

가전 제품과 정보 기기를 위한 Java 기술

시스템공학연구소 박성우·손종문·김태근*

1. 서 론

초기의 Java 기술은 Java 애플릿(applet)으로 구현된 애니메이션, 인라인 사운드, 실시간 상호 작용 등을 이용하여 인터넷 서비스를 위한 GUI(Graphical User Interface)로 사용되었다. Java는 객체 지향, 분산환경 지원, 인터프리터 방식, 플랫폼 독립성, 멀티쓰레딩 지원 등의 장점들을 가지고 있기 때문에 최근에는 전자상거래, 전자출판, 멀티미디어 엔터테인먼트 등과 같이 보다 복잡한 응용에 적용되어 사실상의 인터넷 표준 언어로서의 역할을 하고 있다.

Java는 원래 고객주문형 가전제품의 대화식 제어를 위한 연구과정에서 개발된 언어이다. 1990년 Java 개발의 중심적인 인물인 제임스 고슬링(James Gosling)은 C++ 또는 그와 유사한 언어를 사용하여 가전제품을 위한 내장형(embedded) 소프트웨어를 개발하려고 하였다. 내장형 소프트웨어는 마이크로프로세서를 탑재한 전자제품을 제어하기 위한 소프트웨어이다. 그러나 가전제품을 위한 내장형 소프트웨어 개발에 범용언어가 적합하지 않음을 발견하고 Java 언어를 개발하였다[1]. 최근에 썬(Sun Microsystems)은 Java 기술을 이용하여 인터넷/인트라넷용 소프트웨어 뿐만 아니라 휴대폰, 프린터, 가전제품 등과 같은 내장형 소프트웨어 시장도 점유하려 하고 있다[2].

Java를 이용한 내장형 소프트웨어는 Java 수행 환경을 내장형 디바이스(device)에 탑재함으로써 실행 가능하다. 이러한 방식은 내장

형 소프트웨어 개발에서 발생하는 전통적인 문제-에러 없는 소프트웨어의 개발, 소프트웨어의 재사용, 실시간 처리의 지원, 소프트웨어의 업그레이드, 개발기간 등-를 Java의 장점을 활용하여 해결하고 또한 인터넷 기술에 기초한 전자제품(internet appliance)을 위한 내장형 소프트웨어를 손쉽게 개발하기 위한 것이다.

본 고에서는 가전제품 및 이동형 정보 기기를 위한 내장형 소프트웨어 개발기술과 동향에 대하여 기술한다. 2장에서는 내장형 소프트웨어 개발의 전통적인 문제점을 살펴보고, Java의 어떠한 특징이 이러한 문제점을 해결하는지 소개한다. 3장에서는 썬의 EmbeddedJava와 PersonalJava에 대하여 기술하고 4장에서 내장형 시스템들의 개발 및 상용화 사례를 살펴본다. 마지막으로 5장에서 결론을 기술한다.

2. 내장형 소프트웨어를 위한 Java

기술적, 경제적인 측면에서 내장형 소프트웨어에 대한 요구사항은 신뢰성, 저가, 소프트웨어 재사용, 소규모 메모리, 실시간 지원, 소프트웨어 업그레이드 등으로 나눌 수 있으며, 이러한 요구사항을 만족해야 하는 이유는 다음과 같다.

-신뢰성(reliability): 오류 발생시 수정하기가 어려울 뿐만 아니라 반품의 결정적인 이유가 될 수 있다.

-저가(low cost): 가전 제품 시장은 가격에 민감하다. 즉, 가격 문제로 하드웨어 플랫폼이 바뀔 수 있기 때문에 다양한 플랫폼을 지원할 수 있고 소프트웨어의 작성성이 손쉽고 간단해야

*정회원

표 1 내장형 소프트웨어를 위한 Java의 장점

Java의 장점	설 명	내장형 소프트웨어 개발 관련 특성
이식성 및 다양한 플랫폼(portable and multi-platform)	가상머신을 기반으로 동작하기 때문에 이식성이 높아 다양한 플랫폼을 지원할 수 있다.	저가, 소프트웨어 재사용 및 업그레이드
견고성(robustness)	Java는 메모리를 직접 참조하는 프리미티브를 제공하지 않으며, 메모리 관리는 가상머신의 가비지 컬렉터(Garbage Collector)에 의해 수행된다.	신뢰성
높은 보안성(high security)	가상머신의 바이트코드 검증기(Byte Code Verifier)가 수행이전에 보안상의 문제점을 검사하고, 잠재적인 보안의 허점을 막기 위하여 애플릿의 수행을 제한한다.	신뢰성
객체 지향(object-oriented)	Java는 객체 지향 언어이다.	저가, 소프트웨어 재사용 및 업그레이드

한다.

