

소프트웨어 수명주기에서 품질특성의 중요도 결정 기법

조 재 규* · 이 길 섭** · 이 승 종**

요 약

정보화 사회로 발전이 되면서 고품질 소프트웨어에 대한 관심과 요구가 증대되고 있다. 이에 따라 소프트웨어 제품 품질을 평가 및 관리하는 활동이 개발 활동과 함께 중요시 되고 있다. 또한, 소프트웨어는 관련되는 이해당사자 마다 품질을 바라보는 다양한 시각이 존재할 수 있으며, 개발의 각 단계에서 중점적으로 관리되어야 할 품질특성이 필요하게 된다. 이러한 다양한 관점과 수명주기의 특성을 고려한 품질특성의 중요도는 품질관리 활동을 보다 효과적으로 수행할 수 있게 해준다. 그러나 소프트웨어 품질의 중요도에 대한 연구는 그 필요성과 효과성이 제대로 부각되지 않아 많은 연구가 진행되지 않았다. 본 논문에서는 소프트웨어 제품 품질의 기준이 되는 ISO/IEC 9126을 기반으로 소프트웨어 수명주기에서 품질특성의 중요도를 결정하는 기법을 제시하고자 한다. 제안된 기법은 소프트웨어와 관련되는 주요 이해당사자들을 대상으로 한 설문조사와 AHP 기법을 이용한 중요도 분석을 통해 이루어진다. 이를 통하여 수명주기에서 중요도 우선순위에 의하여 품질특성을 선별할 수 있으며, 소프트웨어 품질을 정량적으로 측정하고 개발 완료 후의 소프트웨어의 품질을 예상할 수 있게 하여 고품질의 소프트웨어를 획득할 수 있게 한다.

A Weight Decision Technique of Quality Characteristics in Software Development Life Cycle

Jae-Kyu Cho* · Kil-Sup Lee** · Sung Jong Lee**

ABSTRACT

The concern and request of high-quality software has been increased according to the rapid transition of society to information-oriented era. Therefore activities of software quality management are regarded as serious as the activities of software development. Moreover, stakeholder of software may have various views of software quality. And some quality characteristics need to be managed with priorities in Software Development Life Cycle (SDLC). Hence we recognized that weights of quality characteristics might be useful to improve efficiency of quality management activities. However, this topic has not actively issued because its necessity and effectiveness has been ignored. This paper presents a technique to decide weights of software quality characteristics in SDLC. It is based on ISO/IEC 9126 which is the standard for software product quality. It is accomplished by questionnaires intended for stakeholder in software development and weight analysis through AHP (Analytic Hierarchical Process) method. Accordingly, it can select quality characteristics by weight in SDLC. And then it helps us to acquire high quality software through measuring software quality quantitatively and expecting the quality after completion of software development.

키워드 : 소프트웨어 품질(Software Quality), 품질특성의 중요도(Weights of Quality Characteristics), 소프트웨어 수명주기(Software Development Life Cycle)

1. 서 론

오늘날 사회는 정보화가 급속하게 이루어지면서 국가기관, 민간기업 및 개인이 실생활에서 직접 또는 간접적으로 소프트웨어에 의존하는 경우가 크게 증가하고 있다. 이에 따라 고품질 소프트웨어에 대한 요구와 관심이 증대되면서 소프트웨어의 개발, 운용, 사용 및 유지보수에 따른 위험요소의 식별이 점점 더 요구되고 있다. 그리하여 소프트웨어 개발에 수반되는 위험요소로 확대가 예상되는 결함사항을

사전에 식별하고 평가하여 소프트웨어 사용에 대한 신뢰성과 안전성을 도모하는 소프트웨어 품질관리 및 품질보증은 소프트웨어 개발과 함께 중요한 활동이 되고 있다[1].

기존의 소프트웨어 개발은 과거의 축적된 소프트웨어 평가경험, 지식의 공유 및 평가기준과 절차에 대한 표준화 노력이 미흡하여 관련 소프트웨어를 개발하는데 있어서 중복된 노력과 비효율적인 작업이 존재하였다. 그리고 품질평가를 통한 품질보증에 대한 모호한 기준으로 평가수행 이후 결함사항에 대하여 객관적인 기준이 제시되어야 함에도 정성적인 기준에 의한 평가로 인하여 품질평가의 객관성 및 신뢰성 확보에 한계가 있었다.

* 준 회원 : 육군지도장

** 정 회원 : 국방대학교 전산정보학과 교수

논문접수 : 2004년 6월 4일, 심사완료 : 2004년 7월 28일

양질의 소프트웨어를 확보하여 시스템의 안정성과 신뢰성을 높이고 유지보수 노력을 최소화하여 경제성을 높이기 위해서는 개발활동의 전체 단계에서 품질평가 및 보증활동이 이루어져야 한다[2]. 초기 계획단계에는 프로젝트의 특성과 품질 요구사항을 철저히 파악하여 품질목표를 설정하고, 수명주기에서는 품질목표의 충족여부를 면밀히 점검하여야 한다. 왜냐하면 소프트웨어는 개발이 완료된 상태에서의 시험평가만으로 그 신뢰성과 안정성을 보장할 수 없고, 일단 개발이 완료된 소프트웨어를 개선하기 위하여 수정 혹은 변경하는데 개발에 투입된 이상의 시간과 비용이 소요되기 때문이다.

소프트웨어 평가 프로세스는 제품을 평가하기 위하여 품질 목표, 품질 모형, 품질 특성 및 척도를 포함한다. 이중 품질 모형에 관한 연구는 Boehm[3], McCall[4] 및 Evans[5] 등을 중심으로 발전이 되었다. 그렇지만, 현재에는 ISO/IEC 9126[6]에서 정의하고 있는 소프트웨어 품질 모형이 널리 활용되고 있는 상황이다.

ISO/IEC 9126에서는 소프트웨어의 품질을 제품 품질(Product Quality)과 프로세스 품질(Process Quality)로 크게 두 가지로 구분하고 있다. 일반적으로 소프트웨어의 품질을 말할 때에는 제품 품질과 프로세스 품질을 혼용하여 사용하기 때문에 명확한 구분이 필요하다.

소프트웨어 제품 품질은 ISO/IEC 9126에서 정의되어 있는 소프트웨어 개발과정에서 만들어진 중간 산출물의 속성 또는 실행가능한 제품의 속성에 대한 사용자의 요구 충족도로 설명된다. 이에 반해 프로세스 품질은 ISO/IEC 12207[7] 소프트웨어 수명주기 프로세스에서 정의된 개발지원, 조직 등을 포함한 프로세스의 품질을 말하며 정의된 프로세스, 산출물, 정량적인 관리 및 최적화 등과 관련된다.

소프트웨어 제품 품질평가의 기준이 되는 ISO/IEC 9126에서는 품질 특성과 부특성에 대해 정의하고 있으며, 이에 관련된 평가 척도를 제시하고 있다. 그러나 품질특성 및 부특성에 대한 중요도가 제시되지 않아 품질 평가 및 관리에 있어서 모든 특성들을 동등하게 처리함으로써 수명주기에서 획득관리자, 개발자, 사용자 및 유지보수자 등 소프트웨어에 대한 이해당사자들이 고려하는 품질특성의 중요도가 반영되지 못하였다. 예를 들면, 재무 프로그램에서 정확성과 신뢰도가 특히 중요하게 요구되는 데에도 다른 품질특성들과 대등하게 관리되어 사용자의 요구를 충족하지 못하거나 덜 중요한 품질특성을 충족하기 위하여 노력을 낭비하는 경우이다.

