

임베디드 소프트웨어를 위한 프로파일 레포팅 뷰어의 설계 및 구현

고방원^o, 신경호, 김상현, 유재우
송실대학교 컴퓨터학과

withfox^o@gmail.com, delios@orgio.net, {shkim, cwwoo}@comp.ssu.ac.kr

A Design and Implementation of A Profile Reporting Viewer for Embedded Softwares

BangWon Ko^o, KyoungHo Shin, SangHeon Kim, CheaWoo Yoo
Dept. of Computing, Soongsil University

요 약

본 논문은 임베디드 소프트웨어 개발자가 임베디드 소프트웨어 개발시 쉽고 편리하게 테스트 및 프로파일의 결과를 분석하고 개발의 효율성을 높일 수 있도록 직관적인 GUI를 가지는 레포팅 뷰어를 설계 및 구현한다. 제안하는 레포팅 뷰어는 프로파일 결과 데이터 처리기와 GUI 레포트 생성기로 구성된다. 결과 데이터 처리기는 임베디드 소프트웨어의 성능 프로파일링을 통해 생성된 문자 스트링 형태의 저수준 결과를 XML 문서로 구조화 하여 객체 형태의 API를 제공한다. 레포트 생성기는 결과 데이터 처리기에 의해 생성된 API 객체를 이용하여 다양한 그래픽 기반 프로파일 레포트 뷰를 출력한다. 사용자는 제안하는 레포팅 뷰어가 제공하는 객체 형태의 API를 통해 자신이 원하는 프로파일 레포트 화면을 구성할 수 있기 때문에 기존 소프트웨어보다 더욱 다양하고 직관적인 레포트 뷰(view)를 생성할 수 있다. 따라서 사용자는 보다 빠르고 다양한 방법으로 성능 분석과 코드 수정이 가능하여 효율적이고 신뢰성 있는 임베디드 소프트웨어를 개발할 수 있다.

1. 서 론

임베디드 소프트웨어는 일반적인 소프트웨어와는 다르게 기기의 제한된 기능을 이용하여 최적의 성능을 발휘해야한다.[1] 그러므로 일반적인 소프트웨어의 경우보다 더욱 많은 성능 측정과 테스트가 필요하며 성능 측정 및 테스트 결과의 분석은 개발의 효율성 향상에 기여할 수 있다. 대부분의 경우 임베디드 소프트웨어 개발은 호스트/타겟 방식에 의한 교차개발(Cross Development) 환경을 통해 이루어진다. 따라서, 임베디드 소프트웨어의 테스트 및 프로파일링은 타겟 시스템에서 수행되며, 호스트 시스템에서 분석을 통해 사용자에게 보고된다. 타겟에서는 테스트 및 프로파일링 시에 각 함수에 대한 CPU와 메모리(스택, 힙 등)의 사용량과 타겟 전체의 메모리 사용량과 같은 자원 사용 정보가 문자 스트링 형태의 저수준 결과로 생성된다. 생성된 결과는 API 형태의 데이터 처리기를 통해 XML형식의 정보로 변환되고 레포트 생성기를 통해 사용자에게 보다 직관적인 형태의 GUI를 가지는 뷰(view)로 제공된다.

본 논문에서는 기존의 테스트 도구들과 비교해 보다 직관적인 사용자 위주의 GUI 설계 기법에 대해 고찰한다. 레포팅 뷰어의 화면 구성 설계 시 최우선적으로 텍스트 위주의 다른 도구에 비해 원형차트(Pie Chart) 및 막대차트(Bar Chart)를 사용하여 결과를 한눈에 파악할

수 있게 설계하였으며, 자세한 각 항목들은 테이블로 표현해 세부 정보의 표현까지 빠짐없이 확인을 할 수 있게 하였다. 그리고 인터페이스 설계 기법을 다룬 여러 참고 문헌과 논문을 활용하여 고려할 사항을 반영하고, XML 형태로 가공된 정보의 상호관계를 분석하여 개발자가 임베디드 소프트웨어 개발 시에 보다 효율적으로 참고할 수 있는 GUI를 구현한다.

본 논문의 구성은 다음과 같다. 1장의 서론에 이어 2장에서는 인터페이스 설계와 관련된 연구 및 기술을 정리하였다. 3장에서는 인터페이스 설계 기법 및 방안에 대해 설명하고 4장에서는 뷰어 GUI 구현 및 구현 결과를 제기하였으며 5장은 결론이다.

