

컴포넌트 기반 WAP 응용 자동 생성기의 구현*

고민정^o, 강이지, 이성혜, 박은희, 음두현
 덕성여자대학교 전산학과

{mjko^o, izzy}@duksung.ac.kr, {hanmong, skylove102102}@hotmail.com, dheum@duksung.ac.kr

Implementation of a Component-based WAP Application Generator

Minjeong Ko^o, Izzy Kang, Sunghye Lee, Eunhee Park, Doohun Eum
 Dept. of Computer Science, Duksung Women's Univ.

요약

모바일 기반 전자상거래 시장이 급성장해 2004년 아시아 시장에서는 80억 달러를 넘어설 것으로 예상된다. 이렇게 급증하는 모바일 전자 상거래의 수요에 맞게 신속한 서비스를 제공하기 위해 WAP 응용의 생산성 향상이 요구된다. 본 논문에서 소개하는 WAPSiteGen은 데이터베이스부터 구축해야 하는 WAP 응용과 기존 데이터베이스를 사용하는 WAP 응용의 신속한 작성을 위해, 무선 응용의 사용자 인터페이스로 사용되는 WML 데크들과 함께 이 데크를 구성하는 카드들 통해 이뤄질 질의를 처리하는 JSP 웹 컴포넌트 및 EJB 컴포넌트들을 자동 생성하여 WAP 응용의 생산성 향상을 지원하는 도구이다. WAPSiteGen은 EJB 및 JSP 컴포넌트 기술을 도입함으로써 3-계층 구조를 갖는 WAP 응용의 확장성, 재사용성, 이식성 등을 향상시킨다. 또한, 비슷한 기능을 지원하는 타 상용 도구들과 달리 WAPSiteGen이 생성하는 응용은 질의의 대상인 개체와 이 개체와 연관된 모든 개체들의 집합을 한 단위로 검색 할 수 있어 연관된 정보들을 신속하게 제공할 수 있다.

1. 서론

세계적인 IT컨설팅 회사인 액센추어와 IT시장 조사기관인 IDC는 한국·중국·대만·홍콩·싱가포르·말레이시아 6개국 225명의 최고경영자(CEO)들을 대상으로 최근 조사한 결과를 토대로 모바일 인구의 급증으로 아시아 지역의 모바일 기반 전자상거래 시장이 급성장해 2004년엔 80억 달러를 넘어설 것으로 예상했다. 특히 한국시장은 2003년 10억 달러를 돌파하고, 2004년에는 30억 달러를 넘어서 가장 큰 시장을 형성할 것으로 전망했다[1]. 따라서 급증하는 모바일 인구의 요구에 부응하기 위해 모바일 전자 상거래 서비스의 생산성 향상이 요구되고 있다.

휴대폰을 이용하는 무선 통신 기술 중 대표적인 것으로는 WAP 포럼에서 제정한 WAP(Wireless Application Protocol)[2]이 있다. WAP을 기반으로 한 무선 인터넷 콘텐츠 서비스는 무선 인터넷 마켓업 언어인 WML(Wireless Markup Language)[3]을 사용해 실현하게 된다. WML에서 카드(card)는 무선 장비의 화면 표시 단위이고 데크(deck)는 하나의 WML 문서로 카드들의 집합을 말한다.

현재의 모바일 전자 상거래 응용은 기존에 유사한 응용이 있어도 처음부터 새로 작성되고 있다. 그러나 소프트웨어의 부패화, 분산화, 통합화, 및 개방화를 지원하는 컴포넌트 기술을 도입하면 응용을 빠르고 효율적으로 생산할 수 있다. 본 논문에서 구현한 WAPSiteGen은 EJB(Enterprise Java Beans)와 JSP(Java Servlet Pages)의 컴포넌트 기술을 도입함으로써 3-계층(3-tier) 구조를 가지는 WAP 응용의 생산성, 확장성, 재사용성, 이식성 등을 향상시킨다. 특히, 무선 응용의 생산성을 높일 수 있어 기하급수적으로 증가하고 있는 무선 응용에 대한 수요를 충족할 수 있다.