따라서 본 논문에서는 소프트웨어 수명주기에서 이해당사자들이 가지고 있는 품질특성의 중요도를 결정하는 과학적인 기법에 대하여 연구하고자 한다. 제시된 기법은 일반적인 자료처리 위주의 응용 소프트웨어를 대상으로 실시하였으며 대상에 따라 별도의 설문대상을 선정하고 분석을 통하여 품질특성의 중요도를 구할 수 있다. 유사한 소프트웨어

의 경우에는 별도의 설문조사를 생략하고 본 논문에서 제시된 중요도를 조정하여 활용함으로써 품질평가의 신뢰성을 향상을 기대할 수 있을 것이다.

본 논문의 구성 및 전개는 2장에서 소프트웨어 품질특성 및 이들 간의 관계를 정립한 품질 모형, 이들 품질 모형을 기반으로 한 품질평가 절차에 대하여 알아본다. 3장에서는 설문조사와 AHP[8]를 이용하는 소프트웨어 품질특성의 중요도를 결정하는 기법을 제시한다. 그리고 4장에서는 결정된 품질특성의 중요도를 활용 및 분석결과를 기술한다. 마지막으로 5장에서는 본 연구의 결과를 요약하고 향후 연구 방향을 제시함으로써 결론을 맺는다.

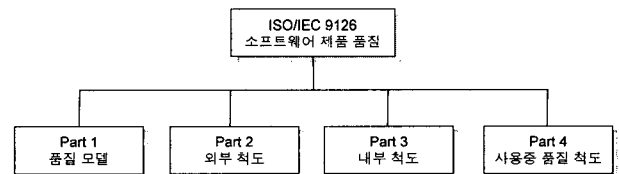
2. 관련 연구

2.1 소프트웨어 품질 모형

소프트웨어 품질을 평가하기 위해서는 우선적으로 소프트웨어 제품에 요구되는 품질특성과 이들에 대한 평가를 위한 평가방법 및 척도를 포함한 평가기준이 필요하다. 소프트웨어 품질은 하드웨어와 달리 보는 사람의 관점에 따라 다르게 해석되는 경우가 많기 때문이다. 소프트웨어 품질특성의 정의 및 평가척도를 위한 품질모형에 관한 연구는 Boehm[3], McCall[4] 및 Evans[5] 등을 통하여 발전이 되었다.

Boehm은 제품 품질을 정량적으로 측정 가능한 품질 척도를 최초로 발표하였고, 이에 따른 품질모형을 제시하였다. McCall은 요구된 제품 품질을 개발과정 동안에 제공되는 개발노력으로 요구된 품질수준을 얻을 수 있을지의 여부를 판단하기 위한 지침을 제시하였다. Evans는 제품 품질을 소프트웨어 개발단계를 중심으로 구분하였다.

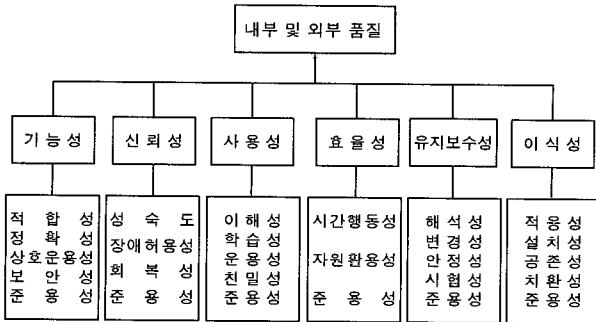
그러나 현재 국제적으로 사용의 효과성을 인정받고 있는 품질평가 모형으로는 ISO/IEC 9126[6]을 들 수 있다. ISO/IEC 9126은 사용자 관점에서 본 소프트웨어의 품질인자에 대한 국제표준으로 소프트웨어 품질을 객관적이고 계량적으로 평가할 수 있는 기본적 틀을 제공해 주기 위해 제정되었다. (그림 1)은 ISO/IEC 9126의 구조를 나타낸 것이다.



(그림 1) ISO/IEC 9126의 구조

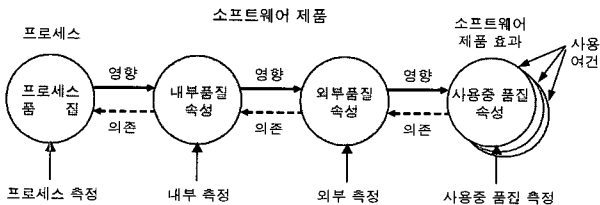
ISO/IEC 9126은 소프트웨어 평가를 위한 품질특성과 사용을 위한 지침으로 소프트웨어 품질을 품질특성으로 구분하고, 이를 다시 품질 부특성으로 세분화한 품질모형은 제시한다. 이러한 부특성은 소프트웨어가 컴퓨터 시스템의 일부로 사용될 때 외부로 나타나며, 내부적인 소프트웨어 속성들

의 결과이다. ISO/IEC 9126의 품질특성을 정리하면 (그림 2)와 같다.



(그림 2) ISO/IEC 9126 소프트웨어 품질특성

ISO/IEC 9126에서는 소프트웨어 수명주기에 따른 품질을 (그림 3)과 같이 제시하고 있다. 여기서 프로세스 품질은 ISO/IEC 12207[7]의 소프트웨어 수명주기내 공정의 품질을 나타내는 것이고, 제품 품질은 내부 또는 외부 속성의 측정치에 의해 평가되는 품질이다. 사용 품질은 특정 환경 및 여건에서 사용되는 제품의 사용자 관점의 품질이다.



(그림 3) 소프트웨어 수명주기에 따른 품질

이러한 품질간의 관계에서 프로세스 품질은 제품 품질의 개선을 가져오고, 제품 품질은 사용품질의 개선을 가져온다. 그러므로 프로세스를 평가하고 개선하는 것이 제품 품질을 개선하는 한 방법이며, 제품 품질을 평가하고 개선하는 일은 사용 품질을 개선하는 방법이다.

소프트웨어 제품을 평가하는 것은 소프트웨어 개발 수명주기의 한 과정이다. 소프트웨어 제품 품질은 내부속성 또는 외부속성에 의해 평가될 수 있다. 내부속성은 통상 중간 산출물의 정적인 속성을 나타내며, 외부속성은 실행시 코드 행위의 속성을 나타낸다. 내부 및 외부 속성을 평가하기 위해 ISO/IEC9126에서는 평가척도(metric)를 제시하고 있다. 내부 평가척도는 설계나 구현 중에 실행할 수 없는 명세서나 원시코드 등 소프트웨어 중간 산출물에 적용할 수 있고, 외부 평가척도는 실행 가능한 소프트웨어나 시스템을 시험, 운영 또는 관찰함으로써 적용할 수 있다.

2.2 소프트웨어 품질평가 절차

소프트웨어 제품 평가에 대한 표준인 ISO/IEC 14598[9]에서는 품질평가 절차를 평가요구사항 설정단계, 평가명세 단

계, 평가계획 설계단계, 평가수행단계로 구분하고 있다.

