2003 추계학술발표회 논문집
한국원자력학회

S/W V&V 정보와 EDM 다이어그램을 이용한 품질특성인자 (QCV) 도출

Development of Quality Characteristics Value (QCV) using S/W V&V results and EDM diagram

김찬수, 정창현
서울대학교 원자핵공학과
서울시 관악구 신림동 산56-1

요 약

현재 원자력발전소 디지털 계측제어 계통의 이용이 늘어나고 있기 때문에 디지털 설비에 한 부분인 소프트웨어의 안전 관련 문제가 크게 대두되었다. 하지만 소프트웨어의 품질을 정량적으로 평가하는 일은 많은 어려움을 가지고 있어 그 활용이 미미한 실정이다. 본 논문에서는 소프트웨어 확인 및 검증 작업과 EDM 모델을 활용하여 소프트웨어의 생명주기별 품질특성인자를 도출하고자 한다. 따라서 소프트웨어 확인 및 검증 작업에서 평가 기준으로 이용된 네 가지 인자(추적성, 완전성, 이해성, 정확성)를 추출해 EDM 모델에 맞추어 재평가하고, 그 결과를 기준으로 다이아몬드 형태의 그래프를 그리고 그 넓이를 품질특성인자로 제안하였다. 본 논문에서는 제안된 방법론을 원전제어계통 소프트웨어 개발공적에 적용하여 타당성을 타진하여 보았다. 그 결과 품질특성인자로 제안된 변수가 품질계수로서 적절한 특징을 가지고 있으며 소프트웨어 개발공정을 잘 반영하고 있음을 알 수 있었다.

Abstract

In current researches, safety-related problems of software (S/W) occur in digital equipments, because digital I&C systems are widely used and expand their ranges to various applications in Nuclear Power Plants. It, however, does not hold general position to estimate an appropriate S/W quality. Thus, the Quality Characteristic Value (QCV) to quantify the S/W quality through each S/W life-cycle is considered. The QCV is obtained as follows: 1) Scoring QCV-factors such as Correctness, Traceability, Completeness and Understandability, 2) Deriving the lozenge graphs: pointing the scored values by setting each factor as each axis and lining between the points 3) Measuring the area of graph, and that is the QCV, and 4) Applying QCV and graphs to Plant Control System. For all of these procedures, the series of quantification frameworks exhibit characteristics as a quality factor and will be applicable to regulatory guide of S/W approval procedure.

1. 서론

현재 원전제어계통은 기존의 아날로그 계측 제어 시스템과 달리 전산 소프트웨어 기술이 주류를 이룬다. 일반적으로 소프트웨어의 신뢰도는 노후화 현상이 없고 비선형적이며 자원을 공유하는 등의 특징으로 인해, 측정하기가 어렵다. 따라서 직접적인 정량적 평가를 하기 힘들기 때문에, 시험을 통해 얻은 자료를 이용하여 신뢰도를 추정하는 연구가 활발히 진행되고 있다.

본 논문에서는 소프트웨어의 고장 확률을 구할 수 있는 방법을 개발하기 위한 선행연구로서, 소프트웨어의 품질특성인자를 도출하였다. 즉 소프트웨어의 생명주기별로 수행되는 확인 및 검증 작업의 결과를 통해 소프트웨어의 각 생명주기별 개발 작업을 정량적으로 평가할 수 있는 품질특성인자를 도출하여 소프트웨어의 개발 공정 자체를 평가하고자 한다. 그리고 이에 따라 소프트웨어 개발의 전 과정이 안전 관련 수준(safety-related level)으로 분류되어 엄격한 관리가 이루어지는 원전제어계통을 대상으로 하여 소프트웨어의 각 생명주기별 업무를 정량적으로 평가하였다.

