

## 발전플랜트 소프트웨어 플랫폼의 요구사항 및 개념구조

주용재\*, 김선익\*, 김해순\*, 김지현\*  
한국전력공사 전력연구원\*

### Requirements and a Conceptual Architecture of a Software Platform for Electric Power Plants

Yong Jae Joo\*, Sun-ic Kim\*, Hae-soon Kim\*, Ji hyun Kim\*  
KEPCO Research Institute\*

**Abstract** - 최근 스마트폰 등의 산업에서와 같이 발전플랜트에서도 가치창출의 중심이 하드웨어 제조·관리 기술에서 최적 제어·운용의 소프트웨어로 급속히 이동하고 있다. 국내외 발전사 및 관련 연구기관들은 발전플랜트 운용최적화를 위한 다양한 제어, 감시, 분석, 설계, 시뮬레이션, 자원관리 등의 다양한 분야에서 독립적인 소프트웨어 개발을 활발히 수행하고 있으나, 이들 간의 데이터, 이벤트 통신 및 서비스 연동은 미흡한 실정이다. 본 연구에서는 발전플랜트 관련 소프트웨어의 서비스 통합 기반을 마련하고, 안정성·제사용성·이식성 등의 품질을 제고하며 개발자가 쉽고 빠르게 소프트웨어를 구현할 수 있는 발전플랜트 소프트웨어 플랫폼을 제안한다. 이는 화력, 원자력 등의 중대형 발전플랜트 뿐만 아니라 점점 확대되고 있는 태양광, 태양열, 풍력 등 크고 작은 다양한 신재생에너지 발전기에 적용가능하며 발전 효율 및 안정성 향상과 발전플랜트 소프트웨어 산업의 활발한 생태계 조성에 기여할 것으로 기대된다.

산업에서 플랫폼기반 제품 경쟁이 강화되고 있다. 소프트웨어 플랫폼이란 하드웨어에 탑재되어 소프트웨어가 하드웨어의 기능을 사용하고, 다른 소프트웨어와 서비스를 주고받을 수 있도록 정의된 운영체제(OS: Operating System), 커널, 하드웨어 드라이버, 라이브러리, 미들웨어 및 API(Application Programming Interface: 응용프로그램 개발에 필요한 표준 Source Code) 등의 집합 체계를 의미한다[6][7]. 발전플랜트의 제어, 감시, 예측, 분석, 관리, 운영 등을 위한 수많은 소프트웨어들은 데이터 공유 및 이벤트 메시지 공유 등의 시스템 통합을 통해 더 높은 신뢰성, 효율성, 안전성을 확보할 수 있다. 기존에도 데이터 및 이벤트 메시지 연계 등을 통한 통합 시도가 이루어져 왔으나, 서로 독립적인 시스템·데이터 구조 채택, 과도한 네트워크 부하 및 통신 지연 발생, 보안 체계 미흡 등의 문제로 인해 완전한 충분한 연계가 이루어지지 못하고 있는 것이 현실이다. 발전플랜트 소프트웨어 통합의 완전성과 효율성을 위해서는 표준화된 소프트웨어 플랫폼의 구축이 필요하다.