평가 요구사항 설정단계에는 품질특성 및 이용 가능한 하부 특성들을 사용하여 품질 요구수준을 정의하는데, 이는 개발 이전에 반드시 정의되어야 하는 것이다. 소프트웨어 제품이 구성요소로 분해 될 때, 전체 제품을 대상으로 선정된 품질 요구사항이 각 구성요소별로는 다를 수 있음을 인식하여야 한다.

평가명세 단계에는 품질 요구사항을 측정할 수 있는 평가척도를 준비하여야 한다. 이 때 소프트웨어 제품의 성질뿐만 아니라 환경과의 상호작용에 대한 척도도 함께 준비하여야 한다. 평가척도를 사용하여 측정된 값이 어느 등급에 속하는지에 대한 품질등급 및 품질수준 기준이 설정되어야 하며, 최종적인 판정기준도 사전에 정의되어야 한다.

평가계획 설계단계에는 평가계획을 수립하고 마지막으로 평가수행 단계에는 실제로 측정하고 등급을 부여하며 수용 또는 기각 등의 판정을 내리게 된다. 측정이라 함은 선정된 평가척도를 소프트웨어 제품에 적용하는 것이며, 판정이란 개별적으로 평가된 품질특성들을 총체적인 관점에서 심사하여 품질수용 여부를 결정하는 것이다.

3. 소프트웨어 수명주기별 중요도 결정 기법

3.1 자료수집

본 연구를 위한 중요도 설정을 위해 설문을 통한 실증분석을 실시하여 그 결과를 제시하고자 한다. 자료수집 대상 기관은 사용자 관점에서의 품질특성 중요도를 조사하기 위해 소프트웨어 발주기관 및 연구소를, 그리고 개발자 관점에서의 품질특성 중요도를 조사하기 위해 소프트웨어 개발업체 및 정부기관내 개발부서를 선정하였다. 대상자는 대상기관의 소프트웨어 사용자, 획득 및 프로젝트 관리자, 응용 프로그래머, 유지보수 담당자 등 총 50명을 선정하여 설문조사를 실시하였다.

설문대상은 자료의 객관성을 확보하기 위해 소프트웨어 수명주기 전체에 관여하는 인원을 다양한 기관에서 선정하였다. 본 설문에 참여한 인원은 획득관리자가 17명(34%), 개발자가 12명(24%), 사용자 11명(22%), 유지보수자 10명(20%)으로 구성되었다. 또한, 설문의 신뢰성과 전문성을 뒷받침하기 위해 전산관련 경력도 10년 이상 되는 인원이 13명(26%), 7~9년이 11명(22%), 3~6년이 14명(28%), 3년 미만인 12명(24%)으로 골고루 분포되게 선정하였다.

설문대상 인원들의 소프트웨어 품질평가의 중요성에 대한 인식과 품질특성을 이용한 품질평가에 대한 인식을 조사해 본 결과, 44명(88%)의 응답자가 소프트웨어 품질평가에 대해 중요하다고 평가하고 있었으며, 37명(74%)의 응답자가 품질특성을 이용한 품질평가가 적절하다고 평가하고 있었다. 또한 ISO/IEC 9126에 대한 이해도 조사에서도 43명(86%)의 응답자가 대체로 이해하고 있는 것으로 분석되었다.

따라서 위와 같이 설문인원에 대한 일반적인 특성을 분석한 결과를 토대로 본 논문에서 조사한 소프트웨어 수명주기에서 품질특성의 중요도 분석결과는 전문성과 신뢰성이 있다고 할 수 있다.

3.2 AHP를 이용한 자료분석

설문결과와 분석하여 수명주기별 중요도를 결정을 위하여 AHP(Analytic Hierarchy Process)[8] 도구인 Expert Choice¹⁾를 이용하였다. 기타 설문자의 구성 및 경력, 품질평가에 대한 일반적인 사항은 MS-Excel을 사용하였다. 특히 AHP 방법을 이용하면 표본의 수와 관계없이 신뢰성과 일관성을 평가할 수 있는 장점이 있다.

계층화 의사결정기법 AHP는 1980년 T. L. Saaty에 의해 처음 소개되었으며, 문제를 계층적으로 분석하여 이원비교(pair-wise comparison)에 의한 판단을 통하여, 여러 개의 목적을 동시에 가진 문제에 대한 의사결정 기법이다. 또한 실무적인 경험이 많은 전문가의 의견을 쉽게 도출하고 체계적으로 분류할 수 있으며, 객관적인 평가요인은 물론 주관적인 평가요인도 수용하는 매우 유용한 의사결정 기법으로 주요 특징은 다음과 같다.

첫째, 정량적인 방법으로 문제를 해석하기 때문에 이해가 용이하고 결과가 명확하다. 둘째, 복잡하고 불분명한 문제에 대해서는 여러 계층으로 분리하여 각 요소를 1:1로 비교하여 상대적 중요도를 분석함으로써 문제를 간결하게 한다. 셋째, 시스템 접근을 통해 주관적인 판단을 하고, 이를 종합함으로써 경험을 살린 객관적인 의사결정을 할 수 있다. 넷째, 일관성 비율 평가로 전문가의 판단을 검증할 수 있다.

AHP를 이용하여 품질특성의 중요도를 결정하기 위하여 다음과 같은 4단계의 절차를 적용하였다.

• 단계 1: 문제의 계층 분화

AHP 기법의 가장 중요한 단계로서 주어진 의사결정 문제를 상호 관련된 의사결정요소들로 계층화 하여 분리하는 과정이다. 계층의 최상위에는 가장 포괄적인 의사결정문제의 목표가 놓여지며, 그 다음 계층에는 의사결정 목표에 영향을 미치는 다양한 속성들로 구성된다. 이들 속성은 낮은 계층일수록 세부적이 되며, 계층의 최하위층은 선택의 대상이 되는 대안들로 구성된다.

• 단계 2: 평가기준의 이원비교 행렬 작성

같은 수준에 있는 요소를 대상으로 이원비교를 실시한다. 속성 *i*의 중요도를 *w_i*, 속성 *j*의 중요도를 *w_j*, 속성 *j*에 대한 속성 *i*의 상대적 비중을 *a_{ij}* = *w_i*/*w_j*라고 정의하여 비교행렬 *A*를 작성한다.

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{21} & \dots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix}$$

이 때 일반적으로 많이 사용되는 평가척도는 Saaty가 제안한 9점 척도이다. 이러한 평가척도는 <표 1>과 같다.

<표 1> AHP에서 사용되는 평가척도

수치 값	정 의
1	◦ 동등하게 중요하거나 선호되는 경우 (Equal importance)
3	◦ 약간 더 중요하거나 선호되는 경우 (Somewhat more important)
5	◦ 강하게 더 중요하거나 선호되는 경우 (Much more important)
7	◦ 아주 강하게 더 중요하거나 선호되는 경우 (Very much more important)
9	◦ 극도로 더 중요하거나 선호되는 경우 (Absolutely more important)
2, 4, 6, 8	◦ 절충하기 위한 조정값 (Intermediate value)
역의 값	◦ 첫 번째 비교하여 두 번째 대안이 우월한 경우로 1/9, 2/9, ..., 8/9로 표기

• 단계 3: 중요도 결정 및 일관성 확인

2단계에서 이원비교를 통하여 얻은 *a_{ij}* 값을 이용하여 평가기준 *C₁, C₂, ..., C_n*이 갖는 중요도를 나타내는 *W₁, W₂, ..., W_n*을 정하는 것이다. Saaty는 이러한 중요도 추정방법으로 다음의 관계식을 이용하는 고유치 방식을 제안하였다.

$$A \cdot W = \lambda_{max} \cdot W$$

여기서 *A*는 이원비교의 결과로 얻어진 정방행렬이며, λ_{max} 는 *A*의 최대고유치, *W*는 고유벡터이다.