2. 관련 연구 및 기술

GUI는 사용자에게 시각 요소(visual component)를 통해 프로그램을 제어하고 동작시킬 수 있는 쉬운 인터페이스를 제공한다. 인간의 운동 체계와 컴퓨터를 이용한 업무 처리시의 행동 특성을 이해한 애플리케이션의 GUI는 그렇지 못한 애플리케이션에 비해 생산성 및 사용의 용이성을 비약적으로 향상시킬 수 있다.

인터페이스 설계는 운영 체제별, 응용 프로그램별로 다양하게 존재하지만 인간의 운동 체계와 시각적 감성을 충분히 고려하고 있다는 공통점이 있다. 본 논문은 원도

우 형태의 운영 체제 중 가장 우수한 GUI 기술을 가지고 있는 Mac OS X 상의 GUI 설계 가이드라인을 중심으로, 뷰어의 화면 구성을 설계하였다. 이를 통해 획득할 수 있는 이점은 직관적인 인터페이스와 사용자의 작업 안내를 통하여 애플리케이션 자체의 기능을 설명하는 시간을 단축할 수 있고, 애플리케이션의 문서화가 보다 간편하게 이루어질 수 있고, 특수한 애플리케이션일 경우 GUI의 적용을 통해 사용자들의 사용 활성화를 쉽게 도모할 수 있다.

사용자 인터페이스의 설계를 위해서 원칙적으로는 컴퓨터공학, 인지공학, 심리학, 사회학 등 다양한 학문이 복합적으로 연구되어야 한다[2]. 그만큼 사용자에 대해 우수한 인터페이스를 설계하는 것은 쉬운 일이 아니다. 본 뷰어는 수많은 인터페이스 설계 관련 기술 중에 일관성과 미적 완전함 위주의 설계가 사용되었다. 다음은 일관성과 미적 완전함에 대한 간략한 설명이다.

1) 일관성

GUI를 일관성 있게 작성하는 것은 두 가지 이유로 중요하다. 첫째는 A라는 애플리케이션에서 B라는 애플리케이션으로 사용자가 정보 및 기술을 전승할 수 있도록 지원한다는 점에서 중요하다. 둘째는 일관성 있는 인터페이스는 사용자에게 애플리케이션 사용에 있어서의 집중력과 친숙함을 가지도록 할 수 있다는 점이다.

2) 미적 완전함

예전의 컴퓨터 애플리케이션의 GUI는 미적인 개념을 충분히 반영하지 못하였다. 물론 컴퓨터 사용의 주요 목적은 정확한 계산의 결과와 이의 활용에 있다.

현재의 컴퓨터 애플리케이션은 본래의 목적에 부가적으로 사용자가 다루기 편리한 인터페이스를 제공하는 추세이다. 이런 점을 위해서는 시각적인 아름다움을 제공하는 GUI의 설계가 필수적으로 요구된다. 미적 완전함을 위해서는 우선 일관적인 그래픽 객체의 제공, 혼란을 주지 않을 정도의 객체 제공, 정교하고 규칙적인 레이아웃 제공을 통한 규칙성의 달성 등이 고려되어야 한다.

3. 인터페이스 설계 기법 및 방안

다음 그림 1은 본 논문에서 제안하는 임베디드 소프트웨어를 위한 프로파일 레포팅 뷰어의 전체적인 구조도이다.

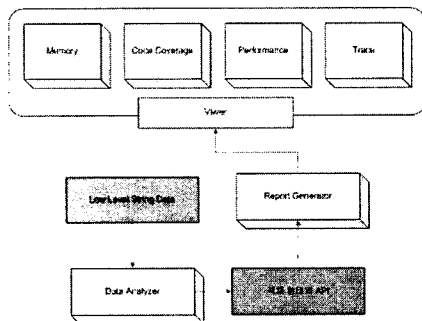


그림 1. 레포팅 뷰어의 전체 구조도

그림 1에서 문자 스트링 형태의 저수준 결과는 API 형태로 처리하는 데이터 처리기를 통해 XML형식의 정보로 변환되고 레포트 생성기를 통해 사용자에게 보다 직관적인 형태의 GUI를 가지는 뷰(view)로 변환된다.

