

하이브리드 모바일 애플리케이션 플랫폼, HyWAI®

한국전자통신연구원 | 이승윤 · 전종홍 · 이원석

1. 모바일 플랫폼 환경의 변화

최근 들어 아이폰을 필두로 스마트폰의 급격한 확산은 모바일 플랫폼 시장에 큰 변화를 가져오고 있다. 아이폰의 등장과 함께 기존 스마트폰 플랫폼 시장의 강자였던 노키아의 심비안은 시장 점유율이 점차 떨어지고 있으며, 애플 아이폰의 급격한 성장과 함께 RIM, HTC 등이 약진하는 등 기존 시장에 큰 변화가 일어나고 있다.

최근 MS의 윈도우즈폰7[1], 삼성의 바다(BADA)[2] 등도 새롭게 합류하면서 전 세계 모바일 플랫폼 시장은 그야말로 춘추전국시대로 접어들고 있다. 이미 10여종이 넘는 모바일 플랫폼이 시장에서 치열하게 경쟁하고 있어 향후 시장판도의 변화가 주목되는 시점이다. 특히, 이러한 환경변화는 기존 무선인터넷시장에서의 기득권자인 이동통신사에게 큰 위협으로 다가오고 있으며, 너무 많은 플랫폼 또한 개발자에게 비즈니스 측면에서 큰 부담으로 다가오고 있는 것이 사실이다.

최근 가트너가 주목해야할 10대 모바일 기술 중 하나로 “플랫폼 중립적인 모바일 응용 개발도구”를 선정한 것도 그러한 환경 변화 속에서 필요한 하나의 중요한 비즈니스 전략임을 간접적으로 해석할 수 있다. 특히, 플랫폼 중립적인 모바일 플랫폼 구축을 위한 큰 변화 중 하나는 바로 웹 기반의 모바일 플랫폼을 지향한다는 것이며, 최근 OMTB BONDI[3], JIL[4], W3C DAP[5] 등이 대표적인 모바일 응용 개발환경을 위한 웹 기반 API 표준화 노력이라고 할 수 있다.

한편, 최근 이와 같은 모바일 플랫폼의 과열 경쟁 현상은 자연스럽게 이통사 간의 상호 공존을 위한 노력으로 이어지며, 공통의 플랫폼 환경 내지는 공통의 개발환경 구축을 통해 모바일 응용개발 비용과 시간을 절약할 수 있는 쪽으로 새로운 표준화 시도가 진행되고 있다. 특히, 최근 전 세계 이통사들이 협의하여 만들기로 한 WAC(Wholesale Application Community, 일명 슈퍼앱스토어)[6]는 급변하는 모바일 시장에서 주도권 회복을 위한 절박한 노력의 일환이라고 할 수 있다.

본고에서는 최근 웹 기반의 모바일 플랫폼에 대한 기술 및 표준화 현황과 함께 ETRI가 개발한 웹 기반 모바일 응용 플랫폼인 하이웨이(HyWAI, Hybrid Web Application Interface) 플랫폼 설명과 함께 미래 모바일 플랫폼의 발전 전망을 하고자 한다.

2. 모바일 웹 애플리케이션 기술

일반적으로 “웹 애플리케이션”이라는 용어는 HTTP를 통해 전달되는 웹 페이지(XHTML 또는 그 변이형과 CSS, ECMAScript로 구성되는)의 집합체들이 웹 브라우저 내에서 애플리케이션 같은 환경을 제공하는 것을 말한다. 즉, 웹 애플리케이션은 여러 페이지를 거치는 대화형 처리 절차를 가지며, 이를 위한 상태 유지와 데이터 유지를 필요로 한다는 점에서, 단순한 웹 콘텐츠와는 구분된다. 이 중에서도 “협의를 웹 애플리케이션”은 브라우저 상에서 동작되는 형태만을 고려하지만, “광의의 웹 애플리케이션”은 HTTP, HTML/xHTML, URI를 필수 동작 요소로 가지며, 브라우저뿐 아니라 독립 애플리케이션으로 동작하는 것도 포괄하는 개념으로 사용된다.