2. 본론

2.1. 소프트웨어 확인 및 검증 (S/W V&V) 작업

통상 신뢰성 있는 소프트웨어를 개발하기 위해 해당 소프트웨어의 요구 분석, 설계, 구현, 시험 등이 품질보증 요건에 따라 정확히 이루어지고 있는지를 확인하고 검증하는 작업을 수행한다. Sequential Model에 따라 소프트웨어를 개발하는 경우, 소프트웨어 확인 및 검증 (S/W V&V)은 구상단계(concept phase), 요건단계(requirement phase), 설계단계(design phase), 구현단계(implementation phase), 시험단계(test phase), 설치 및 유지보수 단계(installation and maintenance phase)에 걸쳐 각각 수행된다.

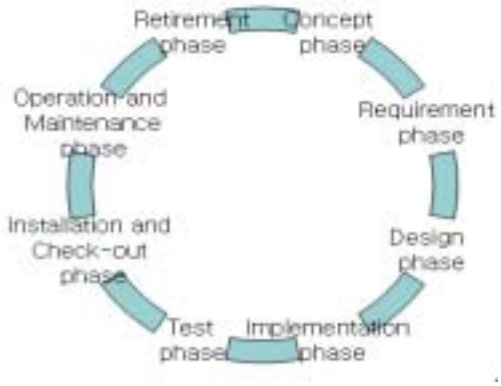

그림 1. 소프트웨어 생명주기.

그림 1과 같이 소프트웨어가 개발되는 경우, 소프트웨어 확인 및 검증 작업은 그림 2와 같은 순서에 따라 진행된다. 각 생명주기별 산출물 자체에 대한 평가와 이전단계 산출물의 내용이 개발프로세스가 진행됨에 따라 얼마나 잘 반영되어 있는지에 대한 평가로 이루어진다. 구체적으로 추적성, 정확성, 이해성, 적용가능성, 일관성, 완전성과 같은 인자를 각 항목별로 평가하고, 이들의 평가결과를 matrix 형태로 표현하게 된다.