또한, 주어진 중요도에 대하여 논리적 일관성이 있는가를 조사하기 위해서는 일관성 비율 CR(Consistency Ration)을 계산해야 한다. 일관성을 조사하기 위한 일관성 비율의 계산식은 다음과 같다.

$$CR = CI / RI$$

여기서 일관성 지수 CI(Consistency Index)는 아래와 같이 계산할 수 있다.

$$CI = (\lambda_{max} - n) / (n - 1)$$

무작위 지수 RI(Random Index)는 평가기준의 개수 *n*의 크기에 따라 <표 2>와 같은 값으로 나타낸다.

<표 2> RI의 값

<i>n</i>	1	2	3	4	5	6	7	8
RI	0.00	0.00	0.58	0.90	1.12	1.24	1.32	1.41

1) Expert Choice와 MS-Excel은 AHP와 스프레드시트를 위한 도구의 상품이다.

계산한 결과로서 CR값이 0.1이내이면 이원비교는 합리적인 일관성을 갖는 것으로 판단하고, 0.2이내일 경우에는 수용할 수 있으나, 그 이상이면 일관성인 부족한 것으로 판단한다[8].

• 단계 4 : 중요도 종합

이전 2,3단계에서 계산된 평가기준들의 중요도를 종합하는 과정이다. 즉, 상위계층에 있는 의사결정문제의 궁극적인 목표를 달성함에 있어서 하위계층에 있는 평가기준들이 어느 정도 영향을 미치는지 또는 어느 정도의 중요성을 갖고 있는지를 알아보기 위해 평가기준들의 종합 중요도를 구하는 단계이다. 평가기준의 종합 중요도는 아래와 같이 구할 수 있다.

$$W_i = \sum(w_j) \cdot (r_{ij})$$

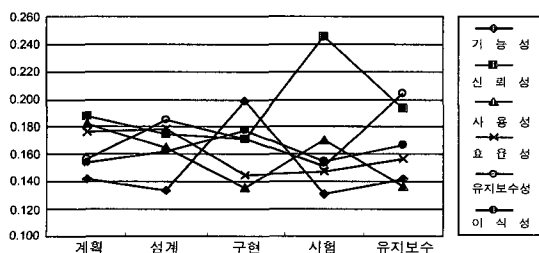
여기서 W_i 는 i 번째 요소의 종합 중요도, w_j 는 평가기준 j 의 상대적 중요도, r_{ij} 는 평가기준 j 에 대한 i 번째 요소의 중요도이다.

3.3 품질특성의 중요도 분석 결과

AHP를 이용하여 실시한 설문분석 결과, 소프트웨어 수명주기와 관련한 각 인원 관점에서 수명주기별 직무별 품질특성의 중요도는 다음과 같다.

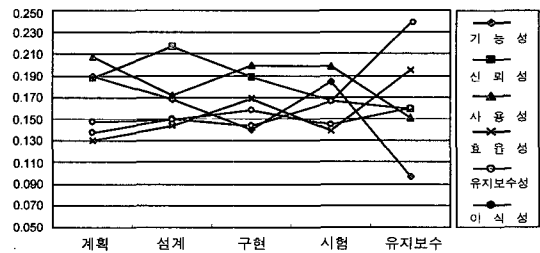
획득관리자의 관점에서 보는 소프트웨어 수명주기별 품질특성의 중요도는 (그림 4)와 같다. 획득관리자들은 개발 전 단계에 걸쳐 신뢰성이 아주 중요한 품질특성으로 인식하고 있고 특히 시험단계에서는 다른 품질특성 비해 아주 높은 비율을 차지하고 있다.

획득관리자들은 소프트웨어 품질과 함께 개발비용과 시간 등이 아주 중요한 요소로 보고 있기 때문에 각 단계에서 이러한 비용과 시간에 영향을 줄 수 있는 품질특성에 대해 집중적으로 관리하는 것으로 나타났다. (그림 4)에서 보는 바와 같이 구현단계에서는 소프트웨어 요구사항에서 분석된 기능이 모두 포함되어 구현되었는지를 나타내는 기능성을 아주 중요한 품질특성으로 보고 있고, 시험단계에서는 신뢰성을, 설계단계 및 유지보수단계에서는 보수성을 중요한 품질특성으로 보고 있다.



(그림 4) 획득관리자 관점의 품질특성 중요도

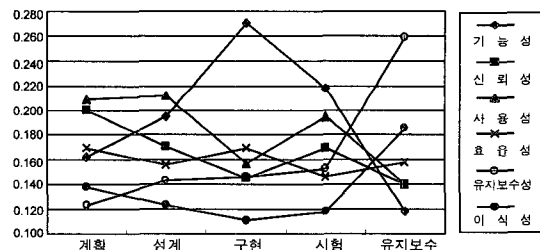
개발자의 관점에서 보는 소프트웨어 수명주기별 품질특성의 중요도는 (그림 5)와 같다. 개발자들은 개발 전 단계에 걸쳐 신뢰성과 사용성을 아주 중요한 품질특성으로 보고 있다.



(그림 5) 개발자 관점의 품질특성 중요도

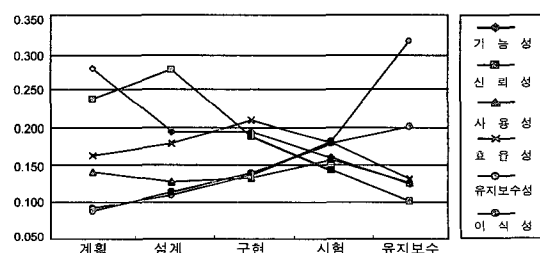
개발자는 소프트웨어 최종품질뿐만 아니라 수명주기 중간에서 생성되는 단위모듈, 통합모듈, 프로토타입 제품의 품질에 대해서도 중요하게 생각한다. 따라서 구현단계와 유지보수 단계에서 자원의 사용량을 나타내는 효율성을 중요한 품질특성으로 보고 있으며, 대부분의 개발자가 직접 시스템 전이 및 유지보수를 담당하는 경향에 따라 유지보수단계에서는 보수성과 이식성을 중요한 품질특성으로 인지하고 있다.

사용자의 관점에서 보는 소프트웨어 수명주기별 품질특성의 중요도는 (그림 6)에 나타나고 있다.



(그림 6) 사용자 관점의 품질특성 중요도

사용자는 소프트웨어 내부의 품질에는 무관심하고 사용성이나, 요구단계에서 제시한 사용자 요구사항의 구현여부에 관심이 많다. 중요도 조사결과에서도 기능성과 사용성이 수명주기 전체에서 아주 중요한 품질특성으로 선정하였다. 다만, 유지보수단계에서는 이미 검증된 기능성 및 사용성 보다는 유지보수성과 이식성을 더 중요한 품질특성으로 선정하고 있다. 또한 성능에도 관련된 특성인 효율성도 각 단계에서 중요한 품질특성으로 나타났다.



