

컴포넌트 기반 임베디드 소프트웨어의 품질관리 방법⁺

김금옥*, 이하용**, 양해솔***,

*호남대학교 정보통신대학

**서울벤처정보대학원대학교

***호서대학교 벤처전문대학원

e-mail:lhyazby@hanmail.net, hsyang@office.hoseo.ac.kr

The Quality Management Method for Embedded Software on the component

Jin, Jin-yu*, Ha-Yong, Lee**, Hae-Sool, Yang*,

*College of Information and Communication, Honam University

**Seoul Univ. of Venture & Information

***Graduate School of Venture, Hoseo University

요 약

유비쿼터스 시대 다가오면서 임베디드 소프트웨어가 다양한 분야에서의 개발이 증대되며 요구사항도 변화되고 있다. 최근에 다양한 컨버전스 서비스를 지원할 수 있는 컴포넌트 구조 기반의 서비스 지향 모바일 미들웨어 플랫폼 기술 개발이 확산되는 추세이다. 기존의 개발방법들은 하드웨어와의 밀접한 관계, 여러 유사 도메인에 대한 고수준의 재사용성 요구 등 임베디드 소프트웨어가 갖고 있는 특성들을 효과적으로 반영하지 못하고 있다. 본 연구에서는 컴포넌트 기반 임베디드 소프트웨어 개발 프로세스를 정립하고 각 단계에서 필요한 최적의 품질관리 체계를 구축함으로써 경쟁력 있는 임베디드 소프트웨어 제품의 개발을 지원할 수 있는 방법론을 확립하였다.

1. 서론

소프트웨어 개발단계에서 오류를 최소화하여 품질과 신뢰성을 확보하는 것이 제품 경쟁력의 핵심이라는 업계의 인식이 확산되고 있다. 컴포넌트 기반의 임베디드 소프트웨어 개발은 산업계에서 성장하고 있는 중요한 분야 중의 하나이다[4].

재사용성과 유지보수성이 우수한 컴포넌트는 e-Business가 활성화 되면서 급속하게 변화하는 복잡한 비즈니스 로직 구현 등의 요구사항을 만족 시킬 수 있는 대안으로 산업계에서 활성화되어 있어야 한다. 기존의 개발방법들은 하드웨어와의 밀접한 관계, 여러 유사 도메인에 대한 고수준의 재사용성 요구 등 임베디드 소프트웨어가 갖고 있는 특성들을 효과적으로 반영하지 못하고 있다. 개발 프로세스를 혁신하고 오류를 최소화하여 임베디드 소프트웨어 개발 생산성과 품질을 동시 제고하여야 한다. 개발 생산성과 제품의 품질을 향상을 위해서는 “계획⇒ 설계 ⇒ 구현 ⇒ 테스트” 단계와 같은 임베디드 소프트웨어 개발 프로세스를 정립하고 각 단계에서 필요한 최적의 품질관리 체계를 구축하여야 한다. 따라서 본 연구에서는 개발 생산성과 품질 향상을 동시 확보할 수 있는 컴포넌트 기반 임베디드 소프트웨어 개발 프로세스 제안하고 적합한 품질평가 체계를 구축하였다[1,2].

2. 관련연구

(1). 국내·외 시장 현황

세계 임베디드 소프트웨어 시장은 '06년 1,148억 달러에서 '10년 1,305억 달러 규모로 연평균 3.9%, 국내는 '06년 84억 달러에서 '10년 100억 달러로 연평균 13.3%의 고성장이 전망된다. 세계 시장이 지속적으로 성장하는 가운데 미국 시장의 비중이 줄어드는 반면 EMEA(유럽&중동아시아) 및 AP(아태지역) 비중이 증가한다. Mobile Application OS 및 RTOS 시장의 경우 : 미국은 53.0%(2005) → 55.5%(2006), EMEA는 27%(2005) → 29%(2006), AP는 17.5% (2005) → 18%(2006)로 증가 및 2009년까지 16.6%의 가장 높은 성장률을 나타낼 것으로 전망된다. 한국의 임베디드 소프트웨어 시장규모는 미국, 유럽, 일본에 이어 4위로 중국과 비슷하다[3].