2.1 하이브리드 응용 기술

모바일 애플리케이션은 전통적인 다운로드 VM(Virtual Machine) 기반의 네이티브 응용(Native Application, 보

표 1 네이티브(native) 응용과 웹 응용, 하이브리드 응용 비교

| | Native Application | Web Application | Hybrid App. |
|------------------------|--------------------|-----------------|-------------|
| Graphic Performance | 상 | 하 | 상 |
| AppStore 판매 (Monetize) | 가능 | 불가능 | 가능 |
| Offline Mode | 가능 | 일부 가능 | 가능 |
| 웹서비스 매쉬업 | 불가능 | 가능 | 가능 |
| Multi-platform 지원 | 어려움 | 용이 | 중간 |
| Storage | Local | 서버, cloud | 모두 |
| Device Capability 이용 | 용이 | 불가능 (개선중) | 용이 |
| 다중 사용자 공동 작업 | 불가능 | 가능 | 가능 |
| SW 경신 방법 | 재설치 | 사용중 수정 | 부분 재설치 |
| 애플리케이션 재활용성 | 소스/Lib 활용만 | 소스 및 SaaS로 | 모두 |
| UI 재작 난이도 | 상 | 하 | 중 |
| UI 표현 능력 | 상 | 하 | 중 |

통 앱(App)으로 불리움)과 더불어 웹 브라우저 기반의 웹 애플리케이션(Web Application, 보통 웹(Web)으로 불리움)으로 구분된다. 일반적으로 네이티브 애플리케이션은 빠른 속도를 제공하고 단말의 기능들을 효과적으로 활용할 수 있다는 장점을 갖는 반면, 많은 단말을 지원해야 할 경우 각각 별도로 개발을 해야 한다는 문제점과 함께 애플리케이션의 재활용과 업그레이드 등이 용이하지 않다는 단점을 갖고 있다. 반면 웹 애플리케이션의 경우, 별도 설치 없이도 계속 업그레이드된 기능을 사용할 수 있고, Open API 등을 통해 손쉽게 매쉬업(mash-up)할 수 있도록 기능을 제공하는 등 재활용을 할 수 있다는 장점을 갖는 반면, 오프라인 처리와 단말의 특성 정보를 활용할 수 없고, 브라우저의 성능에 좌우되며 대용량의 처리 등에 한계를 갖는다는 단점을 갖고 있다(표 1 참조).

따라서, 위와 같은 두 가지 방식의 애플리케이션의 장점을 가질 수 있도록 하며, 보다 빠르고 손쉽게 애플리케이션을 개발할 수 있도록 하기 위해, 네이티브 애플리케이션과 웹 애플리케이션을 합성한 하이브리드형 애플리케이션(Hybrid Application)들이 등장하고 있다. 하이브리드 애플리케이션은 HTML, CSS, JavaScript 등의 웹 표준을 사용하여 프로그래밍을 하면서, 독립된 네이티브 애플리케이션과 동일하게 다양한 단말 하드웨어 자원(카메라, 주소록, GPS 센서 등)들을 제어할 수 있다는 장점을 갖는다. 또한 이러한 하이브리드 애플리케이션은 일반 네이티브 애플리케이션 처럼 동작함으로써 인터넷 접속이나 서버 파일 없이도 동작 가능하며, 앱스토어 등에 올려서 판매하는 형태도 가

능해진다. 이러한 장점 때문에 최근 애플 아이폰, 구글 안드로이드, 팜 WebOS 등에서는 좀더 빠르고 손쉽게 하이브리드형 애플리케이션을 개발할 수 있도록 하는 다양한 웹 런타임(Web Runtime) 엔진들이 개발되고 있다. 써드파티 쪽에서도 여러 솔루션들이 나오고 있는데 대표적인 예가 PhoneGap[7], Titanium[8], Rhodes[9] 등이며, 이들 솔루션은 웹 런타임 기반으로 자바스크립트 등으로 구성된 공통 API를 통해 모바일 애플리케이션을 개발하고 추가 작업 없이 멀티 플랫폼에 대응하는 런타임 코드를 생성하는 방식을 제공한다.

2.2 디바이스 인터페이스 기술

웹 애플리케이션이 갖는 가장 큰 단점 중 하나는 네이티브 애플리케이션과 달리 단말의 하드웨어와 관련된 제어를 할 수 없다는 점이라 할 수 있다. 예를 들어 간단한 애플리케이션을 통해 배터리의 잔량, 주소록의 주소 정보, 단말에 저장된 일정 정보 등을 활용하고자 해도 할 수 없다는 점은 치명적인 약점으로 꼽혀왔다. 특히 이러한 웹 애플리케이션의 약점은 모바일 환경에서 더욱 치명적이라 할 수 있다. 데스크탑의 웹 애플리케이션과 달리 모바일 단말의 경우 플랫폼으로부터의 제약이 많이 갖고 있지만, 반면에 보다 다양하게 디바이스 기능들을 활용할 필요를 갖고 있어 단말 기능 접근에 대한 요구가 훨씬 크다고 할 수 있다.