(그림 7) 유지보수자 관점의 품질특성 중요도

(그림 7)은 유지보수자의 관점에서 보는 소프트웨어 수명주기별 품질특성의 중요도를 나타낸 것이다.

유지보수자는 변경된 요구사항을 수정하고 개량하는 임무를 가진 특성으로 인해 계획, 설계, 구현단계에서는 기능성, 신뢰성, 및 효율성을 중요한 품질특성으로 인지하고 있다. 즉 사용자 요구사항에 대한 추적사항과 유지보수에 큰 영향을 미치는 시스템 성능, 사용자 교육 등에 많은 관심을 가지고 있다. 시험 및 유지보수단계에서는 보수성과 이식성을 아주 중요한 품질특성으로 보고 있다. 소프트웨어 개선에 따른 노력과 예상되는 위험성에 대해 관심을 가지고 있으며, 소프트웨어 변경에 의해 하드웨어 환경에 영향을 얼마나 미치는지에 대해서도 고려하고 있다.

위와 같은 결과를 토대로 전체 인원에 대해서 소프트웨어 각 수명주기별 중요도를 종합하면 <표 3>과 같다.

<표 3> 품질특성의 중요도

구 분	계획단계	설계단계	구현단계	시험단계	유지보수 단계
기능성	0.169	0.164	0.211	0.165	0.126
신뢰성	0.200	0.193	0.168	0.197	0.159
사용성	0.188	0.174	0.151	0.182	0.140
효율성	0.167	0.168	0.165	0.151	0.160
보수성	0.136	0.158	0.156	0.158	0.238
이식성	0.140	0.143	0.149	0.146	0.177

계획단계는 소프트웨어 개발 대상인 업무체계의 통합과 사용자에 대한 정확한 이해가 중요 관리요소이다. 또한, 업무기능이 얼마나 적절하고, 구체적으로 문서화되는지를 관리하는 것도 중요한 요소이다. 따라서 분석결과에서 신뢰성, 사용성, 기능성이 중요한 품질특성으로 조사되었다.

설계단계는 시스템 기능을 구조화하고 상세화하는 것과 주어진 자원 조건하에서 최대의 효율발휘가 중요관리요소이다. 신뢰성, 사용성, 효율성이 중요한 품질특성으로 조사되었다.

구현단계는 요구기능의 설계내용에 충분한 이행, 사용자에 영향을 미치는 각종 처리시간 및 자원관리, 유지보수에 대한 준비가 중요관리요소이다. 기능성, 신뢰성, 효율성이 중요한 품질특성으로 조사되었다.

시험단계는 사용자에 의해서 요구된 기능이 얼마나 정확히 동작하는지, 소프트웨어 운영 중 우발상황에 대한 대처는 적절한지, 명세된 기능이 준비된 시간과 자원을 사용하는지가 중요관리요소이다. 신뢰성, 사용성, 기능성이 중요한 품질특성으로 조사되었다.

마지막으로 유지보수단계는 새로운 소프트웨어에 대한 사용자의 교육과 이해와 새로운 환경에서의 소프트웨어의 적용 등이 중요한 관리요소로 나타난다. 이와 관련하여 보수성, 이식성, 효율성이 중요한 품질특성으로 조사되었다.

한편, 각 수명주기별로 품질 부특성의 중요도를 산출하는

방법은 다음과 같다. 먼저, 같은 품질특성을 가지는 품질 부특성에 따라 이원비교를 실시한다. 이원비교를 실시하여 얻은 품질 부특성의 중요도를 앞 절에서 산출한 품질특성의 중요도에 따라 배분하면 최종 중요도를 산출할 수 있다. <표 4>는 소프트웨어 수명주기별 품질 부특성의 중요도를 나타낸 것이다.

<표 4> 수명주기별 품질 부특성의 중요도

품질 특성	품질 부특성	계획 단계	설계 단계	구현 단계	시험 단계	유지보수 단계
기능성	정확성	0.038	0.045	0.063	0.045	0.029
	상호운용성	0.044	0.040	0.055	0.046	0.034
	표준적합성	0.050	0.040	0.052	0.039	0.032
	보안성	0.037	0.039	0.041	0.035	0.031
신뢰성	성숙성	0.078	0.055	0.056	0.054	0.052
	장애허용성	0.068	0.071	0.058	0.064	0.053
	회복성	0.054	0.067	0.054	0.079	0.054
사용성	이해성	0.050	0.040	0.035	0.039	0.031
	습득성	0.044	0.041	0.033	0.037	0.034
	운용성	0.047	0.050	0.046	0.061	0.04
	친밀성	0.047	0.043	0.037	0.045	0.035
효율성	시간행동성	0.090	0.080	0.072	0.078	0.074
	자원활용성	0.077	0.088	0.093	0.074	0.086
보수성	해석성	0.035	0.034	0.026	0.028	0.051
	변경성	0.031	0.038	0.033	0.036	0.065
	안정성	0.035	0.047	0.049	0.048	0.063
	시험성	0.035	0.039	0.048	0.046	0.059
이식성	환경적응성	0.037	0.037	0.037	0.037	0.043
	설치성	0.029	0.03	0.038	0.034	0.040
	공존성	0.039	0.039	0.038	0.039	0.051
	치환성	0.035	0.037	0.036	0.036	0.043

3.4 설문지의 일관성 분석

분석된 설문자료는 일관성과 신뢰성이 요구된다. 만약, 설문자료의 신뢰성과 일관성이 부족하다면 새로운 자료를 수집해야 할 것이다. AHP에서는 주어진 중요도에 대하여 논리적 일관성이 있는가를 조사하기 위해서는 일관성 비율 CR(Consistency Ratio)을 계산할 수 있다. 본 연구에서는 AHP에서 요구하는 일관성 비율 0.1을 모두 만족하였다. 따라서 설문자료에 대한 신뢰성은 보장된다고 할 수 있다. <표 5>는 설문자료에서 각 단계별 이원비교에서 일관성 비율의 값을 나타내고 있다.

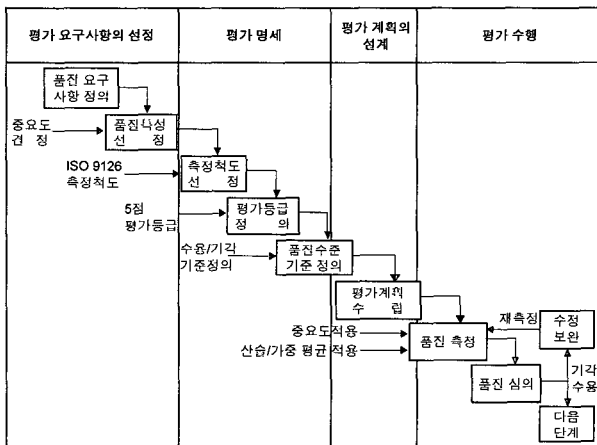
<표 5> 설문자료의 일관성 비율

구 분	분석 단계	설계 단계	구현 단계	시험 단계	유지보수 단계
CR	0.01	0.01	0.01	0.01	0.01
비 고	만족	만족	만족	만족	만족

4. 품질특성의 중요도 활용 분석

4.1 품질평가 절차

2.2절에서 설명된 품질평가 절차는 ISO/IEC 14598에서 제시되었다. 그러나 여기서 제시된 품질평가 절차를 본 연구에 적용하기에는 세부적인 사항이 미흡한 편이다. 따라서 본 연구에서는 ISO/IEC 14598의 품질평가 절차에 품질특성의 중요도를 반영하기 위하여 (그림 8)과 같이 테일러링된 세부적인 절차를 제시하였다. 제시된 세부 절차는 실제 적용 보다는 품질특성의 중요도가 품질평가에 어떠한 모습으로 적용되는지를 증점적으로 설명하고 있다. 품질특성의 중요도를 반영한 평가절차는 평가 요구사항의 설정, 평가 명세, 평가 계획의 설계 및 평가 수행으로 구성된다.



