ISSN 1738-480X (Print)
ISSN 2288-3592 (Online)
Journal of the Korea Society of Systems Engineering
vol.9, No.1: 33p~45p, June 2013

시스템엔지니어링 기법을 적용한 가압중수로 노심관리 지원시스템 개발 사례

염충섭[1)*] 김진일[2)] 송용만[3)]

1) 고등기술연구원, 2) 에스엔에스 이엔지, 3) 한국원자력연구원

A Case Study on the Application of Systems Engineering to the Development of PHWR Core Management Support System

Choong Sub Yeom,[1)] Jin Il Kim[2)], Young Man Song[3)]

1) Institute for Advanced Engineering, 2) SNS Eng, 3) Korea Atomic Energy Research Institute

Abstract : Systems Engineering Approach was applied to the development of operator-support core management system based on the on-site operation experience and document of core management procedures, which is for enhancing operability and safety in PHWR (Pressurized Heavy Water Reactor) operation. The dissertation and definition of the system were given on th basis of investigating and analyzing the core management procedures. Fuel management, detector calibration, safety management, core power distribution monitoring, and integrated data management were defined as main user's requirements. From the requirements, 11 upper functional requirements were extracted by considering the on-site operation experience and investigating documents of core management procedures. Detailed requirements of the system which were produced by analyzing the upper functional requirements were identified by interviewing members who have responsibility of the core management procedures, which were written in SRS (Software Requirement Specification) document by using IEEE 830 template. The system was designed on the basis of the SRS and analysis in terms of nuclear engineering, and then tested by simulation using on-site data as a example. A model of core power monitoring related to the core management was suggested and a standard process for the core management was also suggested. And extraction, analysis, and documentation of the requirements were suggested as a case in terms of systems engineering.

Key Words : Systems Engineering Process, Requirement Definition, Requirement Analysis, PHWR, CANDU (CANadian Deutrium and Uranium), Core Management, NPP (Nuclear Power Plant)

* corresponding author : Choong Sub Yeom, Institute for Advanced Engineering, csyeum@iae.re.kr

* This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(http://creativecommons.org/licenses/by-nc/3.0) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. 서 론

가압경수로형(PWR, Pressurized Water Reactor) 원자력 발전소는 예방정비계획에 의해 원자력 발전소를 정지시키고 핵연료를 교체하는데 비해 가압중수로형(PHWR, Pressurized Heavy Water Reactor) 원자력 발전소는 원자로를 정지시키지 않고 운전 중에 매일 핵연료를 교체한다. 또한 가압중수로의 노심은 380개의 압력관내에 핵연료를 장전하는 구조로 되어 있고, 노심제어를 위한 출력제어장치들의 구조가 복잡하게 이루어져 있다. 이러한 노심의 구조 및 운전 특성상 PHWR의 노심관리는 PWR에 비해 매우 복잡하고, 또한 주기별 노심관리 계획에 따라 매우 복잡한 계산을 수행해야 하는 경우가 많으며, 담당자의 노하우에 따라 업무의 효율성이 달라지기도 한다. 이러한 현안을 해결하기 위한 PHWR 노심관리 자동화를 통한 지원 시스템 개발이 이루어지게 되었다.

가동 중인 원자력발전소에 적용하기 위해서는 다수의 관련 운전 엔지니어와 설계 엔지니어의 참여가 필수적이었으며, 개발 엔지니어를 포함한 다양한 이해당사자의 참여와 더불어 개발 시스템의 다양성과 복잡성을 고려한 개발 프로세스에 대한 정의의 필요성이 인식되었다. 이러한 개발 프로세스 정의 및 관리 방안으로 시스템엔지니어링(SE, Systems Engineering) 기법을 적용하였다[1].

2. SE를 적용한 개발 프로세스 정의

2.1 표준 SE 프로세스

시스템엔지니어링 표준 프로세스는 1969년에 제시된 MIL-STD 499에서 시작되어 1990년대에 EIA/IS 632와 IEEE 1220SO/IEC 등으로 발전하여, EIA/632, IEEE 1220, ISO15288로 변천하고 있다[1,2]. 기술의 발달과 시스템의 복잡성을 고려한 시스템엔지니어링 표준 프로세스의 적용은 각 기술분야에 특화되어 사용되고 있다.

그림 1과 2, 3은 각각 EIA/632, IEEE 1220, ISO 15288의 표준 프로세스이다.

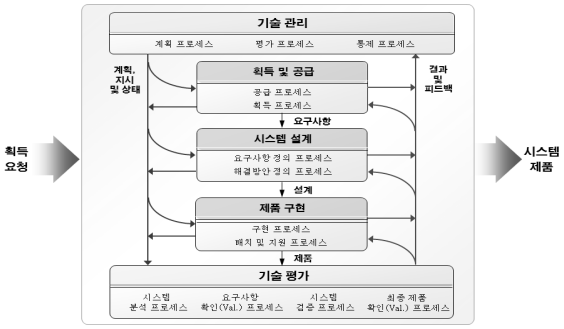

그림 1. EIA/632 SE 표준 프로세스

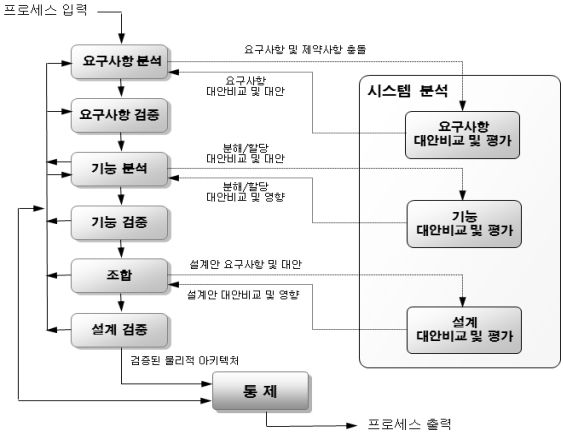

그림 2. IEEE 1220 SE 표준 프로세스

그림 3. ISO/IEC 15288 SE 표준 프로세스

2.2 적용된 SE 프로세스

PHWR 노심관리 지원시스템을 개발하기 위한 표준 프로세스는 IEEE1220과 ISO/IEC 15288을 고려하여 그림 4와 같이 정의하였다.