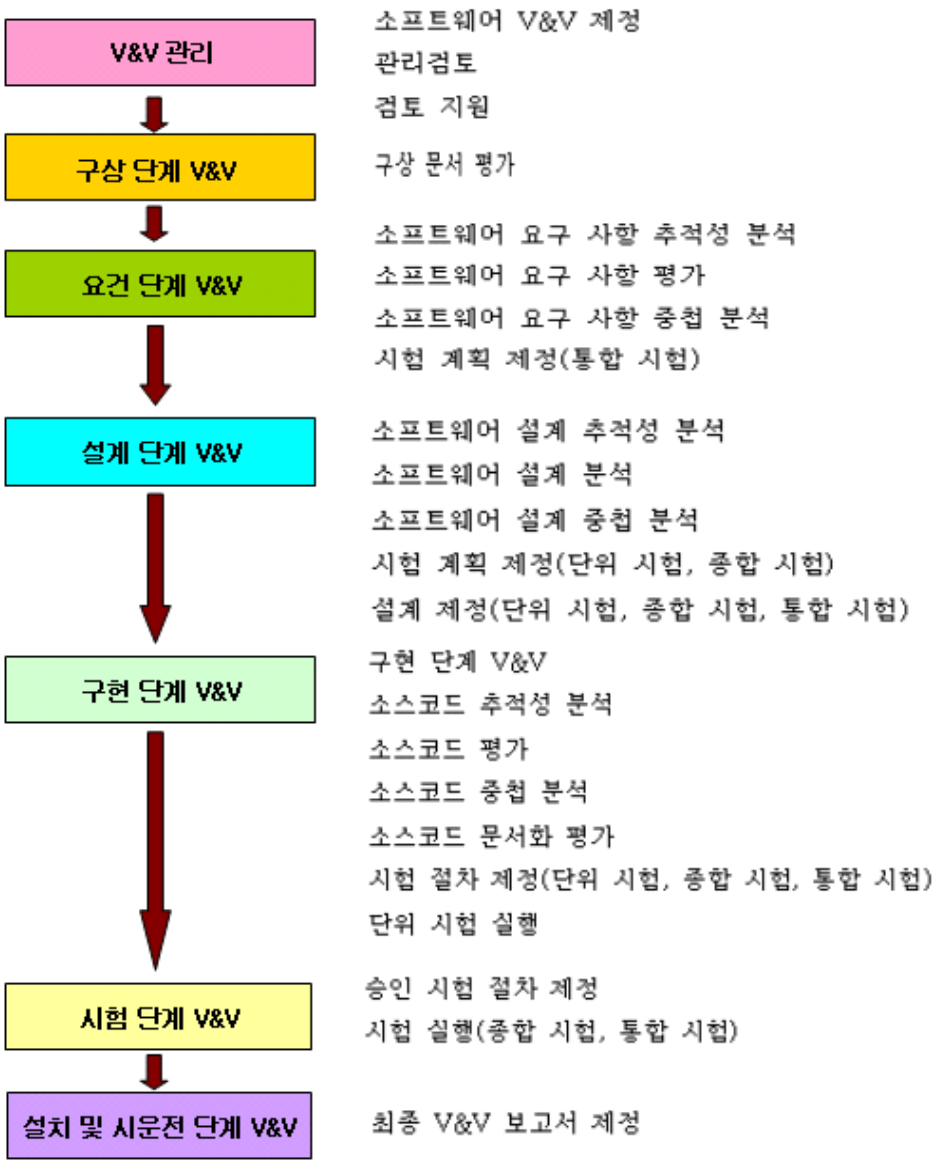

그림 2. 소프트웨어 개발 생명주기별 확인 및 검증 작업.

그림 2의 생명주기별 확인 및 검증 작업 개요는 IEEE 1012를 기준으로 한다. 그림 1과 그림 2에서 알 수 있듯이 소프트웨어의 개발은 순차적으로 진행되고, 확인 및 검증 작업은 개발 프로세스가 합리적으로 진행되었음을 확인하는 과정이 된다.

2.2. 정량적 품질특성인자 도출

본 절에서는 소프트웨어 개발공정의 품질특성을 정량적으로 평가하기 위한 방법을 제안한다. 조직인자를 사용하여 원전 조직을 평가하기 위해 제안되었던 SPOOM-EDM (Self Poly-Oriented Organization Model - Evaluation Diamond Model) 모델을 고찰하고, 이를 응용하여 소프트웨어의 각 생명주기별 개발업무의 품질특성인자를 도출하는 방법을 개발한다.

SPOOM-EDM 방법 중 EDM 방법은 조직인자를 이용하여 조직의 안전도를 평가하는데 SPOOM 방법에서 도출한 조직인자 범주를 그래프의 네 축으로 구성한다. 각 사례 분석을 통해 해당 조직 관련 사건의 인자별 점수를 매기고, 구한 점수를 각 축의 값으로 사각형을 그린다. 이 방법으로 그려진 사각형의 넓이를 구하고, 그 넓이 값을 조직의 안전성을 평가하는 하나의 척도로 삼는다. 즉, 넓이가 넓을수록 조직의 안전성은 높다고 판단할 수 있으며 이 넓이의 변화 및 각 평가요소의 변화를 통해 조직이 갖는 방향성을 파악한다.