(그림 8) 품질특성의 중요도를 반영한 평가절차

먼저, 평가 요구사항 설정 단계에서는 품질요구 수준의 정의와 평가대상 품질특성을 선정하는 세부 단계로 이루어진다. 이 중 품질요구 수준은 소프트웨어 개발전 개발에 참여하는 인원의 합의에 의해 정해질 수 있다. 그리고 평가대상 품질특성의 선정 단계를 위하여 본 논문에서는 개발의 세부 단계에서 평가가 가능한 품질특성 중에서 시간과 비용을 절감하기 위해서 중요도 우선순위에 의해 선정하는 방법을 제시한다. 이에 대한 자세한 사항은 4.2절에서 언급되었다.

평가 명세 단계에서는 선정된 품질특성에 의해 단계별 적용 가능한 내부 평가척도 또는 외부 평가척도를 선정한다. 이어서 척도에 의한 산정된 값에 따른 분류를 위한 품질등급의 기준을 정하며, 수용 또는 기각을 위한 품질 수준 기준을 정한다.

품질평가계획 설계 단계에서는 앞서 정의된 평가 요구사항 및 평가 명세를 기반으로 평가 장소, 일정, 평가자 등 평가수행을 제반 계획을 수립한다. 평가수행 단계에서는 각 품질특성의 중요도를 반영한 품질평가 척도에 의하여 개별 특성의 품질을 측정하고 4.3절에서 설명되는 중요도에 의한 품질측정 방법에 의하여 전체 품질수준을 평가한다. 이 결

과를 최초 정의된 품질 수준과 비교하여 적합 수준이면 개발의 다음 단계의 활동으로 넘어가고 부적합 경우에는 수정 및 보완 후 재평가를 하게 된다.

4.2 중요도에 의한 평가대상 품질특성의 선정

소프트웨어 각 수명주기에서 모든 품질특성에 따른 평가 척도를 평가한다는 것은 많은 비용과 노력이 소요된다. 따라서 소프트웨어를 개발하기 전에 품질목표를 설정하고, 그 품질목표에 부합되는 품질특성에 대해서만 평가한다면, 비용과 노력은 적게 들이면서 보다 높은 품질의 소프트웨어를 획득할 수 있을 것이다.

본 연구에서는 품질목표를 3.5/5점(70%)을 적합한 기준으로 설정하였다. 그러므로 각 수명주기별 품질 부특성을 중요도에 의해 우선순위를 정하고, 그 누적치가 70%되는 내부 특성을 단계별 평가특성으로 선정하고자 한다.

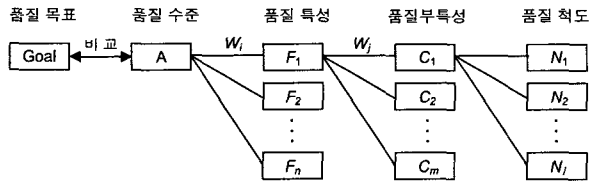
<표 6> 수명주기별 평가대상 품질 부특성

구분	계획단계	설계단계	구현단계	시험단계	유지보수 단계
1	시간행동성	자원활용성	자원활용성	회복성	자원활용성
2	성숙성	시간행동성	시간행동성	시간행동성	시간행동성
3	자원활용성	장애허용성	정확성	자원활용성	변경성
4	장애허용성	회복성	장애허용성	장애허용성	안전성
5	회복성	성숙성	성숙성	운용성	시험성
6	표준적합성	운용성	상호운용성	성숙성	회복성
7	이해성	안정성	회복성	안정성	장애허용성
8	친밀성	정확성	표준적합성	시험성	성숙성
9	운용성	친밀성	안정성	상호운용성	해석성
10	습득성	습득성	시험성	친밀성	공존성
11	상호운용성	표준적합성	운용성	정확성	환경적응성
12	공존성	이해성	보안성	표준적합성	치환성
13	정확성	상호운용성	설치성	이해성	운용성

소프트웨어 수명주기별 평가대상 품질 부특성은 각각 13개 특성이 <표 6>과 같이 선정되었으며, 특히 시간행동성, 성숙성, 자원활용성, 장애허용성, 회복성, 운용성 등 6개 품질 부특성은 소프트웨어 수명주기 전체에서 평가대상 품질 부특성으로 선정되어 아주 중요한 품질 부특성으로 나타났다. 이런 결과는 소프트웨어 개발시 13개 특성에 대해 집중적으로 확인하고 검증하면 적은 비용과 시간을 소모하고 좋은 품질의 소프트웨어를 획득할 수 있을 것으로 예상된다.

4.3 중요도에 의한 품질측정 방법

본 절에서는 개별 품질특성의 측정값을 중요도를 반영하여 전체 측정값으로 환산하는 과정을 설명하고 있다. 먼저, 각 품질특성 및 품질 부특성간의 관계 및 중요도는 (그림 9)와 같이 나타낼 수 있다.



(그림 9) 품질특성 및 품질 부특성간의 관계

(그림 9)에 의해 각 단계별 평가항목에 대한 평가점수는 다음과 같이 산정할 수 있다. 먼저 품질 부특성이 획득한 평점 C_j 는 식 (1)을 이용해서 계산된다. 여기서 평가등급은 5단계 평가로 가정하며, l 은 품질부특성별 평가항목 수를 나타내고 N_l 은 평가항목별 평가점수를 나타낸다.

$$C_j = \frac{\sum_{l=1}^l N_l}{l} \quad (1)$$

품질특성 F_i 의 평가점수는 관련된 품질 부특성의 평가점수 C_j ($j = 1, 2, \dots, m$)에 대한 가중평균으로 식 (2)를 이용해서 계산한다. 여기서 m 은 품질 부특성의 수이고, W_j 는 품질 부특성 C_j 의 중요도이다.

$$F_i = \frac{\sum_{j=1}^m W_j \times C_j}{\sum_{j=1}^m W_j} \quad (2)$$

현 수명주기에서의 품질수준 A 는 품질특성 F_i ($i = 1, 2, \dots, n$)에 대한 가중평균으로 식 (3)을 이용하여 계산한다. 여기서 n 은 품질특성의 수이고, W_i 는 품질 부특성 F_i 의 중요도이다.

$$A = \frac{\sum_{i=1}^n W_i \times F_i}{\sum_{i=1}^n W_i} \quad (3)$$

마지막으로 평가된 현 단계 품질수준 A 값과 품질목표 달성수준 G 를 비교하여 다음 단계로 진행할 것인지, 아니면 이전 단계로 복귀할 것인지를 판단하게 된다.