본 논문에서 소개하는 WAPSiteGen은 WAP 응용의 신속한 작성을 위해, WAP 응용의 사용자 인터페이스로 사용되는 WML 데크들과 함께 이 데크를 구성하는 카드들 통해 이뤄질 질의를 처리하는 JSP 웹 컴포넌트 및 EJB 컴포넌트들을 자동 생성하는 도구이다. WAP 응용에 필요한 데이터베이스는 클래스 다이어그램을 이용해 새로 구축하거나[4] 기존의 데이터베이스를 사용한다. 잘 알려진 상용 WAP 응용 자동 생성기로는 Microsoft의 Visual Studio .NET[5]과 Anybil의 AnyBuilder Enterprise[6], 그리고 Speedware의 MobileDev[7] 등이 있다. Visual Studio .NET, AnyBuilder, 그리고 MobileDev는 GUI를 사용해 WML 데크들을 반자동으로 생성한다. 생성된 WML 데크는 GUI 설계 시 지정된 테이블에 관한 질의 처리 결과만 담을 수 있다. 반면 WAPSiteGen은 응용 데이터베이스로부터 자동으로 무선 응용을 생성하고 생성된 WAP 응용은 현재 질의의 대상인 닷 개체(anchor entity)는 물론 닷 개체와 일대일 또는 다대일 관계 타입으로 연관되는 첨부 개체(appended entity)와 일대다 또는 다대다 관계 타입으로 연관되는 확장 개체(expended entity)들의 집합을 한 단위로 검색한

다. 검색된 정보는 하나의 데크로 구성되어 무선 장비에 전송되어 무선 장비의 인터페이스로 제공된다. 따라서 관심의 대상인 닷 개체와 연관된 정보들을 한번의 질의로 신속하게 접근할 수 있다.

본 논문에서는 WAPSiteGen이 생성하는 WAP 응용의 동작 원리를 보이기 위해 그림 1의 클래스 다이어그램으로부터 데이터베이스를 생성한 후 그를 이용하는 간단한 전자상거래 응용(Titan 유람선 예약 시스템)을 생성한다. 그림 2는 CUSTOMER를 닷 개체로 선택했을 때 생성되는 WML 데크의 일부로 수정과 삭제 수행하는 카드들이다. 이 데크엔 한번의 질의로 닷 개체인 CUSTOMER, CUSTOMER와 일대다 관계인 ADDRESS와 CREDIT_CARD, CUSTOMER와 일대다 관계인 PHONE와 RESERVATION, RESERVATION과 다대일의 관계인 CRUISE, 그리고 CRUISE와 다대일의 관계인 SHIP에 관한 정보를 모두 담을 수 있다. 데크에 포함된 모든 개체들의 정보는 무선 장비에 전송되어 그림 2와 같이 브라우저에 보일 수 있고 사용자의 조작에 의해 수정 및 삭제될 수 있다.

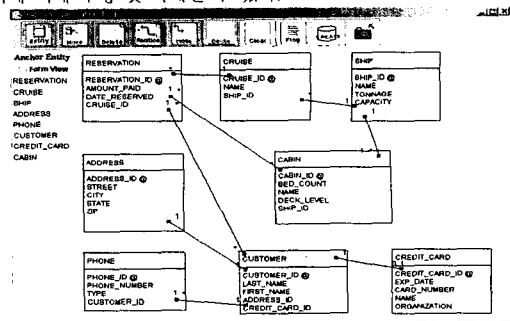


그림 1. Titan 응용의 클래스 다이어그램

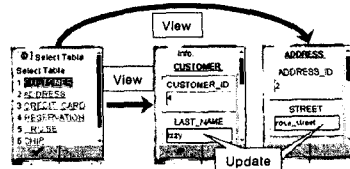


그림 2. CUSTOMER UPDATE/DELETE 데크

2절에서 WAPSiteGen의 개요를 설명하고 3절에서는 WAPSiteGen과 타 상용 WAP 응용 자동 생성기를 비교하고 4절에서는 WAPSiteGen의 구현에 대해 설명한 후, 마지막 절인 5절에서는 본 논

* 본 연구는 한국과학기술원 목격기초연구 (R06-2002-003-01004-0(2003)) 지원으로 수행되었음.