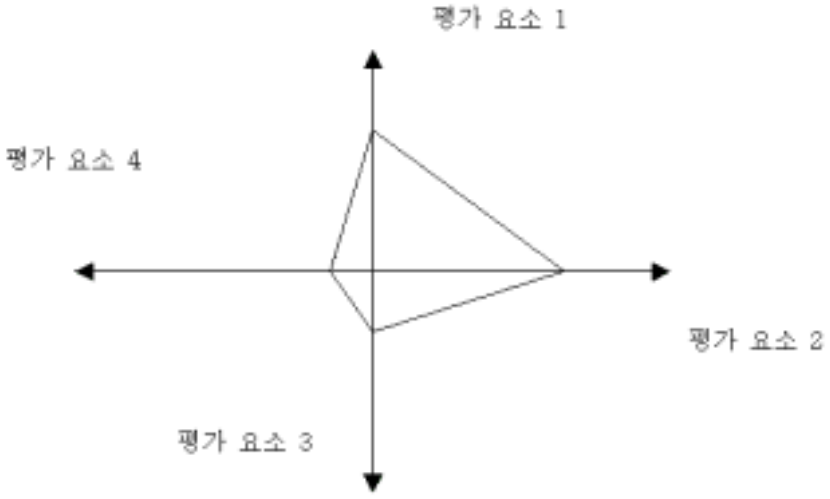

그림 3. EDM 방법의 개략도.

본 논문에서는 이 EDM 방법을 사용하여 품질특성인자를 도출하기 위해 추적성(trackability), 완전성(completeness), 이해성(understanability), 정확성(correctness)을 네 평가요소로 선정하였다. 네 가지 평가요소는 아래와 같다.

- 추적성: 소프트웨어의 각 생명주기별 산출물이 이전 단계의 산출물로부터 확인이 되는 근거 있는 내용인지를 평가하는 요소이다.
- 완전성: 추적성 분석의 역의 과정을 통해 수행되는데, 이전단계에서 발생한 산출물들이 다음단계에 잘 반영 되었는지를 확인하는 요소이다.
- 이해성: 각 문서의 항목이 이해하기 쉽게 기술되었는가를 평가하는 요소이다.
- 정확성: 각 문서의 해당 항목이 얼마나 정확하게 기술되었는지를 평가하는 요

소이다.

이렇게 소프트웨어의 확인 및 검증 작업에 사용되는 네 가지 평가 항목(추적성, 완전성, 정확성, 이해성)을 EDM 모델에 적용시키고자 한다. 즉 4가지 평가 항목들이 그림 2의 개략도에서 각각 한 축을 차지하게 된다. 이렇게 각 축에서 점수를 매긴 점을 이어 EDM 다이어그램을 그린다. 그리고 품질특성인자는 EDM 다이어그램의 넓이로 정의한다. 수정한 EDM 방법을 사용하여 점수를 매긴 요소를 축에 나타내는 규칙은 다음과 같다.

- ○, △, ×로 표시된 각 평가요소의 결과는 ○는 $\frac{1}{n}$, △는 $\frac{1}{2n}$, ×는 0점으로 계산한다. (여기서 n은 평가 대상 항목의 총 수이다)
- 각 축의 최대값은 1이다.
- 모든 소프트웨어 개발 업무가 이상적으로 진행될 경우, 각 꼭지점의 좌표는 $(1,0)$, $(0,1)$, $(-1,0)$, $(0,-1)$이 되고 그 때의 넓이는 2가 된다.
- 품질특성인자는 EDM 다이어그램의 넓이로 정의한다.

2.3. 사례 분석

본 절에서는 소프트웨어 확인 및 검증 작업이 구상단계, 요건단계, 설계단계, 구현단계, 시험단계 등 소프트웨어 생명주기별로 수행된 원전제어계통의 Field Control System (FCS), Engineer Interface System (EIS), Operator Interface System (OIS) 개발공정에 실제 EDM 방법을 적용해보았다. 이에 따라 원전제어계통의 각 생명주기별 개발업무가 어떻게 이루어졌는지에 대한 평가를 하고자 한다. 실제로 요건단계 확인 및 검증작업 rev.0 및 rev.1과 설계단계 확인 및 검증작업을 대상으로 EDM 방법을 적용하였다.