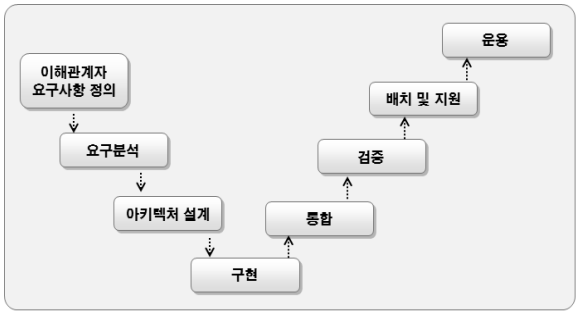

그림 4. PHWR 노심관리 지원시스템 개발 프로세스

3. SE 프로세스를 적용한 시스템 개발

3.1 시스템 정의

PHWR을 운영하고 있는 월성 원자력발전소의 노심관리 업무에 대한 현행 업무 분석을 통해 사용자의 의견을 반영하여 개발하고자 하는 시스템 정의가 이루어졌다.

PHWR의 운전 중 수행하는 노심관리는 그림 5와 같이 “노심관리절차서”에 규정된 핵연료 교체와 신연료 저장, 연료 건전성 확인, ROP(Regional Overpower Protection) 검출기 교정, 냉각재 유량 확인, 열출력 점검, 표준영역중성자속 변경, VFD (Vanadium Flux Detector) 교정, 제어용 백금피복 검출기 교정의 제반업무로 정의된다[3].

노심관리 지원 시스템은 DCCS (Digital Control Computer System)에서 게이트웨이 서버를 거친 현장 데이터 파일을 이용하여 노심관리 프로세스별 알고리즘 및 관련 코드와의 연계 처리등을 통하여 사용자가 노심관리를 효율적으로 수행하도록 노심관리 업무를 지원하는 시스템으로 정의 할 수 있다[4].

그림 5. 주기별 PHWR 노심관리 업무 정의

3.2 시스템 요구사항 정의 및 명세서 작성

3.2.1 요구사항 정의 및 분석

시스템의 성공척도는 실제로 개발 시스템을 현장에서 사용하게 될 발전소 직원들의 다양한 요구사항을 얼마나 잘 만족시키느냐에 달려있다고 볼 수 있다. 이러한 요구사항은 시스템 개발 초기에 식별되어 설계 및 구현으로 연결되어야 한다. 그러나 이러한 요구사항이 시스템 설계나 코딩 도중에 발견될 경우, 또는 잘못된 요구사항을 가지고 시스템을 개발할 경우 비용 및 일정 측면에서 많은 낭비가 발생하게 된다. Bill 등은 소프트웨어 시스템 개발 프로젝트에 있어서 에러를 수정하는데 소요되는 비용이 요구사항 분석단계에 비해 운영 단계에서는 1000배까지 늘어날 수 있음을 그림 6과 같이 발표하였다[5].

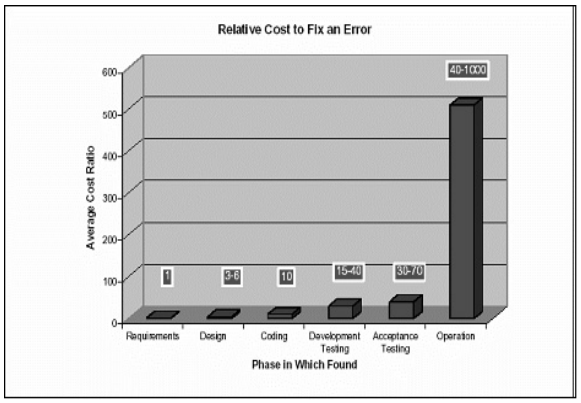

그림 6. 시스템 개발 단계별 에러 수정에 소요되는 상대적 비용

이렇듯 시스템 개발에 있어서 요구사항이 매우 중요하지만 실질적으로 요구사항 분석업무를 수행함에 있어 사용자나 시스템 개발자 모두 어려움을 겪게 된다. 사용자는 요구사항을 잘 표현하는 방법에 익숙지 못하며, 심지어는 자기의 요구사항을 정확히 알지 못하는 경우도 많이 있다. 또한 사업이 진행되는 동안 상황이 변함에 따라 요구사항이 바뀌기도 하고, 시스템 개발이 진행됨에 따라 새로운 가능성을 보고 요구사항이 바뀌는 경우도 있다. 시스템 개발자의 입장에서는 모든 상황에 맞도록 개발된 요구사항 분석방법이 정의되어 있지 않고 일

반적인 방법을 상황에 맞도록 조정해서 사용해야 하며, 요구사항 추출에 있어 사회 심리학적인 문제에 귀착할 수도 있고, 일반적으로 요구사항 분석보다는 설계에 치중하는 경향이 있으며 이해당사자와의 의사소통에 장벽이 존재하는 것이 현실이다. 이러한 상황을 극복하기 위해 요구사항 분석을 위한 충분한 계획과 방법이 필요하게 된다.