제안하는 뷰어는 Java를 이용해 구현되었으며, 생성되는 API는 Java 클래스이다. 뷰어는 다음 표 1과 같은 4가지에 대한 성능 평가 항목을 가진다.

표 1. 테스트 및 프로파일링의 종류

테스팅 및 프로파일링의 종류	기능
Trace	타겟 기반의 Runtime Trace 결과를 UML의 Sequence Diagram 형태로 표현한다.
Memory	Heap 공간에 설정된 메모리 누수를 찾아낸다.
Performance	Function 및 Method의 수행을 나타내어 어플리케이션의 병목지점을 찾아낸다.
Code Coverage	Code 수행에 대한 적용 범위를 완벽하게 분석한다.

뷰어 설계시 핵심적으로 고려해야 할 사항은 다음 두 가지이다.

첫째, 뷰어는 위의 표 1에서 소개한 각 네가지 항목에 대해 문자 스트링 형태의 저수준 결과의 상관관계를 파악하여 이들간의 의존성을 정확하게 인지해야 한다. 각각의 항목을 보기 전에 임베디드 소프트웨어 개발자가 한눈에 파악할 수 있는 종합적인 뷰(view)를 제공함으로써 임베디드 소프트웨어 개발에 익숙하지 않은 사용자들에게도 개발의 편의성을 제공할 수 있는 점이 반드시 고려되어야 한다.

둘째, 결과의 오류에 대한 범위를 지정하고 검증해야 한다[3][4]. 임베디드 소프트웨어는 개발의 특성상 개발시 많은 시간과 더불어 반복적인 테스트 및 프로파일링 작업이 요구된다. 이는 개발의 효율을 저하시키는 요인이 될 수밖에 없다. 따라서 GUI를 설계할 때 각 항목에 대한 간단한 설명과 더불어 오류를 체크하여 사용자에게 피드백할 수 있도록 구현하여야 한다. 이를 위해, 본 논문에서 제안하는 레포팅 뷰어는 임베디드 소프트웨어에 대한 프로파일링의 결과를 사용자에게 일관적으로 제공하기 위해 4가지의 평가 항목 외에 전체 결과에 대한 요약 정보인 overview 레포트 뷰를 제공하며, 이전 프로파일 결과와 피드백을 통해 새롭게 생성된 프로파일 결과에 대한 비교 뷰를 제공한다. 이를 통해, 개발자는 프로파일링 결과를 한눈에 쉽게 파악할 수 있으며, 피드백을 통해 코드 수정이 올바르게 되었는지를 빠르고 효율적으로 확인할 수 있다.

4. 시스템 구현 및 실험

4.1. 구현 환경

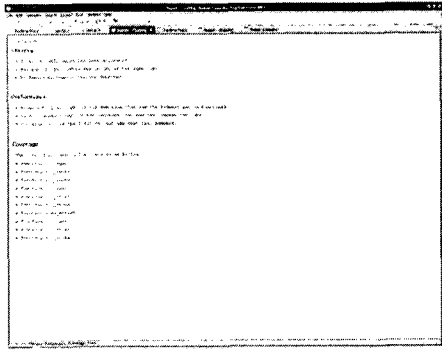


그림 2. 구현된 뷰(Overview 화면)

GUI 구현을 위해 선택한 언어는 Java 이다. 이는 임베디드 개발 환경이 다양한 플랫폼이 될 수 있다는 점을 감안하여 이식성을 고려한 결정이다. 또한 Java는 다양한 그래픽 API가 제공되어 사용자 친화적인 GUI를 구현할 수 있다.

4.2. 뷰의 구성

구현한 GUI는 기본적으로 5개의 탭으로 구성되어 있다. 4가지 평가 항목 외에 추가된 Overview 항목은 간략한 텍스트 표현으로 다른 4가지 항목의 요약을 담고 있다. 그리고 Memory, Performance, Code Coverage의 항목은 원형차트(PieChart) 및 막대차트(BarChart)를 통해 각각의 결과를 그래픽 환경으로 보여주며, Trace 항목은 실시간으로 그려진 Sequence Diagram을 통해 임베디드 소프트웨어의 시간별 자원 사용량과 호출되는 함수 등등의 여러 정보를 파악할 수 있다. 그리고 결과 비교시에는 각각의 탭들이 좌우에 걸쳐 나오게 되어 있어 한눈에 비교가 가능하다. 이런 각각의 항목들은 기존의 임베디드 소프트웨어 테스트 도구인 AstonLinux의 CodeMaker와 Rational의 Rational Test Real-Time과는 차별화된 기능들이며, 이로 인해 임베디드 소프트웨어 개발시의 효율의 향상을 기대할 수 있다.