표 1. 생명주기 및 소프트웨어 모듈별 평가치

생명주기 단계	모듈 종류	추적성	정확성	이해성	완전성
요건 rev.0	FCS	1.0000	0.9508	0.9918	0.9344
	EIS	1.0000	0.9919	0.9973	0.9892
	OIS	1.0000	0.9921	0.9934	0.9790
요건 rev.1	FCS	0.9869	0.9901	1.0000	0.9902
	EIS	1.0000	0.9943	1.0000	0.9501
	OIS	0.9926	0.9985	1.0000	0.9496
설계	FCS	1.0000	0.8169	0.8310	0.3030
	EIS	0.9167	0.9792	0.9732	0.6920
	OIS	0.8278	0.9597	0.9528	0.4631

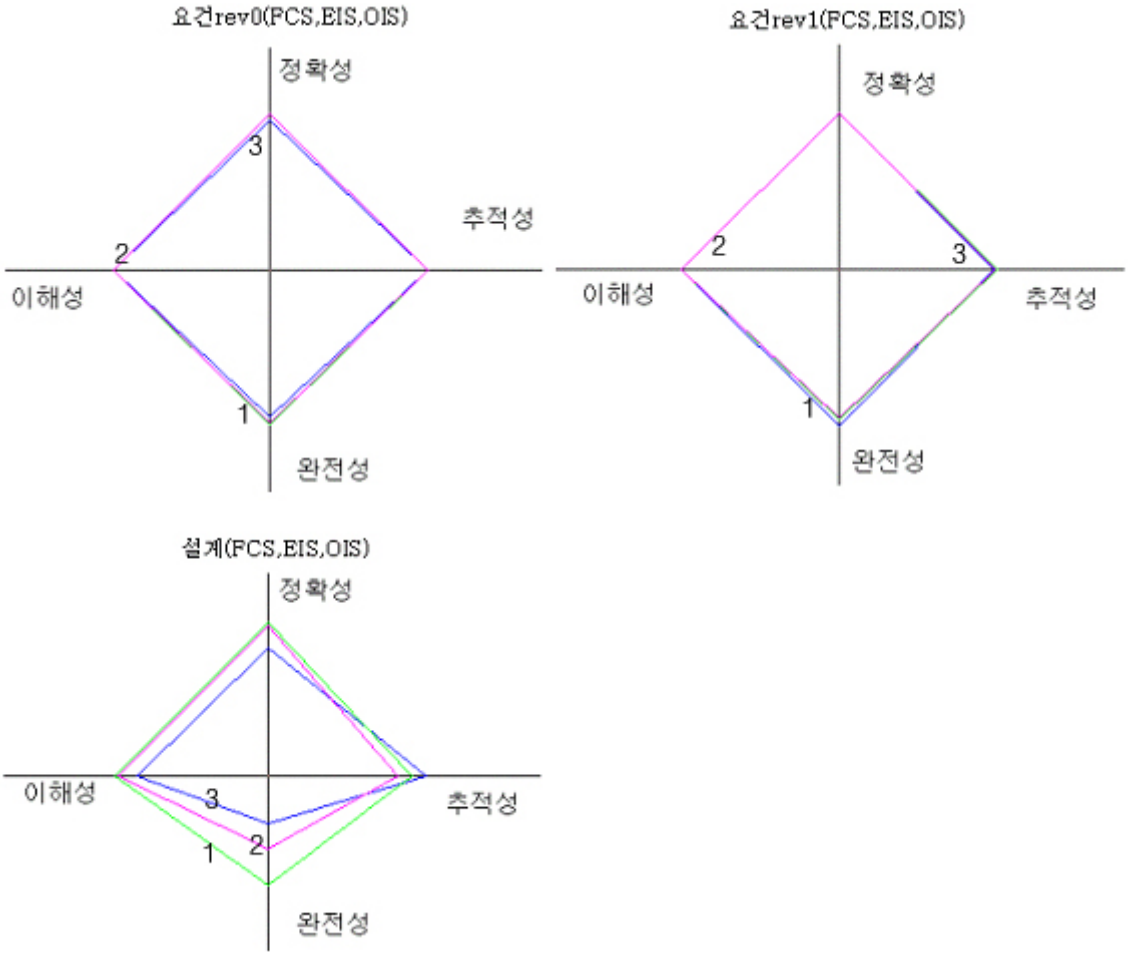

그림 4. 각 생명주기 단계별 EDM 다이어그램(1-> EIS, 2-> OIS, 3-> FCS).