문의 결론과 앞으로의 연구 방향을 정리한다.

2. WAPSiteGen의 개요

본 절에서는 WAPSiteGen의 개요를 설명한다. WAPSiteGen은 프리젠테이션, 비즈니스 로직, 데이터베이스의 3-계층 구조를 가지는 WAP 응용을 자동 생성한다. WAPSiteGen은 클래스 다이어그램을 이용하여 응용 데이터베이스를 새로 구축하거나 기존의 데이터베이스를 사용할 수 있다. 또한, 데이터베이스와 연동하여 비즈니스 로직을 담당하는 EJB 컴포넌트들과 이 EJB 컴포넌트들을 사용하여 사용자 질의를 처리하는 JSP 웹 컴포넌트들을 자동 생성한다. 그리고 WAPSiteGen은 사용자 질의에 대한 결과 데이터를 WML 데크에 담아 프리젠테이션 층을 생성한다.

WML 데크 생성 시 다음의 개체 그룹화 규칙을 통해 한번의 질의 처리로 하나의 데크에 담을 수 있는 개체 타입(테이블)들을 결정할 수 있다.

1. 닷 개체와 일대일 또는 다대일 관계인 개체를 첨부 개체라 하고 일대다 또는 다대다 관계인 개체를 확장 개체라 한다.
2. 첨부 연산은 모든 개체(닷 개체, 첨부 개체 또는 확장 개체)에 적용 가능하며 하나의 데크 내에서 첨부 연산의 회수에는 제한이 없다.
3. 확장 연산은 닷 개체 또는 닷 개체에 직·간접적으로 첨부된 개체로부터 확장할 때 사용된다. 하나의 데크 내에는 한번의 확장을 통해 도달할 수 있는 개체들만이 포함될 수 있다.

WAPSiteGen은 그림 3과 같이 두 가지 방식으로 WAP 응용을 생성할 수 있다. 그림 3의 (a)는 WAP 응용을 위한 클래스 다이어그램으로부터 정보를 추출해 응용 데이터베이스를 새로 구축하고 이 데이터베이스에 대한 비즈니스 로직을 처리하는 EJB 컴포넌트들과 WML 데크, 그리고 이 데크를 구성하는 카드들 통해 이뤄질 질의를 처리하는 JSP 컴포넌트들을 생성하는 첫 번째 방식을 나타낸다. 그림 3의 (b)는 기존 데이터베이스의 시스템 카탈로그로부터 스키마 정보를 추출해 이 데이터베이스에 대한 비즈니스 로직을 처리하는 EJB 컴포넌트들과 WML 데크, 그리고 이 데크를 구성하는 카드들 통해 이뤄질 질의를 처리하는 JSP 컴포넌트들을 생성하는 방식을 나타낸다.