-소프트웨어 재사용(software reuse) : 시장 선점을 하기 위해서는 소프트웨어의 개발 기간이 짧아야 하기 때문에 기존의 소프트웨어를 재사용 할 수 있어야 한다.

-소규모 메모리(small memory footprint) : 내장형 디바이스는 하드웨어 가격을 낮추기 위하여 수백 KB에서 수 MB정도의 메모리만을 갖는다. 그러므로 RAM 사용량이 최소화되는 소프트웨어를 개발하여야 한다.

-실시간 지원(real-time support) : 내장형 디바이스는 이벤트(event)에 신속히 응답하여야 한다.

-소프트웨어 업그레이드(software upgrade) : 내장형 디바이스는 오랜 기간 사용되며 유사 기능을 가진 새로운 디바이스가 생산될 수 있기 때문에 기존 내장형 소프트웨어에 대한 호환성이 있어야 하며 기능 확장이 쉬워야 한다.

이미 기술한 바와 같이 Java는 대화형 가전 제품의 내장형 소프트웨어를 위한 언어로 개발되었기 때문에 위에서 기술한 내장형 소프트웨어의 요구사항을 만족시킬 수 있다. Java를 이용하여 내장형 소프트웨어를 개발할 때 얻을 수 있는 장점은 표 1과 같이 요약될 수 있다. 이러한 장점 외에도 하드웨어가 준비되기 이전에 어떠한 플랫폼에서도 소프트웨어를 개발 및 시험할 수 있으므로 소프트웨어와 하드웨어의

동시 개발로 인한 개발 기간의 단축 및 보다 안정된 소프트웨어 확보가 가능해진다.

3. EmbeddedJava와 PersonalJava

썬은 JAE(Java Application Environment)에 대한 기본적인 방향을 응용 분야와 규모에 따라 EnterpriseJava, PersonalJava, EmbeddedJava, CardJava로 구분하고 있다. 각 JAE는 상위 JAE가 가진 API(Application Program Interface)의 부분 집합과 고유의 API로 구성된다. 예를 들어 PersonalJava는 EnterpriseJava가 가지는 API의 부분 집합과 PersonalJava 고유의 API를 가진다. 본 장에서는 4개의 JAE중 EmbeddedJava와 PersonalJava의 특징과 구성을 기술하고 EnterpriseJava와 비교한다.

3.1 EmbeddedJava와 PersonalJava의 구성

이미 기술한 바와 같이 소규모의 내장형 시스템 상에서 자바의 장점을 활용하기 위해서는 Java 가상머신인 내장형 실시간 운영체제 상에서 운용되도록 개발되어야 한다. 이를 위해서 썬은 가전제품을 비롯한 기존의 전자 제품과 인터넷 기술을 채용한 정보 기기를 위한 Java 플랫폼인 EmbeddedJava와 PersonalJava를 제공하고 있다. EmbeddedJava와 Personal-

표 2 EmbeddedJava와 PersonalJava의 응용 분야 및 하드웨어 플랫폼

구분	EmbeddedJava	PersonalJava
응용 분야	모바일 폰, 페이지, 공장 자동화 장비, 네트워크 라우터, 프린터 등과 같이 제한된 사용자 인터페이스를 가지는 저전력 또는 이동형 디바이스	웹브라우저 폰, PDA, 셋톱박스(set-top box) 등과 같이 네트워킹 기능을 가지며 웹상에서 애플릿에 대한 호환성을 요구하지만 완전한 EnterpriseJava 서비스를 요구하지 않고 하드웨어적인 제약을 갖는 디바이스
하드웨어 플랫폼	25MHz 이상의 16/32bit 프로세서 256KB~512KB ROM 256KB~512KB RAM	50MHz 이상의 32bit 프로세서 약 2MB ROM 512KB~1MB RAM