(2) 표준화 동향

다양한 컨버전스 서비스를 지원할 수 있는 컴포넌트 구조 기반의 서비스 지향 모바일 미들웨어 플랫폼 기술 개발이 확산되는 추세이다. 각종 모바일 서비스 제공에 필수적인 요소들을 포함하는 공통 플랫폼은 없으며, 다양한 기능을 탑재해 단말이 제공하는 서비스에 특화된 플랫폼들이 존재한다.

LiPS, OMA, OMTF 등 국제 표준화 기구를 중심으로 모바일 환경을 위한 플랫폼 표준화 활동이 이루어지고 있으나, 다양한 서비스를 위한 공통 컴퓨팅 플랫폼 관련 표

⁺ 본 연구는 지식경제부와 IITA의 대학 IT연구센터 지원사업의 연구결과로 수행되었음(IITA-2009-(C1090-0902-0032)).

준화는 진행되고 있지 않다. 국내에서는 무선인터넷 서비스를 위한 미들웨어 플랫폼으로 무선인터넷표준플랫폼(WIPI) 규격이 제정되어 상용화 되고 있으며, 3G 등 변화된 시장에 보다 경쟁력을 갖기 위한 새로운 버전의 규격 제정을 위해 활동 중이다.

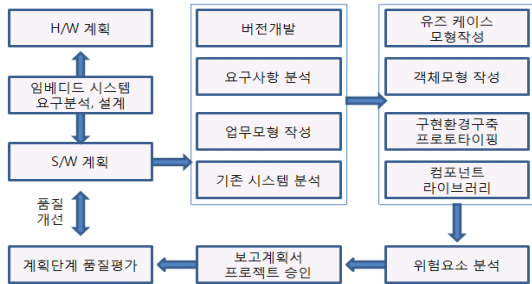
모바일 중심의 컨버전스 시장에 대한 경쟁력 선점을 위한 노력으로 다양한 기기와 서비스를 확장성 있게 제공할 수 있는 통합 플랫폼 개발 방안 모색중이다. 센서 노드들이 IPv6 스택을 보유하는 방식으로 USN 사용자가 어디서든 IPv6 통해 통신 가능하도록 센서네트워크와 IPv6 네트워크를 직접 연동하는 기술인 6LoWPAN이 최근 IETF에서 표준화 진행하고 있다.

3. 컴포넌트 기반 임베디드 소프트웨어 개발

프로세스

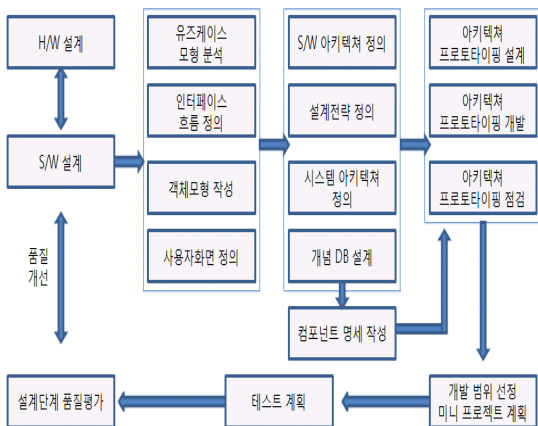
본 연구의 개발방법론은 우선 계획단계, 설계단계, 구현단계, 인도단계의 네 단계로 구분된다. 그리고 각 단계는 명확한 목표설정과 이의 달성을 위한 실행 그리고 평가를 포함하는 작은 프로젝트 수준으로 정의한다. 각 단계의 기준은 다음과 같다.

(1) 계획단계: 개발대상 시스템의 명확한 범위를 확정하며, 이를 검증하기 위한 UI수준의 구현물을 개발함.



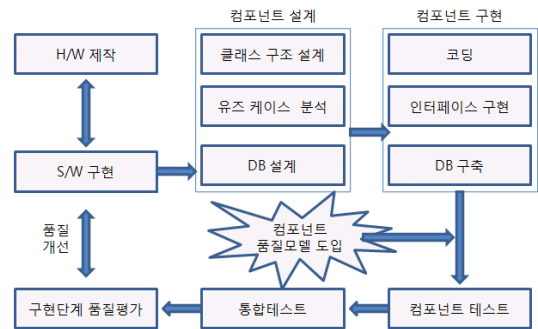
(그림 1) 계획단계 개발 프로세스]

(2) 설계단계: 시스템 구현을 위해 모든 아키텍처적인 위험요소가 제거되어 최종 시스템 아키텍처를 확정하고 각 컴포넌트 도출을 완료함.