이에 W3C(World Wide Web Consortium)에서는 2009년 6월 DAP(Device API and Policy) WG을 신설하였다. 이에 앞서 OAA(Open AJAX Alliance)의 Mobile Device API, OMTP BOND Activity, JIL을 통한 자발적 표준화

표 2 W3C DAP WG 개발 디바이스API 표준 대상 리스트

| API 이름 | 설명 |
|---------------------------|--|
| Calendar | 디바이스의 일정 정보에 접근하기 위한 API |
| Contact | 디바이스의 주소록 정보에 접근하기 위한 API |
| Media Capture | 디바이스 내 오디오, 이미지, 비디오 기능에 접근하기 위한 API |
| Messaging | 디바이스의 SMS, MMS, email 기능에 접근하기 위한 API |
| System Information | 디바이스의 기본적인 속성에 대한 API(배터리 용량, 네트워크 대역폭, CPU부하, 저장 용량, 입/출력 기기) |
| File Writer | 디바이스에 파일을 쓰기 위한 API |
| Gallery | 디바이스 내에 있는 미디어 갤러리(media gallery)에 접근하는 API. |
| Powerbox | 사용자 개인 리소스를 브라우저에서 요청하기 위한 웹 기반 전달 방식 |
| Application Launcher | 디바이스에 인스톨된 어플리케이션(Native Application)에 접근하기 위한 API |
| Tasks API | 디바이스 내에 저장된 업무(task) 관리(업무 추가, 삭제, 수정)를 위한 API |
| Application Configuration | 웹 어플리케이션의 설정(사용자의 선호도나 환경설정)을 변경하기 위한 API |
| User Interaction | 서로 다른 플랫폼에서도 사용자가 웹 어플리케이션을 더욱 수월하게 조작할 수 있도록 하는 API의 집합 |
| Communication Log | 디바이스 내 전화, 문자의 기록(log)의 내용물 및 메타데이터에 접근하기 위한 API |

노력이 있었기 때문에 W3C DAP 표준화 활동은 매우 자연스럽게 만들어졌다고 할 수 있다. 현재 W3C DAP WG은 1단계 표준화 활동을 2010년 말까지 마무리할 계획으로 있으며, 8개 이상의 핵심 API 문서와 요구사항 문서를 개발할 예정이다(표 2 참조).

이와 함께, 2010년 2월에 전 세계 이동사, 제조사들의 협의 하에 WAC(Wholesale Application Community)을 결성하여 모바일 애플리케이션 개발을 위한 개방형 인터페이스 표준화 작업에 착수하였다. WAC은 초기에 JIL과 BONDI 요구사항을 바탕으로 2010년 말까지 기본적인 규격을 만들 계획이며, 궁극적으로는 W3C DAP 표준을 수용한다는 계획을 가지고 있다. 실제로 OMTP는 현재 W3C DAP WG 표준화 작업에 참여를 통해서 BONDI 요구사항을 DAP 표준에 반영시키는 노력을 하고 있다. 그림 1은 현재 WAC에 참여중인 약 24개의 이동사, 제조사들을 보여주고 있으며, 한국에서는 SKT, KT, 삼성전자, LG전자 등이 참여 중이다.

3. 차세대 모바일 웹 플랫폼 요구사항

차세대 모바일 웹 플랫폼에서 고려하는 모바일 단말의 응용의 형태는 네이티브 응용과 브라우저 기반의 웹 응용으로 분류할 수 있다. 네이티브 응용은 모바일 플랫폼에서 제공하는 기본적인 플랫폼을 기반으로 개발된 독립적인 응용을 의미하며, 브라우저 기반의 웹 응용은 모바일 브라우저에서 실행되는 응용을 의미한다. 기본적으로 차세대 모바일 웹 플랫폼은 이 두 가지 환경을 모두 지원을 목표로 한다.

본 플랫폼이 목표로 하고 있는 세부 요구사항은 아래와 같다.