표 3 EmbeddedJava와 PersonalJava 설계시 고려사항과 특징

EmbeddedJava	PersonalJava
실시간 운영체제 상에서 수행된다.	실시간 운영체제 상에서 수행된다.
범용 애플릿을 수행하지 않고 디바이스의 기능을 확장한 Java 애플리케이션을 수행한다.	범용 애플릿을 수행한다.
최소 메모리 사용, 저가, 기능에 대한 제한을 고려한다.	EnterpriseJava보다 적은 메모리와 성능이 낮은 프로세서를 요구한다.
사용자 인터페이스가 제한되어 있다.	해상도가 낮은 디스플레이를 사용하고 대부분 키보드와 마우스는 없이 단순한 사용자 인터페이스를 요구한다.
새로운 기능을 쉽게 추가할 수 있다.	새로운 기능을 쉽게 추가할 수 있다.
실시간 운영체제와 관련 도구의 성능에 의존적이다.	PersonalJava를 사용함으로써 목표 프로세서를 쉽게 바꿀 수 있다.
EmbeddedJava를 채용한 제품에는 Java Powered 로고를 사용한다.	PersonalJava를 채용한 제품에는 Java Compatible 로고를 사용한다.

Java의 개발로 인하여 기존의 데스크탑 환경에서 운영되던 Java는 EnterpriseJava로 이름을 바꾸었다.

표 2는 EmbeddedJava와 PersonalJava의 응용 분야와 하드웨어 플랫폼의 개발기술을 기술한 것이며 표 3은 그 특성을 기술한 것이다. EmbeddedJava는 주로 기존의 내장형 시스템을 위한 소프트웨어에 적합하게 Java 수행 환경을 갖춘 것이다. 또 다른 관점에서는 EnterpriseJava를 축소하여 네트워크 기능을 가지지 않고 전자 제품을 위한 API를 추가한 JAE라고도 할 수 있다. 반면 PersonalJava는 인터넷 환경 및 이동형 통신기기의 범용화로 인하여 네트워킹 기능을 필요로 하는 전자제품을 위한 JAE이다. PersonalJava는 크기나 기능 면에서 EmbeddedJava와 EnterpriseJava의 중간쯤에 위치한다고 할 수 있다.

PersonalJava는 내장형 장비에서 수행되는 네트워크 응용프로그램을 위한 JAE이다. Pe-

sonalJava는 데스크탑 컴퓨터보다는 셋톱박스나 스마트폰 같은 플랫폼을 대상으로 하고 있기 때문에 PersonalJava API의 규모는 JDK1.1 API보다 훨씬 작아야 한다. 따라서 PersonalJava는 최소의 메모리를 사용하면서도 고도로 확장성을 지니게 된다. PersonalJava는 다음과 같은 사항을 만족하도록 설계되어 있다.

- PersonalJava 사양에 따라 작성된 애플릿은 JDK1.1 환경에서 수행되어야 한다.

- PersonalJava 애플릿은 PersonalJava를 지원하는 JDK1.1 패키지와 링크시 호환되어야 한다.

- PersonalJava를 기반으로 하는 제품은 컴퓨터를 사용한 경험이 없는 사람에게도 사용하기 쉬워야 한다.

- PersonalJava는 리모콘, 터치스크린 같은 가전제품의 다양한 입출력 포맷을 다룰 수 있는 유연성을 제공해야 한다.

- PersonalJava는 JDK1.1보다 적은 메모리 양을 요구해야 한다.

- PersonalJava는 JDK1.0.2와 JDK1.1을 위하여 작성된 애플릿의 대부분을 수행할 수 있어야 한다.

현재 PersonalJava 1.0 API는 JDK1.1 API의 부분집합으로 네트워킹 기능을 가진 내장형 소프트웨어를 위한 새로운 API를 포함하고 있다. 썬은 EmbeddedJava와 PersonalJava를 위한 개발 도구로 Romizer 2, 자원 예측기, Configuration Tool, Lint와 유사한 도구, 컴파일러 등을 제공하는데 디바이스 개발자와 응용 프로그램 개발자를 위해서는 다음과 같은 개발 도구를 제공한다.

- JavaConfig : 그래픽 사용자 인터페이스를 사용하며 클래스 선택기(class selector)와 클래스 의존성 분석기(class dependency analyzer)를 포함하며 선택된 클래스들에 대한 메모리 요구량을 예측한다.

- JavaCheck : 클래스 파일이 PersonalJava, EmbeddedJava JAE 사양에 부합하는지 검사한다.

- JavaCodeCompact : 여러 개의 클래스 파일을 압축하여 하나의 다중 클래스 파일로 만들어 주며, 어셈블러와 함께 플랫폼에 독립적인 클래스 파일을 목표 시스템에 의존적인 파일로 변환한다.