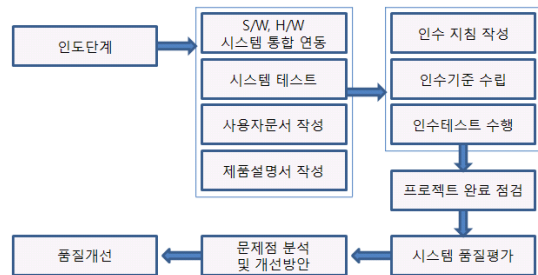
(그림 2) 설계단계 개발 프로세스

(3) 구현단계: 모든 컴포넌트 들이 개발되고 통합 테스트까지 마무리 하였으며, 아키텍처 정의단계에서 정의된 테스트 기준을 통과함.



(그림 3) 구현단계 개발 프로세스

(4) 인도단계: 실제 사용자가 사용하도록 교육과 설치작업을 실시하기 위한 모든 작업이 완료되었으며, 시스템테스트와 인수테스트 기준을 만족시킴.



(그림 4) 인도단계 개발 프로세스

4. 단계별 품질관리 방법 및 사례분석

(1) 임베디드 소프트웨어 품질평가 매트릭

본 연구에서는 ISO/IEC 9126의 국제표준과 임베디드 시스템의 품질특성을 바탕으로 컴포넌트 기반 임베디드 소프트웨어 품질평가 매트릭을 개발함으로써 객관성 및 타당성을 최대화하는 목적으로 하였다. 본 연구에서의 컴포넌트 기반 임베디드 소프트웨어 개발 프로세스를 요구 분석 단계, 설계단계, 구현단계, 인도단계로 분류하여 각 단계에 해당되는 매트릭을 개발하였다. 품질특성 매트릭의 예는 아래와 같다.

기능성-적합성-제품사양 개정율: 제품의 기능이 어느 정도 개정(추가, 변경, 소거)되었는가를 측정하는 지표이다.

$$\text{계산식: } \frac{\text{개정 기능 항목수}}{\text{제품 기능 항목수}}$$

(2) 체크리스트 개발

<표 1> 내부특성의 체크리스트의 예

완전성						
평가요인 항목	세부항목	평가				
		1.0	0.9	0.8	0.7	0.6
1.요구된 기능이 어느 정도 실현되어 있는가?	1.1.1 요구사항서에 기술된 요구 기능중 비논리적인 요구기능은 어느 정도인가? 1%미만(1.0), 10%미만(0.8), 10%이상(0.6)					

	1.1.2 요구된 기능들이 요구사항서 어느 정도 충실히 제공되는가? 100%(1.0), 80%이상(0.8), 70%미만(0.6)				
	1.1.3 요구사항서의 기능중 아무런 효과를 발휘하지 못하는 기능은 어느 정도인가? 2%미만(1.0), 10%미만(0.8), 10%이상(0.6) 계:				
2.요구된 기능이 현재의 기술 수준으로 제공 가능한가?	1.2.1 요구된 기능이 현재의 하드웨어 기술수준으로 지원되지 못하는 기능은 어느 정도인가? 5%미만(1.0), 10%미만(0.8), 10%이상(0.6)				
	1.2.1 요구되는 기능이 현재 소프트웨어 기술수준으로 지원되지 못하는 기능은 어느 정도인가? 5%미만(1.0), 10%미만(0.8), 10%이상(0.6) 계:				
3. 시스템화에 필요한 기능이 명확히 정의되어 있는가?	1.3.1 시스템 각 부시스템별 요구사항서가 있는가? 충분(1.0), 보통(0.8), 불충분(0.6)				
	1.3.2 시스템에서 필요로 하는 기능수와 요구 분석서의 기능수가 어느 정도 일치하는가? 5%이상(1.0), 90%이상(0.8), 90%이하(0.6) 계:				
문제점 분석:					

(3) 요소 데이터 정리

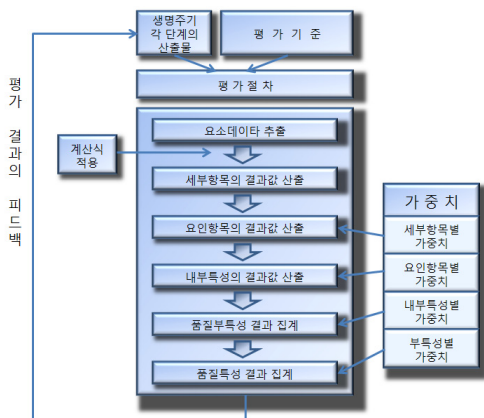
본연구의 요소데이터의 예는 <표 2>와 같다.