- 1) (콘텐츠 신디케이션 기능 및 최적화 기능 지원) 모바일 단말 사용자들은 새롭게 생성되는 다양한 인터넷 콘텐츠를 신속하게 전달하기 위해 데이터 신디케이션(Syndication) 기능과 콘텐츠 최적화 기능을 제공

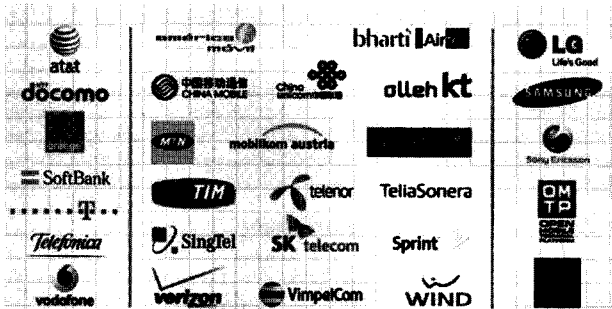


그림 1 WAC 참여 기관

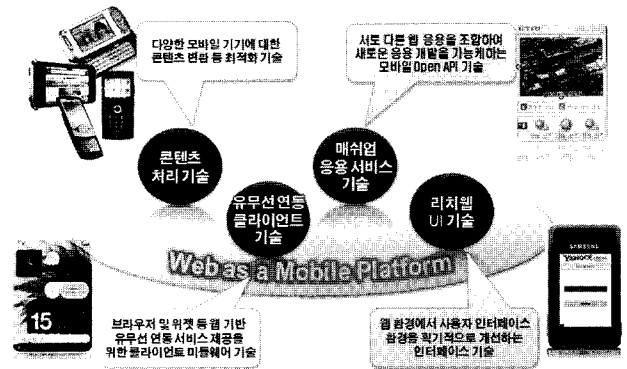


그림 2 차세대 모바일 웹 플랫폼 요구사항

- 2) (오픈API 지원) 네이티브 응용과 브라우저 기반 웹 응용은 기본적으로 기존에 인터넷에서 제공하는 다양한 개방형 API를 활용하여 쉽게 새로운 모바일 응용을 개발할 수 있는 환경을 제공
- 3) (디바이스 API 제공 및 오픈API 기반 매쉬업 지원) 최근에 기존에 개발된 다양한 기능들이 개방형 API 형태로 제공되면서 이를 적극적으로 활용한 매쉬업 응용이 점차 증가하고 있으므로 플랫폼은 이러한 개방형 API와 모바일 단말이 가지고 있는 GPS, RFID 리더 등 다양한 기능들 매쉬업하여 활용할 수 있는 기능을 제공
- 4) (리치웹 인터페이스 지원) 모바일 단말은 데스크탑 보다는 훨씬 제한된 자원을 포함하기 때문에 웹 환경에서의 사용자 인터페이스 기능이 획기적으로 개선될 필요가 있으며, 플랫폼은 이를 위한 기본적인 리치 웹 툴킷을 제공
- 5) (확장 가능한 플랫폼 기능 지원) 차세대 모바일 웹 플랫폼의 많은 부분들은 네이티브 응용과 브라우저 기반 웹 응용에서 공통적으로 활용이 가능하므로 향후 새로운 기능들이 추가될 때 추가된 기능이 공통적으로 활용될 수 있도록 추가하는 메커니즘을 제공

4. HyWAI 플랫폼

4.1 HyWAI 플랫폼 개념

하이브리드 웹 응용이란 하이브리드 자동차와 같이 서로 다른 기술들을 합성한 형태로, 보통 내장형 응용과 웹 응용 기술을 결합한 응용 형태를 의미한다. 즉, 내장형 응용처럼 독립적으로 실행될 수 있으며, 웹 응용처럼 웹 기술을 사용하여 화면 구성과 데이터 처리를 손쉽게 할 수 있는 장점을 지닌다. 또한 다양한 플랫폼에 효과적으로 적용할 수 있다는 장점을 가질 수 있다.