- JavaDataCompact : HTML, image, sound와 같은 데이터를 다중 클래스 파일로 입력하여 어셈블러와 함께 목표 시스템에 의존적인 파일 형식으로 변환한다.

EmbeddedJava에 대한 사양은 현재 발표되어 있지 않아 정확한 규격은 알 수 없는 실정이다.

3.2 EmbeddedJava, PersonalJava 및 EnterpriseJava 비교

이미 기술한 바와 같이 EmbeddedJava, PersonalJava, EnterpriseJava 등의 JAE는 상위 JAE가 가지는 API의 부분집합과 고유의 API로 구성된다. 현재 많은 Java API가 발표되어 있지만 각 API는 JAE를 기본적으로 구성하는 것, JavaSoft가 옵션으로 제공하여 기

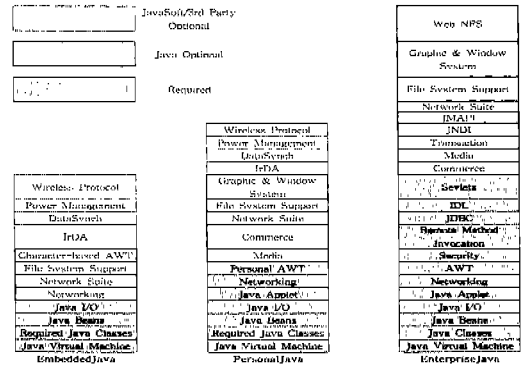


그림 1 JAE 구성 요소 비교

본 구성에 포함할 수 있는 것, JavaSoft 또는 third party에 의해 옵션으로 제공되어 기본 구성에 포함할 수 있는 것 등의 세 가지로 분류된다. 그림 1은 EmbeddedJava, PersonalJava, EnterpriseJava의 구성 요소를 보여주고 있다.

EmbeddedJava는 Java 애플릿을 지원하기 위한 구성 요소를 가지지 않으며 Java 애플리케이션만을 수행할 수 있다. 그리고 모빌폰, 페이지 등과 같은 디바이스의 LCD 패널에 문자를 표시하기 위하여 다른 JAE에는 없는 문자 기반 AWT를 가진다. PersonalJava는 EmbeddedJava와는 달리 Java 애플릿을 가지고 있어 웹브라우저 폰과 같이 제약된 하드웨어 환경이지만 애플릿에 대한 호환성이 요구되는 환경에서도 사용될 수 있다.

4. 자바기술을 적용한 내장형 시스템 개발 사례

미쓰비시와 UCSC(University of California, Santa Cruz)에서는 썬의 EmbeddedJava/PersonalJava 기술이 구체화되기 전에 이미 Java 기술을 적용하여 내장형 시스템을 개발하였다. 또한 최근에 Acorn, Chorus, GeoWorks, Lucent Technologies, Microtec, Microware [7], QNX, WindRiver[5] 등의 실시간 운영체제 업체들이 썬의 EmbeddedJava와 PersonalJava를 라이선스함으로써 자사의 실시간 플랫폼에 통합하여 판매할 수 있게 되었다. EmbeddedJava와 PersonalJava를 라이선스한 업체

들의 시장 점유율이 전체 상용 실시간 운영체제 시장의 50% 이상을 차지하고 있음을 볼 때 향후에는 Java 기술이 내장형 시스템 시장을 석권할 것으로 보인다. 실시간 운영체제를 사용하는 회사에 대한 설문 조사 결과에 의하면 WindRiver사의 VxWorks, Enea OSE System사의 OSE, Microware사의 OS-9, Lynx Real-Time Systems사의 LynxOS, Microtec사의 VRTX 등의 실시간 운영체제들이 전체 시장의 25%, 18%, 16%, 15%, 12% 정도를 차지하고 있는 것으로 나타났다[4]. 주목할 만한 것은 썬이 1997년 상반기에 인수한 정보 기기 개발 업체인 Diba가 Microware사와 협력관계를 맺고 있고 1997년 9월에 Chorus사도 인수함으로써 썬도 내장형 실시간 시스템 시장에 직접 진출하고 있다는 점이다. 통신 회사인 Alcatel, Nortel 및 삼성도 최근에 PersonalJava를 그들의 새로운 웹폰 제품에 결합시키기로 공표하였다.