<표 2> 요소데이터의 예

번호	세부항목		요소 데이터
1.1.1	요구사항서에 기술된 요구 기능중 비논리적인 기능은 어느 정도인가? 점수 B/A	A	요구사항서에 기술된 모든 기능수
		B	요구사항서에 정의된 기능중 비논리적인 기능수

(4) 평가절차

생명주기 각 단계별 품질평가 절차는 우선 각 단계의 산출물과 평가기준을 고려하여 평가절차를 확립한 다음 계산식을 적용하여 요소데이터를 추출한다. 그 다음 세부항목의 결과 값을 산출하며 내부특성의 결과값을 산출한다. 끝으로 품질부특성의 결과를 집계하는 순서로 상세한 평가 절차는 <그림 5>과 같다.



<그림 5> 생명주기 각 단계의 품질특성평가 절차

(5) 사례분석

본 절에서는 본 연구에 기술한 품질평가 메트릭을 도입한 평가 사례에 대하여 기술한다. 연구대상 사례는 임베디드 시스템 개발 업체 중 S사의 PDA 개발팀을 대상으로 단계별 품질관리방법을 도입후의 품질개선 효과를 분석하였다.

<표 3> 요구 분석단계 품질특성 집계표

번호	품질특성	내부특성	가중치	점수
1.	기능성	적합성	0.25	83.27
		정확성	0.25	82.97
		상호운용성	0.1	84.01
		유연성	0.2	78.17
		보안성	0.2	81.19
		계: 81.83		
2.	신뢰성	성숙성	0.3	83.10
		오류허용성	0.2	84.06
		회복성	0.5	84.52
		계: 84.00		
3.	사용성	이해성	0.3	79.30
		습득성	0.2	79.79
		운용성	0.5	78.80
		계: 79.15		
4.	유지보수성	해석성	0.3	84.06
		변경성	0.3	84.06
		안정성	0.2	86.00
		시험성	0.2	84.06
				계: 84.45
5.	이식성	환경적응성	0.2	75.23
		이식작업성	0.3	81.77
		일치성	0.2	75.23
		치환성	0.3	78.17
				계: 78.08
평균		81.50		

<표 4> 설계단계 품질특성 집계표

번호	품질특성	내부특성	가중치	점수
1.	기능성	적합성	0.25	86.17
		정확성	0.25	87.29
		상호운용성	0.1	85.20
		유연성	0.2	83.84
		보안성	0.2	81.82
		계: 85.01		
2.	신뢰성	성숙성	0.3	86.38
		오류허용성	0.2	85.86
		회복성	0.5	85.01
		계: 85.59		
3.	사용성	이해성	0.3	85.50
		습득성	0.2	85.52
		운용성	0.5	84.79
		계: 85.14		
4.	유지보수성	해석성	0.3	85.54
		변경성	0.3	85.54
		안정성	0.2	85.82
		시험성	0.2	86.20
				계: 85.72
5.	이식성	환경적응성	0.2	85.20
		이식작업성	0.3	87.39
		일치성	0.2	85.20
		치환성	0.3	83.84
				계: 85.45
평균		85.38		

<표 5> 구현단계 품질특성 집계표

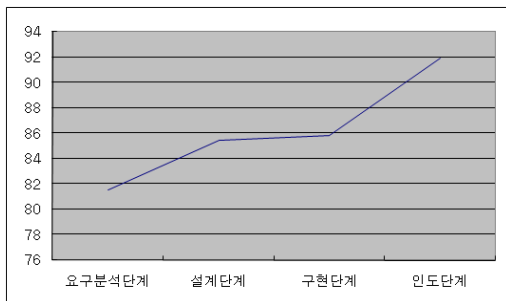
번호	품질특성	내부특성	가중치	점수
1.	기능성	적합성	0.25	85.91
		정확성	0.25	86.04
		상호운용성	0.1	90.36
		유연성	0.2	89.12