그림 5처럼 각각의 단계와 모듈에 따라 차이가 많이 발견되는 경우에 한해 EDM 다이어그램을 그려보았다. 이 때에, 바깥쪽의 다이어그램은 요건 rev.0 단계이고 안쪽 다이어그램은 설계단계이다.

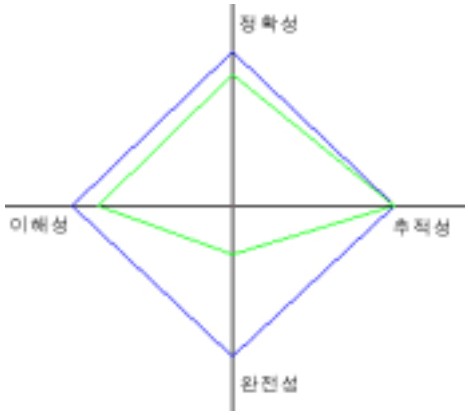

그림 5. 요건 rev.0와 설계의 FCS.

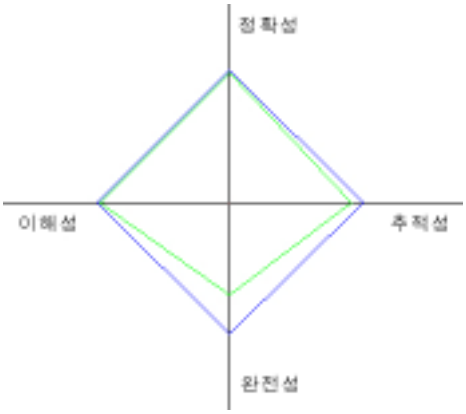

그림 6. 요건 rev.0와 설계의 EIS.

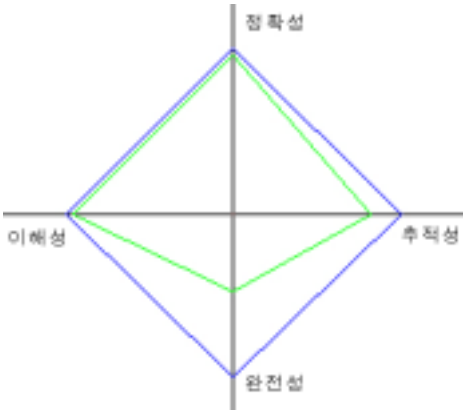

그림 7. 요건 rev.0와 설계의 OIS.

각 다이어그램의 경향을 살펴보면 주로 요건 rev.0와 설계 단계에서 큰 차이를 보이고 있음을 알 수 있다. 그림 4에서 볼 수 있듯이 요건단계에서는 FCS, EIS, OIS 소프트웨어 모듈이 큰 문제없이 개발되었다고 평가 할 수 있다. 그러나 그림 5, 그림 6, 그림 7을 살펴보면 설계 단계의 경우 각 소프트웨어 모듈이 눈에 띄는 차이를 보이며 개발이 진행되었음을 알 수 있다(이는 개발공정이 효과적으로 관리되지 못했음을 의미한다). 이는 요건 rev.0 단계와 설계 단계의 비교를 통해서도 파악할 수 있다. 뿐만 아니라 설계단계 업무에서 OIS 및 FCS의 완전성이 현저하게 떨어져서 전체 설계단계 개발 업무의 질이 상당히 저하되었을 관찰할 수 있다.