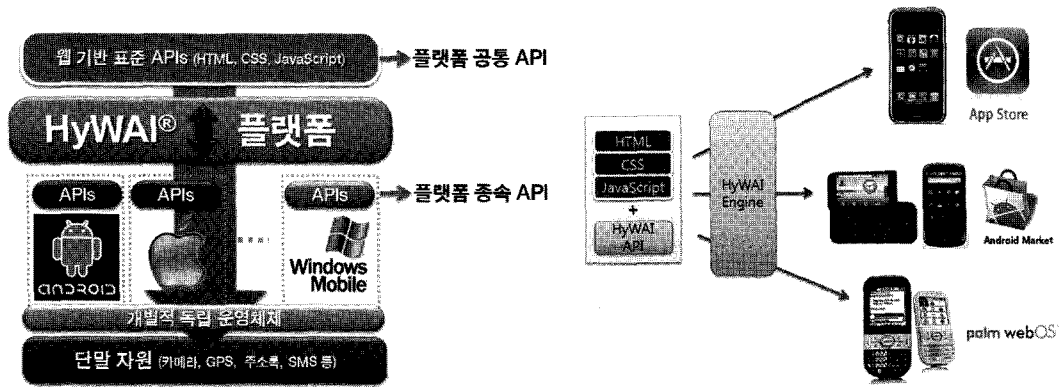


그림 3 하이웨이(HyWAI) 플랫폼 개념도

‘HyWAI(하이웨이, Hybrid Web Application Interface)’ 플랫폼의 첫 번째 장점은 웹 프로그래밍만으로도 모바일 단말의 고유 기능들을 직접 제어할 수 있는 하이브리드형 응용 프로그램을 손쉽게 개발할 수 있도록 한다는 점이다. 예를 들면 GPS를 통해 위치정보 값을 가져오거나, 카메라를 제어해 사진을 찍거나, 단말에 저장된 주소록과 페이스북, 트위터 등을 연계할 수 있도록 하는 것과 같은 인터넷과 연동되는 하이브리드형 웹 응용을 손쉽게 개발할 수 있다. 그림 3은 하이웨이 플랫폼의 개념도를 보여준다.

HyWAI 플랫폼의 두 번째 장점은 호환성과 이식성이다. 최근의 모바일 기기용 운영체제들은 각 사의 영역 확장을 위해 앞을 내다보기 힘들 정도로 치열한 다투를 벌이고 있으며 이로 인해 애플리케이션 개발사 및 개발자들은 하나의 서비스를 제공하기 위해서는 각 플랫폼에 최적화된 형태로 각각 개발해야 하는 점이 문제점을 안고 있다. HyWAI 플랫폼은 중립적인 웹 기술을 이용하여 이처럼 다양한 플랫폼에서 응용 프로그램 개발에 드는 시간과 비용을 최소화 할 수 있도록 하고 있다. 중립적인 웹 표준 기술인 Javascript, HTML, CSS를 사용하여 모바일 애플리케이션을 제작할 수 있도록 하고, Javascript 함수를 통해 디바이스 API를 호출 가능하게 함으로써 네이티브 애플리케이션과 동등한 기능을 제공하면서도 다양한 플랫폼에 효과적으로 이식할 수 있도록 한다.

4.2 HyWAI 기반 하이브리드 애플리케이션 개발을 위한 기술 요소

HyWAI에서의 하이브리드 애플리케이션은 각 플랫폼 별로 준비된 래퍼(wrapper)라는 것을 통해 실행되며, 기본적인 사용자 인터페이스의 구현과 처리는 HTML, CSS, Javascript를 이용하며, 단말의 기능 제어는 HyWAI가 제공하는 디바이스 API를 이용하여 이루어진다.

① CSS(cascading style sheet, 종속형 스타일 시트)
 웹 문서의 전반적인 스타일을 미리 저장해 두는 기술로 월드 와이드 웹 컨소시엄(W3C)에서 표준화한 하이퍼텍스트 생성 언어(HTML)용 스타일 시트. HTML을 이용해서 웹 페이지를 제작할 경우 전반적인 틀에서 세세한 글꼴을 일일이 지정해 주어야 하지만, 웹 페이지의 스타일을 미리 저장해 두면 웹 페이지의 한 가지 요소만 변경해도 관련되는 전체 페이지의 내용이 한꺼번에 변경되므로, 문서 전체의 일관성을 유지할 수 있고 작업 시간도 단축된다. 하나의 스타일을 정의하면 여러 개의 문서에서 사용할 수 있으며 수정이 용이하다.

② HTML(Hypertext markup language, 하이퍼텍스트 마크업 언어)

인터넷 웹 페이지의 하이퍼텍스트 문서를 만들기 위해 사용되는 기본 언어. 문서의 글자 크기, 글자 모양, 그래픽, 색, 문서 이동 등을 정의하는 언어로 구성되어 있으며, 태그(tag)라는 명령어로 구분하여 사용된다.

③ Javascript(자바 스크립트)

미국의 넷스케이프 커뮤니케이션즈사(Netscape Communications)가 개발한 스크립트 언어. 웹 브라우저에서 실행하는 스크립트 언어를 기술한다. 언어 규격은 자바의 부분 집합(subset)으로 되어 있다. 하이퍼텍스트 생성 언어(HTML) 문서를 작성하는 수준의 사용자가 사용하는 것을 주안점으로 하여 자바의 언어 규격으로부터 변수의 형(정수형이나 문자열형 등)을 생략하거나 새로운 클래스 정의를 할 수 없도록 하였다. 여러 회사에서 만든 다양한 자바 스크립트 언어간의 호환성 확보를 위해 ECMA를 통해 표준화된 ECMAScript 규격이 있다.