		보안성	0.2	85.57
		계: 86.96		
2.	신뢰성	성숙성	0.3	85.28
		오류허용성	0.2	81.74
		회복성	0.5	84.45
		계: 84.15		
3.	사용성	이해성	0.3	82.16
		습득성	0.2	82.25
		운용성	0.5	82.12
		계: 82.16		
4.	효율성	실행효율성	0.5	83.38
		자원효율성	0.5	89.60
		계: 86.49		
5.	유지보수성	해석성	0.3	86.07
		변경성	0.3	86.07
		안정성	0.2	86.03
		시험성	0.2	85.85
		계: 86.02		
6.	이식성	환경적응성	0.2	90.36
		이식작업성	0.3	89.04
		일치성	0.2	90.36
		치환성	0.3	89.12
		계: 89.58		
평균	85.89			

<표 6> 인도단계 품질특성 집계표

번호	품질특성	내부특성	가중치	점수
1.	기능성	적합성	0.3	95.31
		정확성	0.3	95.31
		상호운용성	0.2	98.21
		보안성	0.2	89.46
		계: 94.75		
2.	신뢰성	성숙성	0.4	91.58
		오류허용성	0.3	90.40
		회복성	0.3	91.18
		계: 91.10		
3.	사용성	이해성	0.3	89.33
		습득성	0.3	89.33
		운용성	0.4	89.33
		계: 89.33		
4.	효율성	실행효율성	0.5	92.16
		자원효율성	0.5	93.56
		계: 92.86		
5.	유지보수성	해석성	0.25	90.50
		변경성	0.25	90.50
		안정성	0.25	87.71
		시험성	0.25	90.37
		계: 89.78		
6.	이식성	이식작업성	1.0	93.83
		계: 93.83		
평균	91.94			

본연구의 단계별 품질관리 방법을 도입한 품질개선 결과는 다음 <그림6>과 같다.



<그림 6> 품질 개선 효과

5. 결론

본 연구에서는 마르미-D와 컴포넌트 개발방법론을 토대로 생산성과 품질 향상을 동시 확보할 수 있는 임베디

드 소프트웨어 품질관리방법을 개발하였다. 개발 프로세스 혁신하고 오류를 최소화하기 위해서는 “계획⇒ 설계 ⇒ 구현 ⇒ 테스트” 단계와 같은 임베디드 소프트웨어 개발 프로세스를 정립하고 각 단계에서 필요한 최적의 품질검증 단계를 구축하였다. 향후 연구에서는 본연구에 제시한 임베디드 소프트웨어의 개발 프로세스 개선 및 혁신과 실질적인 활용을 통해 국내 임베디드 소프트웨어의 생산성 향상을 시켜 소프트웨어 유통시장의 안정성 확보, 고품질 임베디드 소프트웨어 개발 의욕을 제고하여 고부가가치 임베디드 소프트웨어 개발 등 측면에서 국제경쟁력을 확보할 수 있을 것으로 기대한다.

참고문헌

- [1] “국내 개발체계 기반의 임베디드 소프트웨어 발전방향”, ETRI, 소프트웨어 산업 부문별 시장/기술 전망 세미나, 2004.
- [2] 양해술, “Embedded S/W의 품질평가 모델 개발 연구” 한국정보통신기술협회 위탁과제, 최종보고서, 2002.11.
- [3] Gartner, 2006. www.thegartnergroup.com
- [4] “임베디드 소프트웨어 산업 실태조사”(KESIC), 2007.
- [5] 오영배, 나희동, 박준성, 백두권, “컴포넌트 기반 소프트웨어 개발 프로세스”, 한국정보과학회, 제20권 제3호, ISSN 1229-6821, 2002.3.
- [6] 임형택, 임채덕, 박승민, 김홍남, “임베디드 소프트웨어 개발 도구 동향”, IT정보, www.naver.com
- [7] 김남희, 신석규, “소프트웨어 컴포넌트 시험기술 동향” TTA저널 제88호
- [8] Embedded, Everywhere a Research Agenda for Networked Systems of Embedded Computers, National Research Council, National Academy Press, 2001.
- [9] ISO/IEC 9126, “Information Technology - Software Quality Characteristics and metrics- Part” 1, 2, 3.
- [10] ISO/IEC 14598, “Information Technology - Software product evaluation - Part 1, 2, 3, 4, 5, 6”.
- [11] ISO/IEC 12119, “Information Technology - Software Package - Quality requirement and testing”.