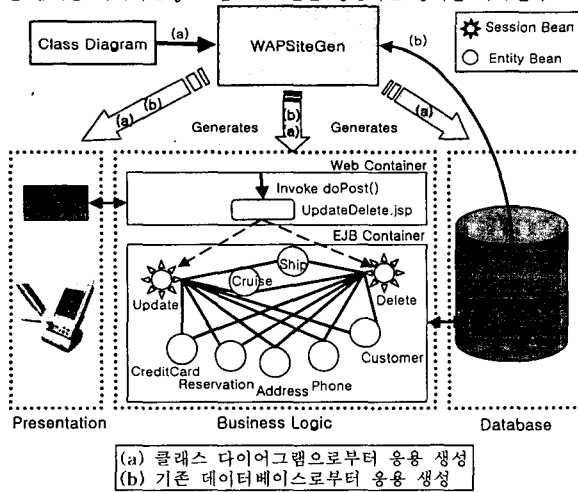


그림 3. WAPSiteGen

WAPSiteGen을 통해 생성된 WAP 응용은 3-계층 구조를 갖는다. 그림 1의 클래스 다이어그램으로부터 CUSTOMER, RESERVATION 등의 테이블이 생성되어 데이터베이스가 구축된다. 비즈니스 로직에는 엔티티 빈인 Customer, CreditCard 등과 세션 빈인 Update, Delete 등, 그리고 JSP 웹 컴포넌트들이 생성된다. 또한, WAPSiteGen으로부터 생성된 WML 데크들은 프리젠테이션 로직을 담당하게 된다. 사용자가 WML 데크를 구성하는 카드들 통해 질의를 하면 JSP 컴포넌트가 이를 처리하기 위해 해당 질의를 처리할 세션 빈을 호출한다. 이때 세션 빈이 데이터베이스에 접근하기 위해 엔티티 빈들을 사용한다. 생성된 EJB 및 JSP 컴포넌트들은 다른 응용에 재사용 될 수 있는 컴포넌트들이다.

2.1 클래스 다이어그램으로부터 응용의 생성

WAPSiteGen은 그림 1의 클래스 다이어그램으로부터 정보를 얻어 카달로그와 참조 무결성을 고려한 데이터베이스를 구축한다. 한 개체가 닷 개체로 선택되면 연관된 첨부 및 확장 개체들을 트리 형태로 TreeDlg 창에 보여준다. 트리에서 개체를 선택하면 선택된 개체의 속성들을 ColumnDlg 창에 보여준다. 사용자는 ColumnDlg 창에서 데크에 포함할 속성들을 선택할 수 있다.

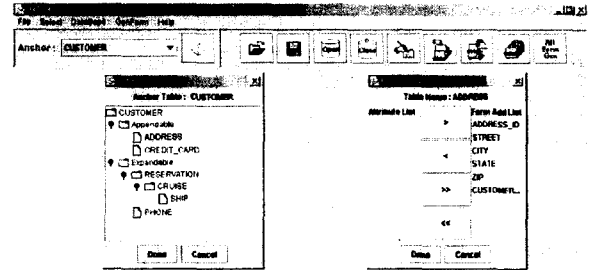


그림 4. CUSTOMER를 닷 개체로 선택 시 TreeDlg 창과 ColumnDlg 창

그림 4는 Titan 응용에서 CUSTOMER가 닷 개체 타입으로 선택된 경우이다. 닷 개체 타입으로 선택된 CUSTOMER와 CUSTOMER에 대해 가능한 모든 첨부 및 확장 연산에 의해 얻어진 개체 타입들인 ADDRESS, CREDIT_CARD, PHONE, RESERVATION, CRUISE, 그리고 SHIP이 모두 TreeDlg 창에 보여 진다. 트리에서 ADDRESS 개체 타입을 선택하면 속성들이 ColumnDlg 창에 보인다. 이렇게 선택된 개체 타입과 속성들은 한 단위로 검색된다.

2.2 기존 데이터베이스로부터 응용의 생성

기존의 데이터베이스로부터 WAP 응용을 작성하기 위해서는 기존 데이터베이스의 시스템 카탈로그로부터 스키마 정보를 SQL문을 통해서 추출한다. WAPSiteGen은 ORACLE8i 서버와 MS-SQL 서버 7.0으로 구축된 기존 데이터베이스를 분석하여 WAP 응용을 생성할 수 있다. 본 논문에서는 ORACLE8i 서버의 경우를 예로 든다.