④ Device APIs

HyWAI API는 각각의 플랫폼별로 제공되는 다양한

표 3 HyWAI 지원 디바이스 API 리스트

| No. | 구분 | API 종류 | 설명 |
|-----|------------|------------|---------------------------|
| 1 | 디바이스 | 위치정보 | 현재 GPS 위치 좌표 읽기 |
| 2 | | 가속센서 | 움직임 변화 벡터의 x, y, z 값 읽기 |
| 3 | | 카메라 | 사진을 찍고 저장 |
| 4 | | 진동 | 디바이스 진동시키기 |
| 5 | | 방향전환 | 디바이스의 방향 변화 감지 |
| 6 | | 제스처/멀티터치 | 하나 이상의 손가락을 사용한 복잡한 입력 지원 |
| 7 | 네트워킹 | 네트워크 가능 여부 | 네트워크 접근이 가능한지 판단 |
| 8 | | 소켓 | 소켓 통신 기능 |
| 9 | | FTP | FTP 클라이언트 기능 |
| 10 | 사운드 | 사운드 재생 | 오디오 파일 재생 |
| 11 | 통화 관련 | 주소록 API | 주소록 애플리케이션 정보 접근 |
| 12 | | SMS API | SMS 지원 |
| 13 | | 전화 API | 전화 걸기 기능 지원 |
| 14 | | 로그 | 로그 관리(SMS, 착발신 정보 등) |
| 15 | DB 및 파일시스템 | SQLite 기능 | 데이터베이스 기능 |
| 16 | | 파일시스템 I/O | 텍스트 또는 바이너리 파일 읽고 쓰기 |
| 17 | | 저장 | 보안이 필요한 정보 저장 기능 |
| 18 | 애플리케이션 관리 | 애플리케이션 설정 | 애플리케이션 정보 관리 |
| 19 | | 애플리케이션 런처 | 애플리케이션 시작 및 관리 |
| 20 | 정보 관리 | 사용자 정보 | 사용자 Preference 설정 관리 |
| 21 | | 디바이스 속성 정보 | 디바이스 정보 관리 |

장치 제어와 핵심 기능 접근에 대한 디바이스 API들을 추상화하여 제공하는 것으로, 하이브리드 응용에서 HyWAI API를 통해 제공되는 JavaScript 함수들을 이용하여 다양한 단말 기능 처리들을 할 수 있도록 한다.

4.3 HyWAI 제공 API

현재 HyWAI 엔진을 통해 제공되는 주요 API들은 표 3과 같이 약 21종의 API를 제공하고 있으며, 이들 API들은 지속적으로 확장될 예정이다. HyWAI API는 궁극적으로 W3C DAP에서 만들어질 Device API와 호환 가능하도록 만들어질 예정이다.

5. HyWAI 기반 애플리케이션 구현 사례: TwitPhone

본 응용은 HyWAI 기반의 샘플 애플리케이션으로서, 스마트폰의 전화번호부와 위치정보를 그리고 인터넷을 통한 구글 맵과 트위터 기능을 조합하는 만드는 매쉬업 응용이다. HyWAI API를 이용하여 휴대폰 전화번호부에 있는 연락처 정보를 읽어 사용자들의 트위터 주소를 가져오고, 트위터의 오픈 API를 이용하여 트위터 메시지들을 가져오고, GPS로부터 받은 위치정보를 구글맵 API를 통해 주소 정보로 변환한 다음, 이를 포

합하는 트윗 메시지를 남길 수 있도록 하는 모바일 응용이다. 이 응용을 통해 사용되는 매쉬업 API 들은 다음과 같다.

- HyWAI API (Device API) : GPS, Contacts, Telephony APIs
- Open API : Google Map APIs, Twitter APIs

그림 4는 HyWAI 기반의 매쉬업 응용인 TwitPhone 응용의 개념도를 보여준다.