Java를 탑재한 환경은 운영체제와 Java가 사용하는 운영체제 API사이의 차이를 극복하기 위하여 적응 계층을 두는 것이 특징이다. 그래서 PersonalJava를 사용하고자 한다면, 실시간 운영체제, Java 적응 계층, PersonalJava 등의 세 가지 소프트웨어에 대한 라이선스 비용을 부담하여야 한다.

5. 결 론

가전제품과 같은 내장형 디바이스가 갖추어야 하는 요구조건은 기술적인 요소보다는 제품의 라이프사이클이나 시장 상황에 의해서 결정된다. 그러나 개발의 어려움에도 불구하고 전 세계적으로 매년 약 10억개의 내장형 프로세서가 거래되고 있으며, 내장형 시스템 산업의 빠른 증가로 2010년에는 내장형 소프트웨어 개발자가 데스크탑 컴퓨터를 위한 소프트웨어 개발자보다 10배 더 많을 것으로 전망되고 있다.

Java 기술을 사용한 내장형 소프트웨어 개발은 사용자 입장의 요구보다는 개발자의 요구를 수용하는 것이다. 현재와 같이 가전제품의 라이프사이클이 짧은 상황에서 내장형 소프트웨어를 개발하는데 가장 중요한 점은 현재 사용

되고 있는 코드를 확장하여 재사용하는 것이다. 내장형 소프트웨어는 한번 개발되면 더 큰 프로그램으로 개발될 수 있어야 하고 다양한 프로세서에서 수행될 수 있어야 한다. 또한 메모리 양이 제한된 환경에서도 수행될 수 있어야 하며 재구성이 가능하도록 모듈화되어야 한다. 이와 같은 요구는 객체 지향적이면서도 플랫폼 독립적인 장점을 가진 Java를 내장형 소프트웨어 개발에 적용함으로써 달성될 수 있다.

향후 가전제품 시장의 영업적인 측면 요구조건인 적기제품출시(time-to-market), 신뢰성, 낮은 가격 등을 만족하기 위해서는 모든 가전제품의 내장형 소프트웨어의 플랫폼을 단일화하고 이를 기반으로 하여 단기간에 내장형 소프트웨어를 개발할 수 있어야 한다. 현재 일본에서는 Java 기반의 가전제품용 내장형 소프트웨어 개발 표준안을 마련하고 있다. 그러나 국내에서는 Java를 사용한 내장형 소프트웨어의 개발이 초기 단계이며, 범용 언어를 사용한 내장형 소프트웨어의 개발 수요마저도 대부분 외국 회사의 제품으로 대체하고 있기 때문에 Java를 기반으로 하는 내장형 소프트웨어 개발 환경이 빠른 시일 내에 구축되어야 할 것이다.

참고문헌

- [1] Sami Shaio, Arthur van Hoff, Herb Jelinek, Java and HotHava : A Comprehensive Overview, Proceedings of COMPCON96, pp. 424-428, 1996.
- [2] Datapro Information Services Group, JavaSoft : Java Programming Language, Jun. 1997.
- [3] <http://java.sun.com/products/personaljava>.
- [4] <http://www.realtime-info.be/encyc/magazine/article/bench2.htm>.
- [5] <http://www.wrs.com/html/trjava.html>.
- [6] <http://www.cse.ucsc.edu/research/embedded/index-new.html>.
- [7] <http://www.microware.com>.

박 성 우



1996 한국과학기술원 전산학 학사
1998 한국과학기술원 전산학 석사
1998~현재 시스템공학연구소
자바센터 근무

손 중 문



1991 부산대학교 컴퓨터공학 학사
1993 부산대학교 컴퓨터공학 석사
1997 부산대학교 컴퓨터공학 박사
1997~현재 시스템공학연구소
부산컴퓨팅 연구실
근무
관심분야: 실시간 시스템, 시스템 프로그램, 내장형 자바 시스템, 멀티미디어 시스템

김 태 근



1987 한양대학교 수학 학사
1990 뉴욕주립대학교 전산수학 석사
1993 뉴욕주립대학교 전산수학 박사
1992~현재 시스템공학연구소
부산컴퓨팅 연구실
실장
1997~현재 시스템공학연구소
자바센터 센터장

● '98 데이터베이스 춘계튜토리얼 ●

- 일 자 : 1998년 5월 21일(목)~22일(금)
- 주 제 : 데이터마이닝
- 주 최 : 데이터베이스연구회
- 문 의 처 : 박종수 교수(02-920-7179)