4.4 중요도를 고려한 품질평가 사례

4.4.1 대상 시스템 개요

품질대상 시스템은 국내 SI 업체인 S, D, L, H사가 컨소시엄을 구성하여 정부기관에 납품한 물류관리 정보체계를 대상으로 하였다. 물류관리 정보체계는 사용부서에서 물류지원부서 및 정책 부서에 이르기까지 보급품 순환주기에 따라 소요, 획득, 저장 및 분배, 처리업무 수행이 가능하고, 보급관리의 목표를 달성할 수 있도록 과학적인 물류자원관리와

업무처리를 보다 효율적으로 수행하고, 물류관리자 및 관련 기관장에게 의사결정에 필요한 보급 정보를 제공하는 체계이다.

물류관리 정보체계는 각 부서가 통합된 자료를 공유 가능한 데이터베이스 시스템으로 구축되고, 사용자 중심의 개방형 구조를 지원하는 클라이언트/서버 구조로서 물류 전용 장비에서 운용되며, 개인용 컴퓨터, 주변장치 등으로 구성되는 하드웨어와 업무처리, 시스템지원, 사용자지원, 통신지원 등의 기능을 수행하는 소프트웨어와 각 제대의 장비를 연결하는 통신회선 및 접속장치 등 인프라넷망을 구성한다.

물류관리 정보체계는 각 단계별 산출물을 작성함으로써 사용자의 요구사항이 정확하게 반영되었는지를 품질관리 체계에 따라 품질관리활동을 실시하였다. 구성관리는 개발 전 과정을 통해 MIL-STD-498[10]을 준수하였으며 <표 7>과 같은 산출물을 작성하였다.

4.4.2 대상 시스템 품질평가 결과

설계단계에서는 소프트웨어 설계기술서 외 3종의 산출물에 대한 평가가 이루어졌으며 56개 항목이 평가되었다. 먼저, 설계단계에서 품질부특성이 획득한 평가점수는 <표 9>와 같다. 평가 항목수가 많은 부특성은 식 (1)을 사용하여 산정한다.

<표 7> 구성관리 산출물

수명주기	MIL-STD-498 DID
계획단계	SDP(Software Development Plan)
	STP(Software Test Plan)
요구분석 단계	OCD(Operational Concept Description)
	SSS(System/Subsystem Specification)
	SRS(Software Requirements Specification)
	IRS(Interface Requirements Specification)
설계단계	SSDD(System/Subsystem Design Description)
	SDD(Software Design Description)
	IDD(Interface Design Description)
	DBDD(Database Design Description)
구현/시험 단계	STP(Software Test Specification)
	STR(Software Test Report)
전이단계	SVD(Software Version Description)
	SUM(Software User Manual)
	SIOM(Software Input/Output Manual)
	SCOM(Software Center Operator Manual)
	COM(Computer Operation Manual)

품질 부특성이 획득한 평가점수를 식 (2)에 대입하면 품질특성에 대한 평가점수를 <표 8>과 같이 산정할 수 있다. 품질특성이 획득한 평가점수를 식 (3)에 대입하면 현 단계에서의 품질평가값 A (assessment value)를 획득할 수 있다.

$$A = \frac{\sum_{i=1}^n W_i \times F_i}{\sum_{i=1}^n W_i} = 3.89$$

이를 품질목표 달성수준 G로 환산하면 아래와 같다.

$$G = 3.89 \times 20 = 77.8 \text{ (점)}$$

대상 시스템에 대한 품질목표 달성수준 G는 사전 정의한 5단계 기준에 의해 설계단계에서의 품질수준은 보통으로 평가되어 다음 단계로의 진행하는데 필요한 최소요구 품질수준을 획득한 것으로 나타났다. 그러나 신뢰성 및 효율성에서 낮은 점수를 획득하였으므로, 프로젝트 관리상 시간 및 비용이 추가적으로 요구된다하더라도 부족한 부분에 대해 보완을 하고 다음 단계로 넘어가는 것이 전체적인 품질향상 및 프로젝트 성공에 도움이 될 것이다.

〈표 8〉 품질부특성의 평가점수

품질특성	품질 부특성	중요도	평가점수
기 능 성	정 확 성	0.045	3.4
	상호운용성	0.040	4.1
	표준적합성	0.040	4.6
신뢰성	보안성	0.039	4.4
	성숙성	0.055	3.5
	장애허용성	0.071	3.4
사 용 성	회복성	0.067	3.6
	이해성	0.040	4.6
	습득성	0.041	4.0
	운용성	0.050	4.0
효 율 성	친밀성	0.043	3.6
	시간행동성	0.080	3.3
보수성	자원활용성	0.088	3.5
	해석성	0.034	4.2
	변경성	0.038	4.3
	안정성	0.047	3.7
	시험성	0.039	4.4
이식성	환경적응성	0.037	4.4
	설치성	0.030	4.7
	공준성	0.039	4.6
	치환성	0.037	4.4

〈표 9〉 품질특성의 평가점수

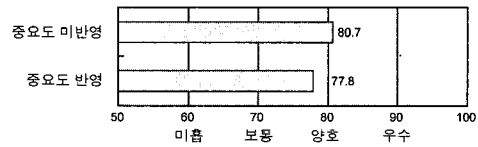
품질특성	중요도	평가점수
기 능 성	0.164	4
신뢰성	0.193	3.5
사 용 성	0.174	4
효 율 성	0.168	3.4
보수성	0.158	4.1
이식성	0.143	4.5

4.4.3 품질평가 결과 분석

본 논문에서 제시하는 중요도에 의한 평가방법은 두 가지 경우를 비교 및 분석하였다. 첫째는 중요도가 반영되지 않

은 경우와 반영된 경우의 평가결과를 비교하였다. 둘째는 중요도가 반영된 전체 품질특성의 평가결과와 중점관리 특성으로 선정된 품질특성의 평가결과를 비교하였다.

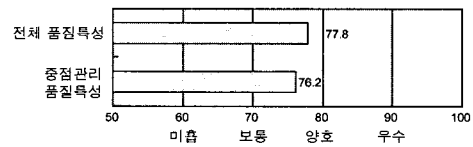
먼저, 중요도가 반영되지 않은 경우와 중요도가 반영된 경우의 평가결과는 (그림 10)과 같이 나타낼 수 있다. 품질특성에 중요도를 미반영한 경우 품질수준은 양호한 것으로 나타난 반면, 중요도를 반영한 경우에는 품질수준이 보통으로 평가를 받았다.



(그림 10) 중요도 반영여부에 따른 평가결과

이런 결과는 소프트웨어 품질평가시 품질특성간의 상관관계를 무시하고 평가를 실시할 경우 평가된 품질이 개발완료 후의 품질을 정확히 반영하지 못하여 개발완료 후 나타나는 오류를 수정하기 위한 많은 비용발생 또는 소프트웨어 개발의 실패를 초래할 수 있다. 따라서 소프트웨어 개발시 품질특성에 대해 중요도를 부여하여 개발함으로써 높은 중요도의 품질특성에는 많은 시간과 비용을, 낮은 중요도의 품질특성에는 적은 시간과 비용을 들여 효율적인 품질관리를 할 수 있을 것이다.