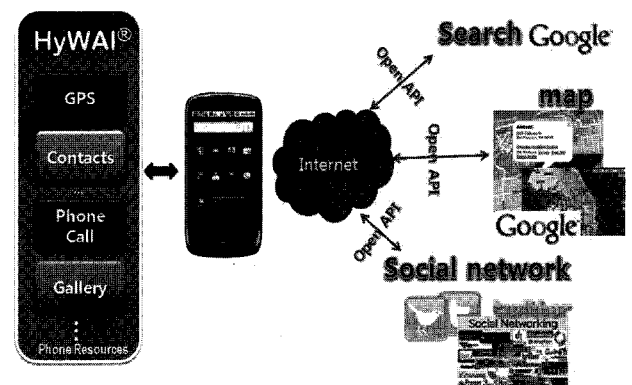


그림 4 하이웨이(HyWAI) 기반 매쉬업 응용, TwitPhone 개념도

```

1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
2 "http://www.w3.org/TR/html4/strict.dtd">
3 <html>
4 <head>
5 <meta name="viewport" content="width=320; user-scalable=no" />
6 <meta http-equiv="Content-type" content="text/html; charset=utf-8">
7 <meta name="generator" content="WebEditor">
8 <style>!--
9 .nmsw_over { background-color:rgb(0,102,255); }
10 -->
11 </style>
12 <title>HYWAI</title>
13 <link rel="stylesheet" href="master.css" type="text/css" media="screen" title="no title" charset="utf-8">
14 <script type="text/javascript" charset="utf-8" src="basic.js"></script>
15 <script type="text/javascript" charset="utf-8" src="hywai.js"></script>
16 <script type="text/javascript" charset="utf-8" src="TwitPhone.js"></script>
17 </head>
18 <body onload="init(); reload();" id="stage" class="theme">
19 <div class="boxTopR">
20 <div class="boxTop-left"></div>
21 <div class="boxTop-mid"></div>
22 <div class="boxTop-right"></div>
23 </div>
24 <div id="twitter_div">
25 <center></center>
26 <div id="magicTable"></div>
27 <br><div align="right">Powered by </div>
28 </div>
29 </body>
30 </html>

```

그림 5 하이웨이(HyWAI) 기반 응용, TwitPhone 소스코드

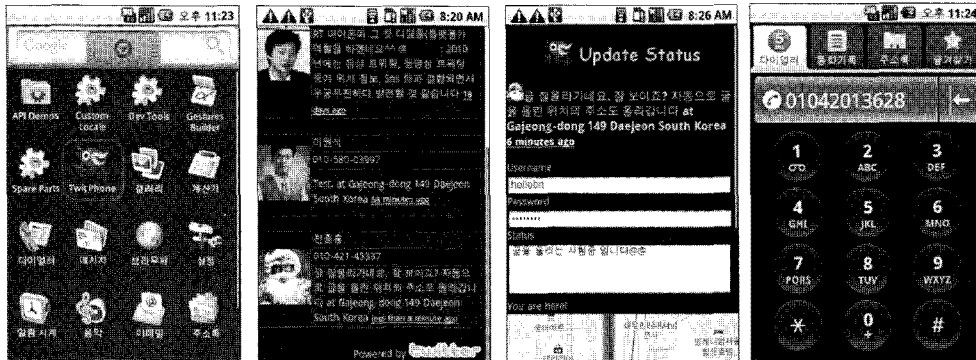


그림 6 하이웨이(HyWAI) 기반 응용, TwitPhone 동작 결과 화면

그림 5는 TwitPhone 응용 구현시 사용된 웹 기반 소스 코드의 일부이며, 그림 6은 TwitPhone의 실행 결과 예제이다.

6. 결론 및 향후 계획

최근 스마트폰과 앱스토어 경쟁에서 알 수 있듯이 오늘날 모바일 분야의 핵심 경쟁력은 애플리케이션과 콘텐츠 경쟁력에서 좌우되고 있다. 빠르게 발전하고 있는 모바일 시장에서 보다 신속하고 보다 쉽게 모바일 응용을 개발 할 수 있도록 하는 것은 중요해지고 있으며, 이러한 이유로 전 세계적으로 디바이스 API 표준과 웹 기반 하이브리드 앱 엔진 기술이 주목 받고 있다.

HyWAI 플랫폼은 점차 복잡해지는 모바일 플랫폼 환경과 개발 환경 속에서 웹 기술만으로 하이브리드형 애플리케이션을 손쉽게 작성할 수 있도록 하며, 다양

한 플랫폼에 소스 호환성과 이식성을 갖고 적용할 수 있도록 한다는 장점을 갖는다. 아이폰과 안드로이드의 경쟁과 스마트폰의 물결에 휩싸이고 있는 국내 환경에서 이러한 모바일 응용 개발 환경은 특히 중요한 의미를 갖는다고 할 수 있다. 아이폰과 안드로이드 플랫폼과 앱스토어가 경쟁하는 시점에서 하이웨이(HyWAI) 플랫폼은 개발자들이 보다 손쉽게 다양한 스마트폰 응용들을 손쉽게 개발할 수 있도록 도와줄 것으로 기대 된다.