아래의 SQL문을 이용하여 시스템 카탈로그 테이블인 USER_TABLES로부터 데이터베이스 내의 모든 테이블 이름을 검색할 수 있다.

```
SELECT TABLE_NAME FROM USER_TABLES
```

시스템 카탈로그 테이블인 ALL_CONSTRAINTS는 테이블들에 대한 관계 정보를 포함한다. 각 테이블에 대한 제약사항 이름인 CONSTRAINT_NAME과 참조되는 테이블의 고유 CONSTRAINT_NAME인 R.CONSTRAINT_NAME 필드를 통해 각 개체의 정보를 추출할 수 있다. 아래의 SQL문을 이용하면 닷 개체인 CUSTOMER와 일대일 또는 다대일 관계로 연관된 첨부 개체들을 구할 수 있다.

```
SELECT B.TABLE_NAME
FROM ALL_CONSTRAINTS A, ALL_CONSTRAINTS B
WHERE A.TABLE_NAME = UPPER('CUSTOMER')
AND B.CONSTRAINT_NAME = A.R_CONSTRAINT_NAME;
```

또한, 다음의 SQL문을 이용하면 CUSTOMER와 일대다 관계인 확장 개체들을 구할 수 있다.

```
SELECT B.TABLE_NAME
FROM ALL_CONSTRAINTS A, ALL_CONSTRAINTS B
WHERE A.TABLE_NAME = UPPER('CUSTOMER')
AND A.CONSTRAINT_NAME =B.R_CONSTRAINT_NAME;
```

WAPSiteGen은 이렇게 추출된 첨부 및 확장 개체들을 역시 그림 4와 같이 트리 형식으로 보여준다.

2.3 데크의 생성

그림 5는 WAPSiteGen이 생성한 Titan 응용의 WML 데크들의 동작원리를 보인다. 생성된 WAP 응용을 실행하면 첫 화면에 데이터베이스 내의 모든 테이블 이름들이 브라우저에 보인다. 그 중 질의 대상 테이블로 CUSTOMER를 선택하면 CUSTOMER에 대한 정보를 입력하거나 검색하는 화면이 보인다. CUSTOMER_ID 입력 창에 4를 입력하고 스크롤바를 내려 select를 선택한 후 Done 버튼을 누르면 CUSTOMER_ID가 4인 개체(닷 개체)와 그룹화 규칙에 의해 닷 개체와 연관된 모든 개체들에 관한 정보가 한번의 질의로 하나의 데크에 담긴다. 이 데크는 무선 장비에 전

송 되어 사용자 인터페이스로 사용되며, 이 데크에 담긴 정보들은 그림 5에서 보는 바와 같이 사용자의 조작에 의해 수정 또는 삭제될 수 있다. 따라서 관심의 대상인 개체와 연관된 정보들을 한번의 질의로 신속하게 접근할 수 있다.

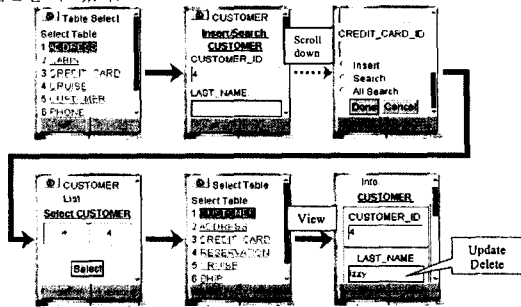


그림 5. WAPSiteGen이 생성한 Titan 응용의 WML 데크들의 동작원리

3. 상용 WAP 응용 생성기와의 기능 비교

본 절에서는 WAPSiteGen과 타 WAP 사이트 생성기의 기능을 비교한다. 표 1은 WAPSiteGen과 상용화된 WAP 사이트 생성 도구인 AnyBuilder Enterprise, Visual Studio .NET, MobileDev의 기능을 비교한 것이다.