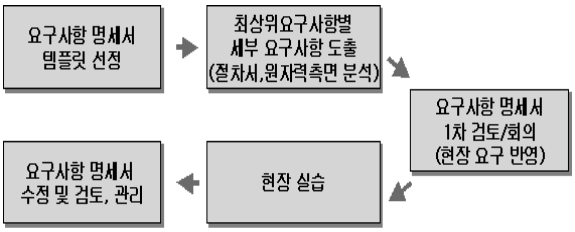

그림 7. 세부 요구사항 분석 프로세스

본 연구에서는 그림 7과 같이 요구사항 분석이 이루어졌으며, 두 가지의 방법으로 하위 세부 요구사항의 추출이 이루어졌다. 첫 번째는 상위 요구사항의 선정과정에서 도출된 방법론이나 알고리즘, 가정 등에서 생산된 기능상의 세부 요구사항을 추출하는 과정이 이루어졌다. 두 번째는 원자력 발전소의 노심관리 업무가 모두 절차서에 의해 수행되고 있기 때문에 중수로 노심관리 업무 자동화 시스템의 요구사항은 월성 발전소의 절차서가 큰 뼈대를 이루게 된다. 세부적으로는 각 절차서에서 수행하도록 규정된 업무를 효율적으로 수행하기 위한 각 담당자들의 요구사항이 있게 된다. 절차서의 내용은 문서화된 내용으로서 파악이 용이하며 명확하지만, 각 담당자들의 세부 요구사항은 매우 다양하고, 담당자들이 일목요연하고도 정확하게 전달하리라는 보장이 없다. 이에 본 연구에서는 이러한 요구사항들을 빠짐없이, 정확하게 기술하기 위해서 요구사항의 추출과 추출된 요구사항을 체계적으로 기술하기 위해 절차서를 기본으로 한 도제법 (apprenticing)을 사용하였다[6,7].

이 과정에서 핵연료관리, 검출기관리, 원자로안전관리 업무 지원, 노심 출력 감시 및 데이터 통합관리를 시스템의 5대 사용자 요건으로 정의하여 규정하였다. 정의한 노심관리 지원 시스템과 관련된 발전소 계통, 관련 문서 등을 고려하여 개발하고자 하는 시스템의 범위 및 규모를 정의하고, 위의 사용자 요구사항으로부터 11가지의 시스템 설계의 상위 기능요구사항(Functional requirements)을 도출하였다.

식별한 상위 요구사항에 대한 정량화 및 세분화된 시스템 기능 요구사항에 따라 원자력 측면에서의 분석내용을 근거로 세분화된 시스템 요구사항을 정의하여 시스템 요구사항 명세서(SRS, System Requirement Specification)를 작성하였다.

작성된 SRS에서의 상위 기능요구사항은 아래와 같다.

시스템은 중수로 노심출력감시 기능을 지원해야 한다.

- 시스템은 운전 이력 및 노심관리 수행 결과 데이터가 통합관리 되어야 한다.
- 시스템은 핵연료교체와 관련된 업무를 지원해야 한다.
- 시스템은 신연료관리와 관련된 업무를 지원해야 한다.
- 시스템은 연료건전성 확인 업무를 지원해야 한다.
- 시스템은 ROP (Regional Overpower Protection) 검출기 교정 업무를 지원해야 한다.
- 시스템은 냉각재 유량 확인 업무를 지원해야 한다.
- 시스템은 열출력 점검 업무를 지원해야 한다.
- 시스템은 표준영역 중성자속 변경 업무를 지원해야 한다.
- 시스템은 VFD(Vanadium Flux Detector) 교정 업무를 지원해야 한다.
- 시스템은 제어용 백금피복 검출기 교정 업무를 지원해야 한다.

3.2.2 요구사항 명세서 템플릿 선정

요구사항 분석 초기에 가장 중요한 것은 추출된 모든 요구사항을 가능한 빠짐없이 문서화하는 것이다. 이를 위해서는 요구사항을 범주화 및 구조화하는 것이 중요한데 요구사항 템플릿이 이러한 기능

을 제공해 준다. 또한 요구사항 템플릿을 이용하여 요구사항을 문서화 해 놓으면 개발자와 사용자간의 의사소통을 위한 도구로 사용할 수 있을 뿐 아니라 모든 이해당사자들이 문서화된 요구사항을 보고 숙고할 수 있는 기회를 제공하게 된다. 또한 설계를 위한 확고한 기반을 제공하며 시스템의 시험 및 검증 계획의 수립에 도움을 줄 수 있다.

소프트웨어 요구사항 명세서 템플릿은 IEEE, European Space Agency, US DOD등에서 제시하고 있는데, 본 과제에서는 이 중 IEEE 830을 채택하였다. IEEE 830을 선택한 이유는 개발하는 시스템이 비록 안전과 직결되는 소프트웨어 시스템으로 보기는 어렵지만, 미국 원자력규제위원회에서 원자력 발전소의 안전시스템에 사용되는 소프트웨어의 요구사항을 명세하는 템플릿으로서 IEEE 830을 사용할 것을 Regulatory Guide 1.171에서 규정하고 있기 때문이다[8,9]. IEEE830에서 제시하는 소프트웨어 요구사항 명세서 템플릿의 개요는 그림 8과 같다.

IEEE 830의 제3장 중 기능 요구사항은 소프트웨어의 모드, 사용자, 객체, 서비스, 입력, 출력 기능 계층구조 등에 따라 구조화 할 것을 권장하고 있다. 시스템은 노심관리 업무와 이에 다른 절차서를 기준으로 계층화하여 요구사항 분석의 효율성을 높이고자 하였다. 또한 IEEE 830 템플릿은 대상 소프트웨어나 프로젝트 환경에 맞게 조정하여 사용함이 바람직하다. 이에 본 과제에서도 IEEE 830템플릿을 조정하여 사용하였다.

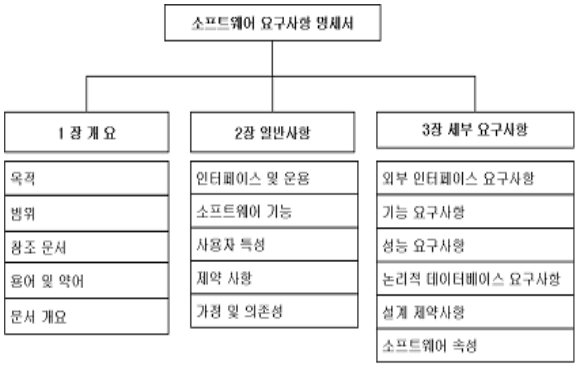

그림 8. IEEE 830 SRS 주요 내용

3.2.3 요구사항 명세서 초안 작성

표 1과 같이 요구사항 명세서 초안의 내용은 월성 발전소 절차서를 중심으로 기술되었다. 그러나 절차서의 내용에는 반드시 필요한 내용만 기술되어 있기 때문에 소프트웨어 개발팀이 이해하기에는 어려운 점이 많이 있었다. 또한 절차서에 기술된 업무를 수행하기 위해서 월성 현장의 엔지니어들은 별도의 엑셀파일을 이용하는 경우가 많이 있었다. 따라서 요구사항 명세서 초안 작성을 위하여 월성 발전소의 절차서 분석 업무와 현장 요구사항 반영을 위한 각 담당자별 요구사항 추출 업무를 수행하였다.

