 지원하는 파서를 사용한다.
- Sync Processor
클라이언트와 서버간에 동기화 정보를 주고 받은 후 메시지에 담긴 내용을 분석하여 동기화한다.
- Network 관리
SyncML 메시지를 전송하는 모듈로 소켓을 기반으로 한다.
- 전역 스키마 동기화 관리
전역 스키마 동기화 관리 모듈은 동기화 대상별로 전역 스키마 정보를 관리한다. 동기화 대상은 사용자와 단말기에 무관하게 스키마 구독 건당 식별자를 부여한다.

4.2 synchronizer 동작 모델

메타 정보 동기화기는 미디어이터 서버와 연동하여 동기화 될 전역 스키마를 출판 및 관리하는 sync-publisher 와 무선 단말기에서 동기화를 요청하고 동기화를 수행하는 sync-subscriber 로 구성되어 있다.

4.2.1 sync-publisher 의 동작

[그림 3]은 sync-publisher 의 동작 및 구조를 나타내고 있다.



[그림 3] sync-publisher 의 동작 및 구조

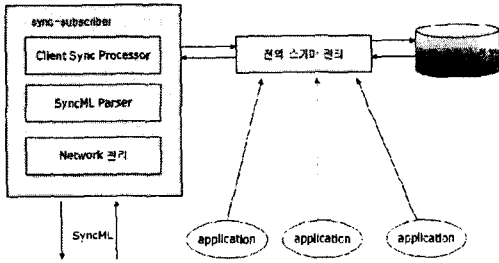
sync-publisher 는 sync-subscriber 가 구독한 전역 스키

마에 한해서 관리하고 있으며 이 스키마에 변경이 발생하면 DataBlender.m 의 스키마 관리 모듈이 변경 내용을 공유 메모리에 반영한다. Sync-publisher 는 공유 메모리의 정보를 읽어봄으로써 마지막 동기화 이후에 발생한 변경 내용들을 확인할 수 있다.

4.2.2 sync-subscriber 의 동작

sync-subscriber 는 미디어이터 서버로부터 구독 가능한 스키마 리스트를 얻어 온 후 sync-publisher 에게 자신이 구독할 스키마 리스트를 알려 준다. 이 때 sync-publisher 가 구독 전에 대한 식별자를 생성하여 sync-subscriber 에게 부여하고 자신도 이 식별자를 관리한다. 구독한 전역 스키마 정보는 로컬 저장소에 저장되고 전역 스키마 관리기나 응용 프로그램이 이를 공유하여 사용한다. 응용 프로그램에서의 동기화 시작은 트랜잭션 시작시에 한다.

그림 [4]는 sync-subscriber 의 동작 및 구조를 나타내고 있다.



[그림 4] sync-subscriber 모듈 구조

4.3 동기화

4.3.1 동기화 프로토콜

sync-publisher 와 sync-subscriber 간의 동기화 프로토콜은 SyncML Sync Protocol 1.0.1 에서 제안하는 7 개의 Sync type 중에서 “One-Way Sync from Server only” 타입을 채택한다[4][5]. 이는 클라이언트에서의 스키마 변경은 발생하지 않고 서버에서만 발생하므로 서버에서의 변경만을 클라이언트에 반영하면 되기 때문이다.

4.3.2 동기화 명령

서버에서 전역 스키마 정보의 변경이 발생하는 경우는 스키마 이름이 변경되는 경우, 기존의 스키마가 삭제되는 경우 스키마의 권한이 변경되는 경우 세 가지가 있다. 이를 표현하기 위해 SyncML 의 Command 중 <Replace>와 <Delete>, <Add> Command 를 활용한다.

(1) Replace(스키마)

미디어이터 서버에서 변경된 전역 스키마 정보를 무선 단말기에도 교체하도록 하는 SyncML 표현은 아래와 같다.

```
<Replace>
<CmdID>3</CmdID>
<Item>
<Source><LocName>BookSchema</Locname>
</Source>
```

```
<Data>eBookSchema</Data>
</Item>
</Replace>
```

위 표현은 “BookSchema” 라는 전역 스키마명을 “eBookSchema”로 변경하라는 것을 나타낸다.