표 1. 상용 WAP 응용 생성기와의 기능 비교

	WAPSiteGen	AnyBuilder Enterprise	Visual Studio .NET	MobileDev
프리티젠레이션	한 데크에 표현 가능한 개체	몇 개체 첨부 개체 확장 개체	몇 개체	몇 개체
	데크 생성	자동	반자동	반자동
비즈니스 로직	컴포넌트 사용유무	O	X	X
	생성코드	JSP와 EJB 컴포넌트	ASP, JSP, PHP	HTML, WML, cHTML
데이터베이스	데이터베이스 생성	클래스 다이어그램 이용	X	X
	기존 데이터베이스 사용	O	O	O
전체	생성된 응용의 구조	3-계층	2-계층	2-계층
	생성된 응용의 재사용성, 확장성, 이식성	+++	+	+

기존의 WAP 응용 자동 생성기인 AnyBuilder Enterprise, Visual Studio .NET, MobileDev는 데이터베이스를 새로 생성하지 못하고 기존의 데이터베이스만을 사용하는 무선 응용을 생성한다.

AnyBuilder Enterprise는 SQL문을 바탕으로 사용자 인터페이스로 사용될 WML, HDML, mHTML과 ASP, JSP, PHP 등의 질의 처리코드를 반자동으로 생성한다. Visual Studio .NET은 폼에 필요한 GUI 요소들을 선택, 구성하여 HTML, WML, cHTML의 코드를 생성한다. MobileDev는 Query Wizard를 사용하여 질의 처리를 하는 ASP, PERL 코드를 반자동으로 생성한다. 반면, WAPSiteGen은 데이터베이스에 대한 비즈니스 로직을 처리하는 EJB 컴포넌트들과 WML 데크, 그리고 이 데크를 구성하는 카드를 통해 이뤄질 질의를 처리하는 JSP 컴포넌트들을 자동으로 생성한다. 그리고 상용 생성기들은 하나의 데크에 몇 개체의 정보만을 담는데 반해 WAPSiteGen이 생성한 데크는 몇 개체뿐만 아니라 첨부 개체와 확장 개체의 정보까지 한번의 질의로 한 데크에 포함할 수 있다. 따라서 일대일, 다대일, 일대다로 연관된 개체들을 한번에 조작성할 수 있다는 장점이 있다. 또한, 상용 생성기들은 2-계층의 WAP 응용을 생성하는 데 반해 WAPSiteGen은 3-계층의 WAP 응용을 생성한다.

4. WAPSiteGen의 구현

본 절에서는 WAPSiteGen 프로토타입의 구현을 소개한다. 그림 6은 WAPSiteGen의 컴포넌트 계층구조이다.

SelectVersion 클래스는 데이터베이스를 새로 구축할 것인지, 기존의 데이터베이스를 사용할 것인지를 선택하게 한다.

새로 데이터베이스를 구축하는 경우, Entity, Relationship 클래스 등이 사용되어 클래스 다이어그램을 작성할 수 있는 환경을 제공한다. 데이터베이스는 Oracle8i 서버나 MSSQL 서버 7.0을 선택할 수 있고, 사용자의 선택에 따라 CreateDB 클래스가 테이블들을 생성한다. AnchorCombo 클래스는 그림 4와 같이 콤보박스에 테이블의 이름들을 나타낸다.

기존 데이터베이스의 사용을 선택하면, 사용자가 Oracle8i 서버나 MSSQL 서버 7.0을 선택하여 JDBC 또는 ODBC를 통해 Oracle8i 서버나 MSSQL 서버 7.0의 데이터베이스와 접근하여 기존 데이터베이스의 시스템 카탈로그로부터 스키마 정보를 추출하여 AnchorCombo 클래스의 콤보박스에 테이블들의 이름을 나타낸다.