월성 발전소의 노심관리업무는 모두 절차서에 기술된 바에 따라 운영된다. 따라서 모든 용어와 수행해야 할 업무, 기본적인 노심관리 프로세스에 대한 내용은 모두 절차서에 기록되어 있었다.

각 절차서에 기술된 내용을 실제로 수행하는 상세 절차는 절차서에 기술되지 않은 내용도 포함하고 있어 소프트웨어 개발자들의 입장에서는 이 내용이 현장의 실질적인 요구사항인 것이다. 이러한 세부 프로세스들이 현장 담당자들과 개발자 사이에서 논의되었다.

표 1. 최종 요구사항 명세서 구조

1장 서론	1.1 목적 1.2 범위 1.3 참조 문서 1.4 문서개요 1.5 미결사항 리스트
2장 시스템 일반사항	2.1 시스템 인터페이스 및 운용 2.1.1 시스템 인터페이스 2.1.2 사용자 인터페이스 2.1.3 하드웨어 인터페이스 2.1.4 소프트웨어 인터페이스 2.1.5 커뮤니케이션 2.1.6 운용 2.2 시스템의 기능 2.3 시스템 구성 2.3.1 시스템 구성안 2.3.2 소프트웨어 구성 모델 2.4 제약사항 2.4.1 입력 데이터 2.4.2 웹 기반 애플리케이션 2.4.3 모델링 언어 2.4.4 개발 언어 2.4.5 데이터베이스 2.4.6 클라이언트 2.4.7 개발 환경 2.5 가정 및 기타 의존 사항 2.6 기타 사항 2.6.1 COMPAS 시스템 데이터 흐름도 2.6.2 COMPAS 시스템 기능도 2.6.3 COMPAS 시스템 사용자 인터페이스 구성도 2.6.4 사용자 인터페이스 안
3장 세부 요구사항	3.1 핵연료 관리 3.1.1 RFSP 수행 3.1.2 핵연료 교체 3.1.3 신연료 인수, 하역, 저장 및 재고관리 3.1.4 건전성 확인 3.2 원자로 안전관리 3.2.1 원자로 국부 과출력 보호 검출기 교정 3.2.2 냉각재 유량 확인 3.2.3 열출력 점검 3.3 핵 검출기 관리 3.3.1 제어용 백금 피복 검출기 교정 지원 기능 3.3.2 표준 영역 중성자속 3.3.3 바나듐 검출기 교정 지원 기능 3.4 업무기술 지원 3.4.1 반응도 계산 3.4.2 연료교체 진행 확인 3.4.3 Gateway 운전자료 수집 3.4.4 RFSP Standard Input 관리 3.4.5 출력양식의 Scan Image Date 관리
4장 시스템 성능 및 보안 요구사항	4.1 성능 요구사항 4.2 신뢰성 4.3 보안성
5장 용어 및 약어	
6장 부록	

표 2. 요구사항 분석에 따른 세부 기능 요구사항 추출 템플릿

목적	절차서를 위주로 한 해당 업무의 목적을 기술			
입력 데이터	출처	데이터명	형태	위치
	Gateway			
	PDAS			
	RFSP			
	NUCIRC			
	ROVER-F			
	사용자			
프로세싱	입력을 출력으로 변환하기 위한 프로세스 기술			
출력데이터	배포처	출력물명		
	문서			
프로세싱	해당 프로세스의 수행 절차			

본 연구에서 개발하고자 하는 중수로 노심관리 시스템의 특성은 각 절차서 별로 수행해야 할 기능 중심으로 구성되어야 한다는 것이다. 이에 각 절차서 별로 요구되는 기능을 체계적으로 추출하기 위해서 기능 추출 템플릿을 작성하였다. 표 2는 본 연구에서 사용한 기능 추출 템플릿을 보여주고 있다. 입력데이터 부분에서 게이트웨이, RFSP (Reactor Fueling Simulation Program) 등은 중수로 노심관리 업무 자동화 시스템과 인터페이스를 갖는 운영 프로그램 및 시스템들이다. 표 1과 같은 템플릿을 이용함으로서 각 기능에 대해 데이터의 입, 출력 관계, 입력 데이터를 출력으로 변환하기 위해 필요한 프로세스 및 에러처리 등의 내용을 빠짐없이 추출할 수 있었다.

기능 요구사항 외에도 성능 요구사항, 보안 및 신뢰성 요구사항, 기타 인터페이스 요구사항들도 식별되었고 명세서에 기술되었다. 또한 IEEE 830 템플릿의 내용을 기반으로 요구사항 명세서 목차를 조정하였다. 조정의 기본 원칙은 내용이 미미한 부분은 통합하여 가능한 내용이 복잡해지지 않도록 하는 것이었다.

이처럼 템플릿을 조정하여 사용하는 이유는 IEEE 830이 모든 소프트웨어의 요구사항 명세에 사용할 수 있도록 작성되었기 때문이다.

요구사항 명세서 초안을 작성함에 있어 일부 요구사항은 좀 더 상세한 분석이나 사용자들의 의사결정이 필요한 부분이 있었다. 이러한 내용들은 요구사항 명세서 작성 중에 정리되었으며 사용자들과의 검토를 위한 입력사항으로 사용되었다.