다이어그램을 통해 해당 개발업무에 대한 평가는 넓이를 통해 이루어지고 (품질특성인자), 각 축의 길이는 소프트웨어 각 모듈의 네 가지 평가 요소가 된다. FCS 모듈의 경우, 완전성이 현저하게 낮게 평가되어 개발업무의 일관성이 부족하다고 판단된다.

그림 5, 그림 6, 그림 7을 통해 소프트웨어 모듈이 요건단계와 설계단계의 두 생

명주기를 거치면서, 개발 업무의 품질이 어떻게 변화했는지 알 수 있다. 요건단계 확인 및 검증 결과, rev.0과 rev.1은 상당히 만족할 만한 수준에서 요건단계 개발 업무가 진행되었음을 시사한다. 실제 소프트웨어 개발 과정 중 요건변경이 있어서 해당요건에 관한 확인 및 검증 작업을 다시 수행한 결과가 요건단계 확인 및 검증 작업 rev.1이므로 이와 같은 결과는 당연하다. 하지만 요건단계와 설계단계 업무 간에는 여러 평가인자가 상당한 차이를 보였다. 이를 통해 설계단계에서 문제점이 있었다고 추정할 수 있다. 구체적으로 평가인자 추이가 어떻게 변했고, 그 원인이 무엇인지 추정한 것은 다음 표에 기술하였다. (아래 표의 추정 원인은 산점도를 그려서 두 변수사이의 개략적인 상관관계를 살펴보는 것과 동일한 방법으로 파악할 수도 있다.)

표 2. 평가 요인의 변화 원인 추정

모듈	평가 요인의 변화 추이	추정 원인
FCS	설계 단계가 진행되면서 개발 업무의 완전성이 크게 낮아지고 정확성 및 이해성 역시 소폭 감소	요건단계에서는 개념문서를 통해 초안이 잡힌, 요구 조건을 보다 구체화하면서 충실한 업무가 진행되었다. 반면에 설계단계에서는 요건문서에 대한 철저한 분석이 없이, 「단기간 소프트웨어 개발」에 초점을 맞추어 업무를 진행하여 요건과는 거리가 있는 설계 업무가 수행된 것으로 판단된다.
EIS	설계 단계로 진행되면서 완전성이 현저히 감소	
OIS	설계 단계로 진행되면서 완전성 및 추적성이 감소	

표 3. 각 생명주기 단계 및 모듈별 품질특성인자 값

생명주기 단계	모듈 종류	품질특성인자(QCV)
요건 rev.0	FCS	1.8775
	EIS	1.9784
	OIS	1.9646
요건 rev.1	FCS	1.9673
	EIS	1.9444
	OIS	1.9409
설계	FCS	1.0253
	EIS	1.5792
	OIS	1.2667

표 3을 통해 그림 5, 그림 6, 그림 7에서 비교한 단계들의 품질특성인자 값을 다시 살펴보면, 품질특성인자 값에 차이가 있음을 알 수 있다.

표 4. 요건 rev.0 단계와 설계 단계의 모듈별 품질특성인자 값 비교

생명주기 단계 및 모듈 종류	품질특성인자(QCV)
요건 rev.0 FCS	1.8775
설계 FCS	1.0253
요건 rev.0 EIS	1.9784
설계 EIS	1.5792
요건 rev.0 OIS	1.9646
설계 OIS	1.2667

EDM 다이어그램에서 쉽게 구별되듯이 소프트웨어 확인 및 검증의 네 가지 요소로 구성된 품질특성인자 값 역시 많은 차이를 보이고 있다. 표 4는 요건단계와 설계단계의 품질특성인자 값을 나타낸다. 해당 표에서 알 수 있듯이 요건단계는 모든 품질특성인자가 비교적 이상적인 값인 2에 가까운 좋은 값을 나타내고 있다. 따라서 완전성, 이해성, 정확성, 추적성이 바르게 보장되는 개발업무가 진행되었다고 판단할 수 있다. 하지만 품질특성인자나 EDM 다이어그램을 살펴보면, 설계단계의 경우에는 개발 업무가 바르게 진행되지 않았다는 것을 알 수 있다.