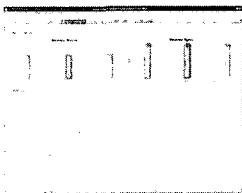


그림 3-1 Memory

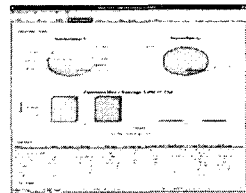


그림 3-2 Performance

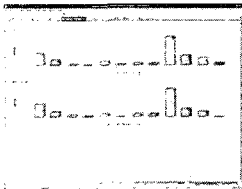


그림 3-3 Code Coverage

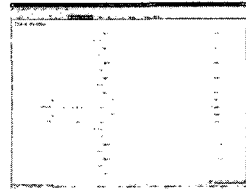


그림 3-4 Trace

4.3. 구현 결과

구현된 뷰어는 그림 2와 같다. 탭으로 모든 항목을 표

시함으로써 GUI 설계시 일관성을 가지게 되고 사용자는 집중력과 친숙함을 가지게 된다.[1]

그림 2의 아래쪽의 탭을 선택함으로써 그림 3과 일관적인 그래프 위주의 그래픽 객체를 제공했으며, Java에서 제공해주는 정교하고 규칙적인 레이아웃 제공을 통해 미적인 완전함[1]까지 고려했으며, 각각의 항목에 대한 결과 화면을 확인할 수 있다.

그리고 그림 4와 같이 서로 다른 프로파일링 결과의 비교시에 각각의 프로파일링 결과를 확인할때와 같은 결과 화면을 좌우로 배치함으로써 일관성을 유지^[1]하도록 구현되었다.

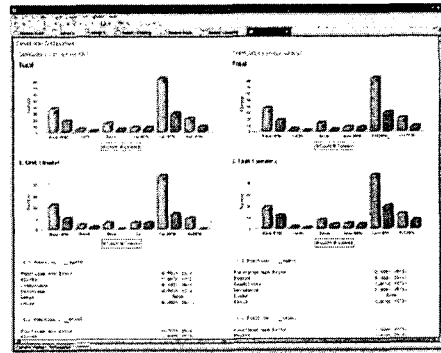


그림 4. 서로 다른 두 프로파일링 결과의 비교

5. 결론 및 향후과제

본 논문은 임베디드 소프트웨어를 위한 테스트 및 프로파일 레포팅 뷰어의 GUI의 설계와 구현에 대한 것이다. 텍스트 위주의 다른 임베디드 소프트웨어 레포팅 도구와는 차별화된 그래프 위주의 사용자 친화적이고 직관적인 뷰어를 구현해 임베디드 소프트웨어 개발시의 테스트 및 프로파일링 결과의 분석시간을 감소시켜 개발시에 소요되는 시간을 줄여주고 임베디드 소프트웨어 개발에 익숙하지 않는 개발자들에게 보다 편한 환경을 제공함으로써 개발의 효율성 증대에 기여할 수 있을 것이다.

[1] 공기석, 손승우, 임채덕, 김흥남, “내장형 실시간 소프트웨어의 원격디버깅을 위한 디버그에이전트의 설계 및 구현”, 한국정보과학회 가을 학술발표논문집 Vol. 26, No 2, pp.125~127, 1999.

[2] Apple Computer, “Inside Mac OS X, Aqua Human Interface Guide Line”, Apple Computer Inc., 2002. 6.

[3] 이남지 외 3명, “홈네트워크 디바이스 관리를 위한 UPnP 프레임 기반의 GUI 시스템”, 한국정보과학회 봄 학술발표논문집 vol.30, 2003. 3.

[4] 조은경, 최영미, “사용자의 가상 경험을 유도하는 인터페이스 설계”, 한국멀티미디어학회 춘계학술발표논문집, 2003. 3.