3.2.4 현장실습 및 요구사항 명세서 검토

요구사항 명세서 초안은 월성 발전소의 절차서와 현장 담당자들의 설명에 기초해서 소프트웨어 개발팀에서 작성하였다. 이 과정에서 불명확한 요구사항이나 사용자들이 결정해야 하는 내용이 식별되었으며 이를 기초로 요구사항 명세서 1차 검토를 수행하였다. 이 검토과정에서 불명확한 요구사항에 대한 논의가 이루어졌으며, 결정해야 할 내용에 대해서 토의하고 결정하였다. 또한 이러한 과정에서 알려지지 않았던 새로운 요구사항이 발생하기도 하였다. 요구사항 명세서 1차 검토를 통해서 많은 쟁점사항들이 해결되었고, 소프트웨어 개발팀은 사용자들의 진정한 요구사항에 한 걸음 더 접근할 수 있었다.

2차 검토를 위해 핵연료 담당, 검출기 담당, 원자로 안전관리 분야로 구분하여 월성 원자력 발전소에서 실제 노심관리 업무를 수행하는 엔지니어들의 업무 공동 수행을 통해 요구사항에 대해 잘못 생각하고 있던 부분과, 생각지 못했던 부분이 발견되었으며 이러한 내용을 바탕으로 요구사항을 정련할 수 있었다.

3.2.4 요구사항 최종 검토 및 관리

요구사항 최종 검토는 요구사항 분석의 마지막 단계이며, 소프트웨어 설계의 시작단계로 진입하게 되는 마일스톤이다. 따라서 요구사항 최종 검토회의에서는 모든 사용자와 개발자들이 요구사항 명세서의 내용을 이해하고 동의하는 것이 중요하다. 요구사항 최종 검토를 통해서 합의 된 요구사항 명세서는 설계 베이스라인을 형성하게 되며 향후 설계의 기초가 된다.

또한 요구사항의 추적성 측면에서 아래와 같은 양방향 추적성 검토를 하였다[10].

- 변경 영향 분석
  - 요구사항 변경에 영향을 받는 모든 산출물
  - 모든 산출물에서 문제점 및 변경에 의해 영향을 받는 모든 요구사항
- 요구사항의 현재 상태 평가
  - 누락 요구사항 식별
  - 요구사항에서 나타나지 않는 산출물 식별

본 연구에서는 요구사항 명세서 초안을 정련하여 작성한 요구사항 명세서 수정본을 월성 발전소의 각 담당자들이 검토하였으며 소프트웨어 검증 전문업체에서 일반적인 검증 기준에 따른 검증을 수행하였다. 월성 발전소에서는 그 동안 협의한 결과가 적절히 반영되어 있으므로 별 다른 수정 요청이 없었다. 이러한 결과에 따라 최종 검토된 요구사항 명세서를 베이스라인으로 설정하고 프로세스 설계 단계로 진입하였다.

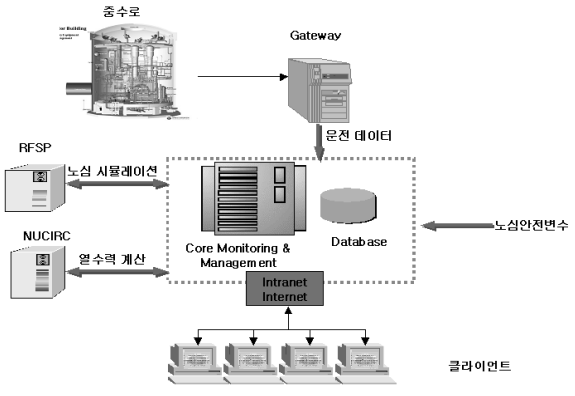

그림 8. 시스템 물리적 아키텍처

3.3 시스템 아키텍쳐 설계

시스템의 성공척도는 실제로 개발 시스템을 현장에서 사용하게 될 발전소 직원들의 다양한 요구사항을 얼마나 잘 만족시키느냐에 달려있다.

3.3.1 단위 기능 모듈 설계

기본설계는 요구사항 사양서를 상세설계 전 단계에서 구체화하는 과정이라 할 수 있다. 기본설계 과정에서는 유스케이스(Use case) 모델링과 사용자 인터페이스(Interface) 모델링이 이루어졌다[11, 12].

3.3.1.1 유스케이스 모델링

유스케이스 모델링은 주로 유스케이스 다이어그램에 의해 이루어진다. OMG(Object Management Group)에서 작성한 UML(Unified Modeling Language) 규격서에 기술된 유스케이스 모델링에 대한 설명은 다음과 같다.

유스케이스 다이어그램은 액터(Actor)와 유스케이스(Use Case) 간의 관계를 나타낸다. 액터는 사용자나 외부시스템이 될 수 있으며 유스케이스는 액터에게 시스템이 제공하는 기능으로 볼 수 있다. 유스케이스 다이어그램에서 액터는 일반화 관계에 의해 서로간의 관계를 표현할 수 있으며, 유스케이스 간에는 포함, 확장의 관계를 가질 수 있다.

유스케이스 모델링 부분에서는 유스케이스의 식별, 유스케이스 다이어그램 작성, 유스케이스 규격 작성 업무를 수행하였다. 유스케이스는 액터(Actor)가 시스템을 어떻게 사용할 수 있는지를 나타낸 것으로 볼 수 있다. 본 연구에서의 액터는 각 절차서 업무의 담당자로 볼 수 있으며 이 담당자들에게 필요한 유스케이스는 절차서에 규정된 업무를 수행하는 것이므로 절차서를 기준으로 유스케이스를 식별하였다.

유스케이스 다이아그램을 이용한 모델링의 한 예로서, 그림 9는 신연료 관리에 대한 유스케이스 다이어그램을 보여주고 있다. 그림 9에서 액터는 노심 업무 담당자이며 신연료 인수, 하역, 저장을 위한 하부 유스케이스들이 나타나 있고 이들 간의 포함

및 확장관계가 표현되어 있다. 또한 각 유스케이스에 대한 커멘트도 볼 수 있다. 유스케이스 다이어그램은Borland사의 Together 2005 for VS .NET을 사용하였다.