이번의 개발된 HyWAI 플랫폼은 안드로이드 플랫폼을 중심으로 핵심 API들을 구현하였고, iPhone 플랫폼에서는 일부 API들만을 구현하였다. 향후에는 이번에 개발된 위치정보, 가속센서, 카메라, 연락처, SMS, 제스처 등 21개의 핵심 API이외에도, RFID, 블루투스, XMPP 등과 같은 다양한 기능들을 계속 추가할 예정이다.

또한, HyWAI 플랫폼은 W3C의 Device API 표준과 완벽한 호환성을 가질 수 있도록 확장할 예정으로 있으며, 안드로이드 외에도 리모(LiMo) 등의 다양한 모바일 플랫폼을 지원할 수 있도록 지속적인 확장을 추진할 계획이다.

참고문헌

- [1] Windows Phone 7, <http://www.windowsphone7.com/>
- [2] Samsung Bada, <http://www.bada.com/>
- [3] OMTP BONDI, <http://bondi.omtp.org/>
- [4] JIL, <http://www.jil.org/>
- [5] W3C DAP, <http://www.w3.org/2009/dap/>
- [6] WAC, <http://www.wholesaleappcommunity.com/>
- [7] phoneGap, <http://www.phonegap.com/>
- [8] Titanium, <http://www.appcelerator.com/>
- [9] PhoneGap, <http://phonegap.com/>
- [10] Rhodes, <http://rhomobile.com/>
- [11] 모바일웹2.0포럼, <http://www.mobileok.kr> (or) <http://www.mw2.or.kr>

약력



이승윤

1999 ETRI 표준연구센터 입사(선임연구원)
 2003~현재 ETRI 표준연구센터 서비스융합표준연구팀장(책임연구원)
 2004~현재 TTA 국제표준전문가
 2006~현재 TTA 웹프로젝트 그룹(PG605)의장
 2005~현재 ASTAP IRT EG 라포쳐

2006~현재 ITU-T SG13 Editor
 2008~현재 W3C 대한민국사무국 사무국장
 2009~현재 ISO/IEC JTC 1 SC38 SGCC 컨버너
 현재 한국전자통신연구원 표준연구센터 서비스융합표준연구팀 팀장
 관심분야: 차세대 웹, 모바일 웹, 유비쿼터스 웹, 클라우드 컴퓨팅, IPTV, 미래인터넷, e-Book 표준 등
 E-mail : syl@etri.re.kr
 Twitter : @seungyun



전종홍

1996~1999 한국정보시스템 기술개발연구소 주임연구원
 1999~현재 ETRI 표준연구센터 선임연구원
 2004~현재 TTA 웹프로젝트 그룹(PG605) 부의장
 2008~현재 TTA 모바일 웹 실무반(WG6051)의장
 2009~현재 모바일 웹 2.0 포럼 One Widget AG 의장

2006~현재 TTA 국제표준전문가
 현재 한국전자통신연구원 표준연구센터 서비스융합표준연구팀 선임연구원
 관심분야: 모바일 웹, 웹2.0 응용, 유비쿼터스 웹, 소셜 웹, 웹 기술 표준화
 E-mail : hollobit@etri.re.kr
 Twitter : @hollobit



이원석

2003~현재 ETRI 표준연구센터, 선임연구원
 2005~현재 W3C 대한민국 사무국 코디네이터
 2006~현재 TTA 국제표준전문가
 2006~2008 ITU-T SG13 에디터
 2008~현재 W3C Media Annotation WG 에디터
 2009~현재 ISO/IEC JTC1 WC29 WG11 에디터

2009~현재 W3C Device APIs and Policy WG 에디터
 2010~현재 모바일 웹 2.0 포럼 HTML5 AG 의장
 현재 한국전자통신연구원 표준연구센터 서비스융합표준연구팀 선임연구원
 관심분야: 차세대 웹 표준, 유비쿼터스 웹, 모바일 웹, 시맨틱 웹, HTML5, 웹 플랫폼
 E-mail : wslee@etri.re.kr
 Twitter : @wonsuk73