(2) Delete(스키마)

미디어이터 서버에서 삭제된 전역 스키마 정보를 무선 단말기에서도 삭제하도록 하는 SyncML 표현은 아래와 같다.

```
<Delete>
<CmdID>3</CmdID>
<Item>
<Source><LocName>eBookSchema</Locname>
</Source>
</Item>
</Delete>
```

위 표현은 “eBookSchema”라는 전역 스키마를 삭제하라는 것을 나타낸다.

(3) Add(권한)

```
<Add>
<CmdID>6</CmdID>
<Item>
<Source><LocName>DVD</Locname></Source>
<Data>syllce:READ,mclee:WRITE</Data>
</Item>
</Add>
```

위 표현은 “DVD”라는 스키마에 syllce 사용자는 READ 권한을 mclee 사용자는 WRITE 권한을 부여하라는 것을 나타낸다.

(4) Delete(권한)

```
<Delete>
<CmdID>6</CmdID>
<Item>
<Source><LocName>DVD</Locname></Source>
<Data>mylee:WRITE,mio:READ</Data>
</Item>
</Delete>
```

위 표현은 “DVD”라는 스키마에서 mylee 사용자는 WRITE 권한을 mio 사용자는 READ 권한을 취소하라는 것을 나타낸다.

4.4 동기화를 위한 정보 관리

sync-publisher 는 sync-subscriber 가 구독한 스키마 리스트를 관리하고 있어야 하며 각 스키마에 변경이 발생하면 변경 시간과 변경 내용을 관리 하고 있어야 한다. 이를 위해 sync-publisher 는 복제 정보 테이블과 변경 로그 테이블을 관리한다.

복제 정보 테이블에서 subscribed client 는 구독 신청자에 대한 식별자이다. 클라이언트로부터 구독 신청이 들어 오면 sync-publisher 가 식별자를 생성하여 관리하고 이 식별자는 구독 신청을 한 suscriber 도 알고 있어야 한다. 따라서 한 단말기에서 한 사용자 또는 여러 사용자가 각각 다수의 구독 신청을 할 수 있고 이들은 부여받은 식별자를 이용하여 구별된다. Published Info 는 각 구독 전에 대하여 구독한 스키마 명의 리스트로 관리되며 각 스키마에 대해서 구독해

간 시간을 subscribed time 으로 관리한다. 이 시간은 클라이언트 단말기의 시간이 아니고 sync-publisher 의 하드웨어 시간으로 한다.

변경 로그 테이블에서 Published Info 는 구독을 허락한 스키마명을 나타내며 Update time 은 해당 스키마가 변경이 발생한 시점을 나타내고 Update reason 은 이름 변경, 권한 변경, 삭제 등을 나타낸다. 다음의 <표 1> 은 복제 정보 테이블의 모습이고 <표 2>는 변경 로그 테이블의 모습이다.

<표 1> 복제 정보 테이블

Subscribed client	Published Info	Subscribed time
1	Book	20020403T131935Z
	DVD	20020302T100933Z
2	Book	20020306T130925Z
	DVD	20020303T130935Z
3	Customer	20020304T110923Z
100	Sports	20020102T100933Z

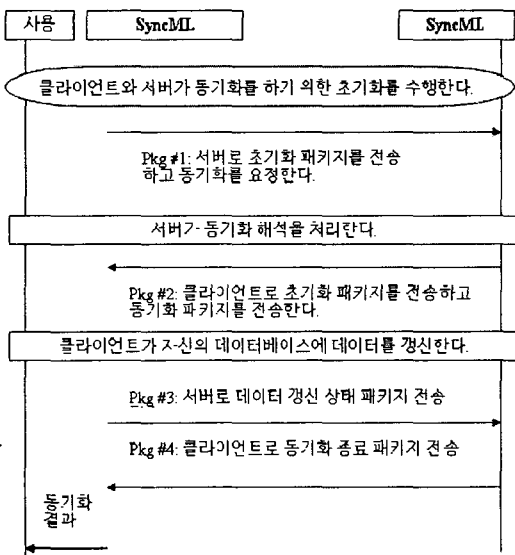
<표 2> 변경 로그 테이블

Published Info	Update time	Update reason
Book	20020302T100933Z	Name change
DVD	20020303T130935Z	Privilage change
Sports	20020303T110923Z	Delete