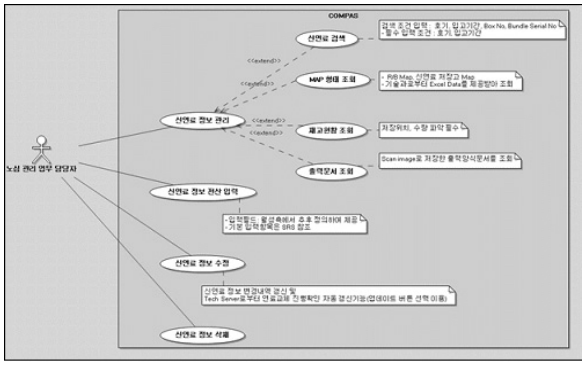

그림 9. 신연료 관리 유스케이스 다이어그램(예시)

중수로 노심관리 현안 자동화 시스템은 다수의 사용자가 별개 혹은 공동의 작업을 수행하는 것을 지원해야 하며 노심관리 업무 특성 상 입출력 데이터의 무결성(integrity) 및 정확성(accuracy) 확보와 시스템의 가용성(availability)이 요구된다. 본 시스템에서는 Microsoft 사의 .NET 프레임워크를 기반으로 한 ASP.NET 기술을 사용하여 .NET 표준개발 가이드에 따라 웹 기반 시스템을 설계하였다. .NET에서의 웹 애플리케이션은 ASP.NET으로 구현되며, 클라이언트는 웹 브라우저를 통해서 애플리케이션에 접근하게 된다. 그러므로 자연히 웹 서버가 요구된다. 웹 애플리케이션의 장점은 클라이언트 컴퓨터에 별다른 설치가 필요 없으며, 언제 어디서나 인터넷 접속이 가능하면 애플리케이션에 접근 가능하다는 것이다. 또한 서버 쪽만 변경해주면 되므로 애플리케이션 유지 보수가 매우 쉽다. 웹 애플리케이션의 단점은 풍부한 기능과 UI(User Interface)를 제공하지 못한다는 한계점이 있다는 것이다. 그러므로 불특정 다수의 사용자(특히 외부 사용자)를 대상으로 하는 애플리케이션의 경우 웹 애플리케이션으로 구현되는 것이 바람직하며, 복잡한 UI를 필요로 하는 업무용 애플리케이션에는 다소 적합하지 않다.

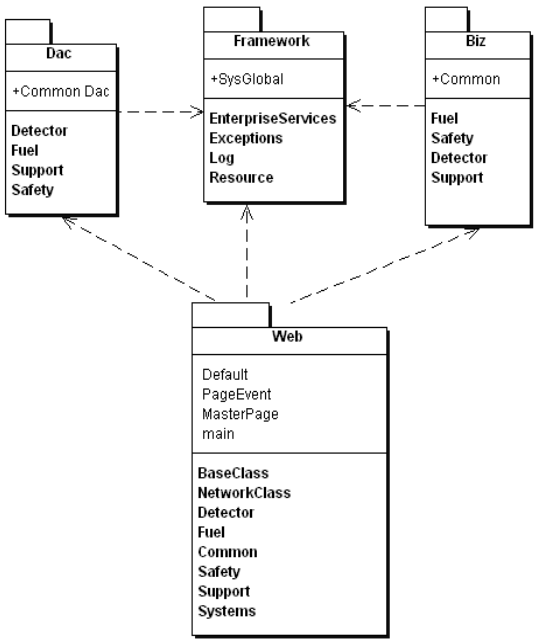

그림 10. 기반 클래스 다이어그램

따라서 본 시스템에는 스마트 클라이언트(Smart Client)를 이용하여 복잡한 기능이 요구되는 모듈을 구현하고자 한다. 스마트 클라이언트는 웹 애플리케이션의 배포 및 유지보수의 장점과 Windows Form 애플리케이션의 풍부한 UI 및 기능을 결합한 것으로, High-Fidelity Client라고 부르기도 한다. 스마트 클라이언트는 Windows Form 애플리케이션을 웹 서버에 올려놓는 것만으로 애플리케이션의 배포 및 버전관리가 가능하다. 스마트 클라이언트 시나리오에서는 Windows Form과 마찬가지로 클라이언트 컴퓨터에 .NET Framework가 필요하였다.

.NET 기반의 웹 애플리케이션을 효과적으로 구현하기 위해 필요한 기반 클래스를 설계하였으며, 그림 10에서 보는 바와 같이 이 클래스에 파생되는 클래스를 붙여 나가는 방식으로 세부 클래스를 설계하였다.

3.3.2 데이터베이스 설계

정보시스템은 조직의 운영과 의사결정에 필요한 정보를 제공하기 위한 조직 전체적으로 통합된 정보이용자와 컴퓨터의 결합시스템이다. 이러한 정보

시스템의 개발은 일반적으로 시스템 개발 수명주기(Systems Development Life Cycle, SDLC)라고 불리는 방법론을 따라 진행되며, 정보시스템 개발의 일부분인 데이터베이스 개발 프로세스 또한 시스템 개발 수명주기 방법론을 따라 진행된다.

데이터베이스 개발 수명주기는 소프트웨어 시스템 개발에 있어 데이터베이스의 효율적 설계는 시스템 개발에 있어 아주 중요하다. 그러므로 데이터베이스 개발 또한 시스템 개발 프로세스와 연계되어 수행되어야 한다. 데이터베이스의 개발과정에 이러한 개발 프로세스를 따라야 하는 것은 정보시스템 구축을 위해서는 안정적인 데이터의 구축과 관리가 우선적으로 요구되는 조건이기 때문이다. 따라서 과거의 프로세스 중심적, 애플리케이션 로직 중심적, 파일 구조 중심적인 데이터베이스 구조 보다는 관계형 데이터베이스를 이용한 데이터 중심적인 시스템 개발 방법이 발달하게 되었다. 데이터 또는 데이터베이스의 분석과 설계는 관계형 데이터베이스 관리 시스템(Relational Database Management System, RDBMS)의 등장으로 보다 복잡해지고 체계화되고 있다.