다음으로는 중요도가 반영된 전체 품질특성의 평가결과와 중점관리 특성으로 선정된 품질특성의 평가결과는 (그림 11)과 같이 나타난다.



(그림 11) 품질특성 적용범위에 따른 평가결과

중점관리 특성은 <표 6>에서와 같이 자원활용성 외 12개 특성이며 이들은 전체 품질특성의 70% 중요도를 가지고 있다. (그림 11)의 평가결과를 분석해 보면, ISO 9126에서 정의한 21개 품질 부특성 전체를 대상으로 품질 평가하고, 관리하는 것보다 중점관리 특성인 13개 특성을 대상으로 평가하고 관리하여도 전체 특성을 관리하는 것과 비슷한 효과를 얻을 수 있고, 보다 더 적극적으로 품질수준을 평가할 수 있다.

따라서 소프트웨어 개발간 품질평가 및 품질관리 활동을 효율성을 위해서는 중점관리 특성만 선정하여 관리하는 것이 개발완료 후의 품질수준을 향상시키고, 비용과 시간을 단축하는 좋은 방법이라 할 수 있다.

5. 결론 및 향후 연구과제

소프트웨어의 품질에 대한 관심이 높아지고, 대규모 소프

트웨어 개발에서 품질저하의 심각성이 대두됨으로써 품질보증 활동 및 품질평가를 통한 고품질 소프트웨어 개발의 중요성이 높아지고 있다. 따라서 품질평가에 있어서는 기존의 주관적인 평가방식을 과감하게 버리고 보다 더 객관적인 방법을 동원하여 실시해야 하며, 소프트웨어 개발초기부터 소프트웨어 품질특성을 고려하여 개발하고 단계별 품질관리를 철저히 하는 것이 중요하다.

본 연구에서는 소프트웨어 품질관리체계에 따라 소프트웨어 수명주기별 체계적이고 표준화된 품질평가를 실시하여, 발견된 오류에 대한 피드백을 실시하고 단계별 생산되는 산출물의 품질을 높이기 위해 품질특성의 중요도를 결정하여 품질평가에 이용하는 방법을 제시하였다. 품질특성은 국제적으로 인정받고 있는 ISO/IEC 9126 모형에 근거한 품질특성을 이용하였다.

중요도 결정은 객관적인 판단을 할 수 있도록 전문가 집단에 의한 AHP 방법을 사용하였다. 이를 통해 소프트웨어 개발에 참여하는 모든 인원의 관점을 바탕으로 품질특성, 품질 부특성에 관한 중요도를 제시하였다. 이러한 연구결과는 품질특성에 관한 존재하는 관계성 확인에 그치고 있는 기존의 소프트웨어 품질에 관한 연구에 비해 보다 더 정량적으로 품질특성간의 상관관계를 정립할 수 있고, 품질평가에 객관적인 판단기준인 중요도를 이용하는 발전된 기준이 될 것으로 본다.

그러나, 본 연구를 더욱 발전시키기 위해서는 제시된 품질특성의 상관관계 및 중요도를 여러 프로젝트에 적용하여 그 실효성을 판단하고, 지속적으로 보완해야하고, 소프트웨어 수명주기 뿐만 아니라 소프트웨어 유형에 대한 상관관계 및 중요도를 포함하여 수명주기와 유형이 모두 고려된 모형을 개발해야 할 필요가 있다. 이러한 부분들은 향후 연구과제로 지속적으로 연구가 수행되어야 할 것이다.

참고 문헌

[1] G. C. Schulmeyer and J. I. McManus, "Handbook of Software Quality Assurance," 3rd Ed., Prentice Hall, 1999.
 [2] J. W. Horch, "Practical Guide to Software Quality Management," Artech-House Publisher, 1996.
 [3] B. W. Boehm, "Software Engineering Economic," Prentice-Hall, 1981.
 [4] J. A. McCall et. al., "The Automated Measurement of Software Quality," IEEE, 1981.
 [5] M. W. Evans and J. J. Marciniak, "Software Quality Assurance and Management," John Wiley & Sons, 1987.
 [6] ISO/IEC 9126-1, "Information Technology - Software Product Quality - Part 1 : Quality Model," ISO/IEC JTC1/SC7/WG6, 1999.
 [7] ISO/IEC 12207-1, "Information Technology - Software life cycle processes - Part 1 : Life cycle data," ISO/IEC JTC1/SC7, 1997.
 [8] T. L. Saaty, "The Analytic Hierarchy Process," McGraw-Hill, New York, 1980.

[9] ISO/IEC 14598-1, "Information Technology - Software Product Evaluation - Part 1 : General Overview," ISO/IEC JTC1/SC7, 1998.
 [10] US-DoD, "Software Development and Documentation," MIL-STD-498, 1994.
 [11] A. C. Gilles, "Software Quality : Theory and Management," Chapman & Hall, 1992.
 [12] J. K. Cho and S. J. Lee, "An Evaluation Model for Software Quality Improvement," Multi-Conference '03SERP, 2003.
 [13] 정기원, 윤창섭, 김태현, "소프트웨어 프로세스와 품질", 홍릉과학출판사, 1997.
 [14] 조재규, 이승중, "소프트웨어 품질향상을 위한 품질평가 모형에 관한 연구", 한국정보과학회 춘계학술대회, 2003.
 [15] 조재규, "소프트웨어 개발단계별 정량적 품질평가 모형", 국방대학교 석사학위논문, 2003.
 [16] 조재규, 이길섭, 이승중, "소프트웨어 개발단계별 중요도를 고려한 품질특성", 정보과학회 제6회 소프트웨어공학 학술대회논문집, 제6권 제1호, 2004.



조 재 규

e-mail : dr-cjk@hanmail.net

1994년 금오공과대학 기계공학 학사
 2004년 국방대학교 전산학 석사
 1997년~2000년 학생중앙군사학교
 2000년~2001년 육군본부 군수참모부
 2003년~현재 육군 지도창

관심분야 : 소프트웨어 감리, 정보시스템 감사, 소프트웨어 품질 보증 및 시험, 차세대 네트워크



이 길 섭

e-mail : gislee@kndu.ac.kr

1985년 금오공과대학 전자공학 학사
 1987년 한국과학기술원 전산학 석사
 1992년 한국과학기술원 전산학 박사
 1991년~1995년 육군전산소 개발실
 1996년~1998년 국방부 정보체계국

1999년~2003년 육군 교육사

2003년~현재 국방대학교 전산정보학과

관심분야 : 정형기법, 분산시스템 모델링, 소프트웨어공학(프로세스, 개발방법론, 시험)



이 승 중

e-mail : ljck@kndu.ac.kr

1983년 육군사관학교 전산학 학사
 1990년 미주리주립대 전산학 석사
 1993년 미주리주립대 전산학 박사
 1994년~1995년 육군전산소 광역전산담당
 1995년~1996년 국방전산소 표준제도담당

1996년~2001년 육군사관학교 전산학과 교수

2001년~현재 국방대학교 전산정보학과 교수

관심분야 : 소프트웨어공학, 차세대 네트워크