4.5 동기화 시나리오

동기화는 복제된 메타 정보를 사용하는 응용 프로그램, sync-subscriber, sync-publisher 의 상호 작용에 의해 이루어진다.

[그림 5]는 MSC(Message Sequent Chart)로서 패키지 교환을 통해 동기화 과정을 보여주고 있다.



[그림 5] 동기화 과정

sync-publisher 는 서버 데몬으로 항상 실행 상태에 있다. 응용 프로그램은 sync-subscriber 가 구독해 온 스키마 정보 및 권한 정보를 사용한다. 스키마 구독은 스키마 명이나 스키마 명이 나열된 파일을 읽어들이며 datablender_subscribe 라는 유틸리티를 사용해 구독해올 수 있다. 동기화 요청은 응용 프로그램이 따로 요청하지는 않고 트랜잭션 시작시에 자동으로 동기화 요청이 이루어지므로 응용 프로그램은 항상 최신의 메타 정보로 동기화 되어 있다는 전제하에 수행된다.

트랜잭션이 시작되면 sync-subscriber 는 부여 받은 자신의 식별자를 sync-publisher 로 전송하고 동기화를 요청한다. 그러면 sync-publisher 는 동기화를 담당할 쓰레드를 생성하고 subscribed time 이후에 발생한 변경 내용을 찾아서 sync-subscriber 로 전송한다. Sync-subscriber 는 sync-publisher 로부터 전송 받은 변경 내용을 반영하고 성공했음을 sync-publisher 에게 알린다. Sync-publisher 는 subscribed time 을 갱신하고 동기화를 종료한다. 이러한 동기화 과정은 응용 프로그램이 모르는 사이에 내부적으로 일어난다.

5. 결론

본 논문에서는 XML 기반 DB 통합 미들웨어인 DataBlender 시스템의 한 요소인 DataBlender.s 에 대해서 설계 내용과 처리 과정을 기술하였다. DataBlender.s 는 동기화를 요청하는 sync-subscriber 와 동기화 요청을 받아 변경 내용을 전달해 주는 sync-publisher 로 이루어져 있으며 동기화를 표현하고 처리하기 위해 SyncML 1.0 을 기반으로 하고 있다. 동기화 프로코콜은 서버에서의 변경만을 반영할 수 있는 “One-Way Sync from Server only” 방식을 채택하였고 변경 정보를 관리하기 위해 sync-publisher 는 복제 정보 테이블과 변경 로그 테이블을 관리한다.

본 논문에서 제안한 DataBlender.s 를 활용하면 무선 환경에서 무선 단말기가 스키마 정보를 복제해 온 후 서버 정보와의 일치성을 보장할 수 있기 때문에 서버와의 접속을 최소화 할 수 있다.

향후에는 스키마 정보 동기화 뿐만 아니라 사용자 정보 동기화, DB 데이터 동기화 등으로 확장되어야 할 것이다.

참고문헌

- [1] Integrating Remote Workgroups & Occasionally Connected Devices with the Enterprise, The Sybase Strategy
- [2] J. P. Boone, J. Pederson, Extending Enterprise Data and Applications to Mobile Devices using DB2 Everyplace, IBM's White Paper
- [3] Mi-Young Lee, Myung-Joon Kim, Kyu-Chul Lee, "Design of an XML-based Database Integration Middleware : DataBlender," Proc. Of ICIS 2002, pp. 863-867, 2002
- [4] SyncML Representation Protocol, Version 1.0.1, <http://www.syncml.org>
- [5] SyncML Sync Protocol Version 1.0.1, <http://www.syncml.org>