3.3.2.1 데이터 모델링

데이터베이스를 구현하기 위한 사전 단계로서 DBMS에 독립적이며, 순수한 업무관점의 개략적인 데이터 모델링을 하였다. 요구사항 분석을 토대로 만들어지는 모델은 다음 상세 설계 단계인 논리적 데이터 모델의 바탕이 된다. 각 주제 영역별로 구체적인 속성이 드러나지 않은 정도의 개략적 ERD (Entity Relationship Diagram) 정도만 그려진다. 개념 데이터 모델은 추상화된 최상위 수준의 데이터 설계라고 할 수 있다. 개념 데이터 설계를 위해 중수로 노심관리 업무는 핵연료관리, 원자로 안전관리, 핵검출기 관리 등으로 분류하였고, 각각의 업무 구분에 대해 도출된 키워드를 토대로 데이터베이스 레이아웃(Lay-Out)을 구성하였다.

개념 데이터 모델은 정규화(Normalization)를 해주어야 바람직한 모형이 된다. 직관적으로 엔티티에 속성을 배치하다보면 이를 테이블로 옮길 때 RDBMS의 무결성을 손상시킬 수 있다. 따라서 정보의 손실이나 불필요한 정보의 도입 없이 일관성과 최소한의 중복 및 최대의 안정성을 확보하기 위해 데이터 구조를 정규화하게 된다. 되도록 엔티티(테이블)들을 쪼개는 작업이다. 정규화는 개념 모델링이 어느 정도 이루어진 후 하는 것이 바람직하겠지만 ERD를 어느 정도 완성하다보면 1, 2차 정규화는 알게 모르게 이루어진다. 또한 정규화를 집중하여 하더라도 3차 이상의 정규화는 하지 않아도 큰 문제는 없다. 정규화의 이점을 다음과 같이 정리할 수 있다.

- 데이터 저장공간을 최소화한다.
- 데이터베이스에서 데이터가 불일치할 위험을 최소화시킨다.
- 갱신 및 삭제시의 예외발생을 최소화한다.
- 데이터 구조의 안정성을 최대화한다.

3.3.2.2 데이터베이스 상세 설계

개념 데이터 설계가 데이터베이스 관리 시스템(Database Management System, DBMS) 및 하드웨어 구조와 완전히 독립된 것이라면 논리적 설계에서 만들어지는 모델은 이 개념적 모델을 DBMS가 처리할 수 있도록 맵핑하는 과정이라고 보면 된다. 개념 데이터 모델을 통해 구성한 데이터베이스 레이아웃을 바탕으로 요구사항 분석과정에서 도출한 키워드를 이용하여 각 테이블에 포함될 필드를 정의하였다. 그림 11은 논리적 데이터베이스 설계의 일부로서 데이터베이스 스키마와 객체관계도(Entity Relationship Diagram, ERD)이다[13]. 관계형 데이터베이스가 DBMS의 주를 이룸에 따라 ERD(Entity Relationship Diagram)의 작성은 데이터베이스 설계에서 가장 중요한 요소가 되었다. 이러한 데이터베이스의 설계과정을 하향식(Top-down) 접근방법이라고 하는데, 이는 상위의 추상적인 스키마에서 출발하여 보다 구체적인 스키마로 발전시켜 나가는 전략이다. 이 방법은 주로 복잡한 데이터베이스를 설계할 때 적합한 방법이다.

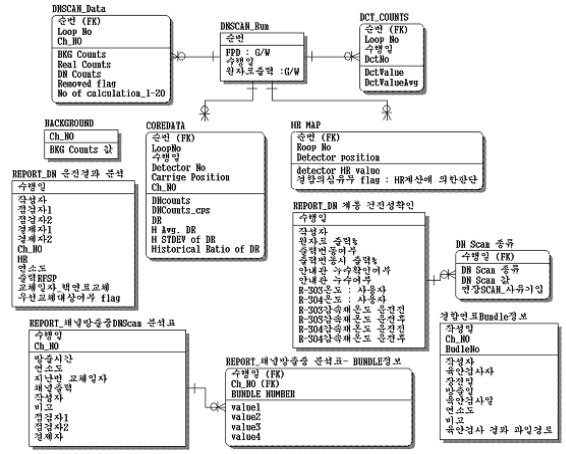

그림 11. 논리적 데이터베이스 설계

그림 12. 물리적 데이터베이스 설계

물리적 설계는 시스템을 고려한 논리적 설계를 구현이 가능한 물리적 데이터베이스 구조로 전환하고 DBMS의 조건에 맞게 성능을 최적화하는 과정이라 할 수 있다. 그림 12는 논리적 데이터 스키마를 구체화하여 작성한 물리적 데이터 스키마를 ERD로 나타낸 것이다.

3.4 시스템 구현 및 검증

시스템의 구현에 있어 Visual Studio 2005를 사용하여 .NET 기반으로 하였으며, 개발 과정에서 소스 형상관리를 위해 Visual Source Safe (VSS)를 사용하였다.

구현 완료 후, 최상위 시스템 기능 요구사항의 만족 여부를 검증하기 위해 월성 원자력 1발전소에서 기존의 업무 수행 방식 결과와 비교 검토를 수행하였다. 즉, 시스템 기능 요구사항에 따른 검증 요구사항에 따라 구현된 기능을 통합하여 웹 서비스를 통해 시스템을 구동하여 시험한 후, 현장에 설치 및 운영을 시작하여 노심관리업무에 실제 적용하고 현장 업무데이터와의 비교를 통해 시스템 개발결과를 검증하였다.

구체적인 검증은 표 3에서 나타난 바와 같이 상위 기능 요구사항에 따른 각 노심관리 업무별로 기존의 업무방식에 의한 업무자료와 COMPAS를 이용한 업무자료 결과를 비교하고 서로 결과가 상이할 경우, 원인을 분석하여 수정 및 보완하는 방식으로 진행하였다.