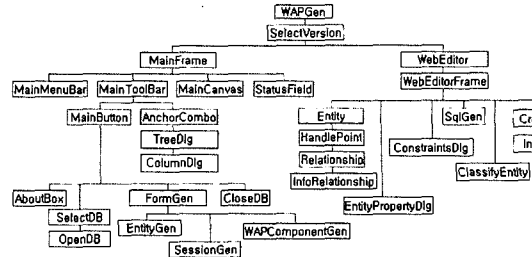


그림 6. WAPSiteGen의 컴포넌트 계층구조

AnchorCombo 클래스는 콤보박스에서 선택한 테이블명을 인식한다. TreeDlg 클래스는 선택된 테이블과 그 테이블과 첨부 및 확장 관계에 있는 테이블들을 트리 구조로 나타낸다. 트리 구조에 나타난 각 엔티티들은 ColumnDlg 클래스를 통해 데크에 표시될 테이블의 속성을 선택할 수 있다. EntityGen 클래스와 SessionGen 클래스는 EJB 컴포넌트를 생성하며, WAPComponentGen 클래스는 WAP 응용의 사용자 인터페이스로 사용되는 WML 데크들과 이 데크를 구성하는 카드를 통해 이뤄질 질의를 처리하는 JSP 웹 컴포넌트들을 자동 생성한다.

5. 결론 및 향후 연구

급증하는 모바일 전자 상거래의 수요에 맞게 신속한 서비스를 제공하기 위해 WAP 응용의 생산성 향상이 요구된다. 본 논문에서는, 데이터베이스를 사용하는 WAP 응용의 작성을 위해 WAP 응용의 사용자 인터페이스로 사용되는 WML 데크들, 그리고 이 데크를 구성하는 카드를 통해 이뤄질 질의를 처리하는 JSP 웹 컴포넌트 및 EJB 컴포넌트들을 자동 생성하여 WAP 응용의 생산성 향상을 지원하는 도구인 WAPSiteGen을 소개하였다. WAPSiteGen은 EJB와 JSP 등의 컴포넌트 기술을 도입함으로써 3-계층 구조를 갖는 WAP 응용의 확장성, 재사용성, 이식성 등을 지원한다. 또한, WAPSiteGen이 생성하는 WAP 응용은 상용 생성기가 작성하는 응용과 달리 현재 질의의 대상인 몇 개체는 물론 몇 개체와 일대일, 다대일 관계에 있는 첨부 개체와 일대다 또는 다대다 관계인 확장 개체 등을 한 단위로 검색한다. 이렇게 검색된 정보는 데크의 형태로 변환되고 무선 장비에 전송되어 무선 장비의 인터페이스로 제공된다. 따라서 관심의 대상인 개체와 연관된 정보들을 한번의 질의로 신속하게 접근할 수 있다.

향후, 본 논문에서 구현한 WAPSiteGen을 지리정보 응용에 적용할 것이다.

참고문헌

- [1] 이지훈, 한국 모바일 전자상거래 시장 내년 10억달러 돌파, <http://www.sewoo.net/news/news0410.htm>
- [2] 권오성, Converter : Integration between wired Internet and Wireless Internet, http://matilda.snu.ac.kr/doc/analysis/wap_converter.htm.
- [3] Wireless Markup Language, WAP Forum, Apr. 30. 1998. <http://www.wapforum.org>.
- [4] Doohun Eum and Toshimi Minoura, Web-Based Database Application Generator, *IEICE Trans. on Information & Systems, Vol. E86-D, No.6*, pp. 1001-1010, Jun. 20.
- [5] Visual Studio .NET에서 Mobile Web Forms를 사용하여 모바일 웹 응용 프로그램 만들기, <http://www.microsoft.com/korea/msdn/vstudio/techinfo/articles/clients/mobilewebforms.asp>.
- [6] AnyBuilder Enterprise, <http://www.anybil.com>.
- [7] MobileDev, <http://mobiledev.speedware.com>.