요건단계와 설계단계의 품질특성인자 값 차이는, 요건이 설계에 충분히 반영되지 않았음을 의미한다. 따라서 설계단계에서 요건단계의 요구 사항을 충분히 분석하여 설계에 반영하는 작업이 필요하고, 이런 점을 재검토해야 한다. 실제로 소프트웨어 개발자들이 요건을 충분히 분석하는 절차 없이 세부 모듈을 구현한 후 설계서를 작성하는 일이 많아 여러 번의 시행착오가 있었다. 이는 세부 모듈 사이의 연계문제를 야기했고, 결국 소프트웨어 개발비용을 증가시키는 결과를 낳았다고 판단된다.

3. 결론

본 논문에서는 소프트웨어의 고장 확률 등을 구할 수 있는 방법을 찾아보기 위한 연구로 품질특성인자를 도출하였다. 즉, 소프트웨어 개발 생명주기 별로 수행되는 확인 및 검증 작업의 결과를 이용해 소프트웨어의 각 생명주기별 품질특성인자를 도출하여 소프트웨어의 개발 공정을 정량적으로 평가하였다.

원자력발전소의 조직을 평가하기 위해 제안된 SPOOM-EDM 방법 중, 각 평가인

자에 점수를 주어 다이아몬드 모양의 사각형을 그리는 EDM 다이어그램을 이용하였다. 여기에 소프트웨어 확인 및 검증작업의 결과 얻을 수 있는 평가 요소인 정확성, 추적성, 완전성, 이해성을 평가인자로 삼았다.

요건단계 및 설계단계에 수행된 소프트웨어 확인 및 검증작업의 결과를 이용해 각 개발단계에서 취약하게 진행된 부분을 확인하였다. 그리고 실제 개발된 원전 소프트웨어의 문제점을 비교해 보았다. 이는 소프트웨어 인허가 관련 업무의 보조 수단으로 사용할 수 있을 것으로 생각된다.

추후 일반적인 제품의 제조공정에 사용되는 공정능력지수와 같은 객관적인 평가기준으로 자리매김하기 위해서는 보다 많은 문헌 연구와 사례 적용을 통해 이론적으로 수정된 EDM 방법을 보강해야 한다. 그리고 네 가지 평가인자 각 축의 증가감소분이 통계적으로 유의하게 변했는지 검정해 보는 과정을 포함해야 할 것이다. 이를 통해 객관성을 확보할 수 있다. 앞으로는 교육·인력변화·훈련·세미나 등의 변화시점을 확인하고 그에 따라 개발업무 개선이 있었는지를 유의성 검정을 통해 요건단계 뿐 아니라 모든 생명주기에서 이와 같은 변화가 소프트웨어의 개발에 미친 영향에 대해 살펴보는 작업이 필요하다.

4. 참고문헌

[1] IEEE 1012-1987, "SOFTWARE VERIFICATION & VALIDATION PLAN"

[2] http://isa.kaeri.re.kr/, 한국원자력연구소 종합안전평가팀 홈페이지

[3] "PCS 소프트웨어 V&V 계획서(WRD-RDV-PCSS-001)", ㈜ 우리기술

[4] "PCS 요건단계 V&V 보고서(WRD-RDV-PCSS-003)", ㈜ 우리기술

[5] "PCS 설계단계 V&V 보고서(WRD-RDV-PCSS-005)", ㈜ 우리기술

[6] SayHyung Kim, "The Development of SPOOM-EDM Method for Evaluating Nuclear Power Plant Organization using Organizational Factors", Seoul National Univ., 2003

[7] Sommerville, “Software Engineering,” Part Two Requirements and Specification, Addison-Wesley, 1995