일차적으로, 실제 PHWR 형인 월성 원자력 1발전소 1호기의 운전상태에 대한 시뮬레이션을 통해 시스템의 설계에 따른 검증을 핵연료 교체시점에서 23회 실시하였다. 현장 취득 데이터를 이용해 핵연료 교체에 따른 각 기능 요구사항의 검증을 실시하였다. 그 중 채널의 중성자속 평가 기능 검증 결과, 핵연료 교체 중에 오차가 증가하지만 3%이내 범위에서 평가 기능이 수행되고 있음을 알 수 있었다.

표 3. 상위기능 요구사항에 따른 검증 요구사항

출력감시	출력감시	– 채널/번들별 출력 분포 감시 및 출력 이력 관리 – 검출기 및 제어장치 위치 감시 – 현장 설치 시험
연료관리	연료교체 신연료관리	– 핵연료 교체 선정 프로그램, 핵연료 교체 이력, 월간 핵연료 교체 이력 일정 작성 및 관리

	연료건전성 확인	– 신연료 입고 및 재고 현황 관리 – DN Scan 분석 파일 서버 관리 – 현장 설치 시험
안전관리	ROP 검출기 교정 냉각재 유량확인 열출력 점검	– ROP DC 검출기 교정값 계산 – 노심 냉각재 유량 확인 – 열출력 계산 및 점검 – 현장 설치 시험
검출기 관리	Zone Pt. 점검 표준영역 중성자속 변경 Vanadium 검출기 교정	– 노심데이터 수집 및 Zone Pt. 검출기 교정 여부값 계산 – 표준영역 중성자속 변경 여부 계산 및 변경전/후 결과 비교 – Vd 검출기의 S Factor, Kj Factor 및 검출기 교정 여부 계산 및 교정 결과 비교 – 현장 설치 시험
데이터통합 관리	데이터통합 관리	각 기능에서 검증 – 현장 설치 시험

4. 결 론

가압중수로 노심관리에 있어 원자력발전소의 가동성과 안전성을 향상시킬 목적으로 발전소 현장의 데이터 파일을 활용하여 노심관리 지원 시스템을 개발하였다.

시스템은 가압중수로 운전 중 수행하는 노심관리는 원자력발전소를 효율적으로 관리할 목적으로, 사용자의 요구에 의해 5대 주요 기능이 시스템 최상위 기능 요구사항을 만족시키는 범위에서 정의되었다. 정의된 요구사항은 노심관리절차서 및 사용자와의 회의를 통해 세부 요구사항이 도출되어, 원자력측면에서 분석되었다. 분석된 요구사항들은 IEEE 830에 따른 개선된 사용자 요구사항 명세의 템플릿을 이용하여 시스템 요구사항 명세서에서 정리되었고, 이에 따라 시스템의 모델링 및 설계가 이루어졌다.

이 과정에서 기능 추출 템플릿을 통해 작성한 데이터베이스 모델의 활용도가 전체 89개의 테이블 중 80개에 이르렀으며, 입, 출력 데이터를 통해 도출한 테이블 필드는 누락 없이 모두 사용되었다.

이러한 결과는 시스템 요구사항의 수렴이 효과적으로 이루어졌음을 나타내고 있으며, 시스템 공학 측면에서 요구사항 추출, 분석, 문서화에 대한 수행 사례를 제시하였다. 그리고 시스템 요구사항 명세서 작성을 통해 원자력분야에서 노심관리 지원시스템의 최상위 규격서 개념의 문서를 제시했다.

또한 요구사항 분석 과정을 통해 가압중수로 노심의 출력 상시 감시에 있어서 애로 사항을 해결함으로서 노심 출력 상시 감시 방법론을 제시하였다.

마지막으로 시스템의 시험을 통해 살펴보면, 기존의 수행시간을 최대한 90%이상 단축시킨 예와 같이 노심관리의 표준 체계를 제시하였다.

References

[1] INCOSE Systems Engineering Handbook Ver.3.2 INCOSE-TP-2003-002-03.2, 2010

[2] 김진일, 김진훈, 박중용, 이중윤, 시스템엔지니어링 프로세스 EIA-632, 2006. 04.

[3] 월성 1 발전소 노심관리 업무 절차서(술 0-8-**) - 월성 원자력 본부 제 1발전소 안전부

[4] PC-based CANDU DCC Emulator System -DCC Program Development User Manual, AECL, 1995, Canada

[5] Bill et al, “Error Cost Escalation through the Project Life Cycle”, INCOSE

Symposium, 2004

[6] 김진일, 염충섭, “중수로 노심관리 자동화시스템의 요구사항의 추출 사례에 관한 연구,” 2006 한국 시스템엔지니어링협회 춘계워크샵, 2006

[6] Harold J. Jeydt, Tools for Requirement Discovery, Creation and Elicitation, INCOSE Prodeedings, 2002 – 도제법

[8] IEEE-SA Standard Board, IEEE Std 830-1998 IEEE Recommended Practice for Software Requirement Specification, 1998.

[9] US NRC, Regulatory Guide 1.171-Software Unit Testing for Digital Computer Software Used in Safety Systems of Nuclear Power Plants

[10] CMMI for Systems Engineering/Software Engineering, Version 1.02 (CMMI-SW/SW, V1.02); CMMI Staged Representation, CMU/SEI-2000-TR-018, ESC-TR-2000-018; Continous Representation, CMU/SEI-2000-TR-019, ESC-TR-2000-019; Product Development Team; Software Engineering Institute; November 2000. 3

[11] 한/카네기멜론대학 기술교류협회, 최신 소프트웨어 공학 기법, pp. 12 - 60, 2002.

[12] OMG, UML 2.0 Superstructure Specification, Adopted Specification, pp. 511, 2003

[13] Chen. P., “The Entity-Relationship Model-Toward a Unified View of Data,” ACM Transaction on Database System, pp. 9-36, Vol. 1, No.1, March, 1976

사 사

이 논문은 2013년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(원자력연구개발사업, No. NRF – 2012M2A8A4011779)