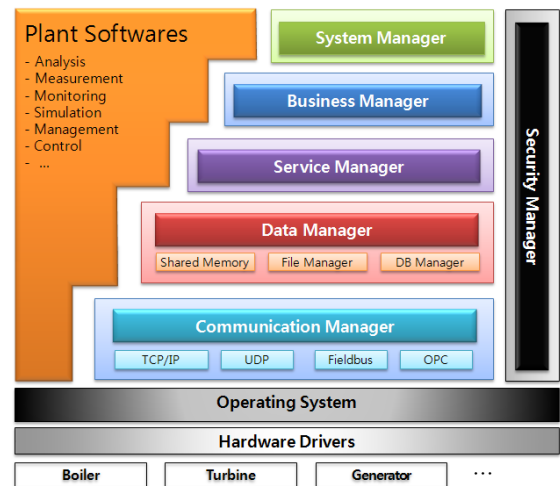
## 1. 서 론

최근 스마트폰, PC, 인터넷, 자동차 등의 산업에서 제품의 가치와 경쟁력이 설비나 기기 등의 하드웨어의 설계·제조·조립 기술에서 하드웨어를 효율적으로 제어, 운용, 사용하는 소프트웨어 기술로 급격히 이동하고 있다. 국내외 화력·원자력 발전플랜트에서도 보일러, 터빈, 발전기 등 하드웨어의 설계 및 제조 기술보다 여러 설비, 장치, 계통을 통합적으로 감시·예측·제어하여 안정적이고 효율적으로 운용하는 소프트웨어 기술이 핵심 경쟁력으로 요구되고 있다. 정보통신(ICT: Information and Communication Technology) 기술의 발달로 발전플랜트에도 제어, 감시, 분석, 설계, 시뮬레이션, 자원관리, 운전훈련 등 수많은 소프트웨어들이 개발되고 있다. 그러나 하드웨어 제조사 및 여러 소프트웨어 개발 기관에서 개발하는 시스템 소프트웨어 또는 응용 소프트웨어들은 독립적인 구조, 데이터, 통신방식으로 구현되기 때문에 운용최적화를 위한 통합연계 서비스가 매우 어려운 뿐만 아니라 이식성, 유지보수성, 신뢰성 등 소프트웨어 품질 및 개발 생산성이 높지 못한 것이 현실이다. 이에 최근 화력발전소 통합감시제어시스템[1][2] 및 수력발전소의 통합운영시스템[3]의 개발 및 검증에 관한 연구가 이루어지고 있다. M. Strembeck and O.U.Phal[4]은 오픈 표준 및 기술을 적용하여 분산 발전플랜트의 이종 제어시스템을 통합제어할 수 있는 웹기반 통합 플랫폼을 구축하였다. 이러한 기존 연구는 대부분 기존의 이종 제어 및 응용 소프트웨어의 통신 및 데이터 전송 체계를 표준화하여 연계를 강화하고 서비스를 제공하는 것을 목표로 한다. 본 연구에서는 다양한 발전플랜트 소프트웨어의 안정성, 이식성, 제사용성을 증대하고 생산성을 확보하며 더욱 다양하고 활발한 발전플랜트 소프트웨어 산업의 생태계를 조성하기 위한 발전플랜트 소프트웨어 플랫폼을 제안한다.

## 2. 본 론

### 2.1 소프트웨어 플랫폼 구축의 필요성

플랫폼이란 “다양한 용도에 공통적으로 활용할 목적으로 설계된 유품형의 구조물”로서 공통의 활용 요소를 바탕으로 보완적인 파생 제품이나 서비스를 개발·제조할 수 있는 기반을 의미한다[5][6]. 플랫폼 기반의 제품을 개발하면 부품 표준화 및 공용화, 소프트웨어 제사용 등으로 개발비용 절감뿐만 아니라 급속히 변화하는 요구에 대응하여 적기 생산이 가능하며, 오픈 이노베이션을 통해 제품 혁신을 빠르게 달성할 수 있다[6]. 이러한 장점으로 인해 과거에는 주로 자동차나 PC 등의 하드웨어 산업에서 부품 및 서비스 체계 표준화 및 공용화를 통한 플랫폼 기반 제품 개발이 성행했으나 최근에는 인터넷 스마트폰, 스마트 TV 등의 스마트 디바이스나 Facebook(<http://www.facebook.com>), Amazon(<http://www.amazon.com>) 등의 SNS(Social Network Service)와 인터넷 비즈니스, 그리고 Mplayer(<http://www.mplayerhq.hu/>), VLC(<http://www.videolan.org/>) 등의 미디어 플레이어에 이르기까지 광범위한 소프트웨어



〈그림 1〉 발전플랜트 소프트웨어 플랫폼의 개념 구조

### 2.2 발전플랜트 소프트웨어 플랫폼 요구사항

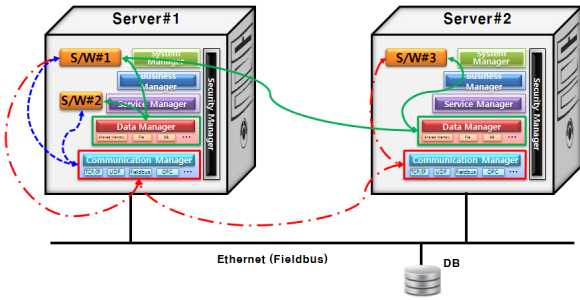
발전플랜트 소프트웨어 플랫폼은 다음과 같은 요구사항을 만족하도록 설계 및 개발되어야 한다.

- 안정성: 발전플랜트의 특성상 제어, 감시, 운영 소프트웨어의 안정성은 필수적으로 보장되어야 한다. 다양한 환경의 현장에서 발생할 수 있는 시스템 메모리, 데이터 입출력 및 저장, 통신 등의 장애 등 이상 상황에 강건하고 유연하게 대응하며 자가복구 또는 관리자로부터 신속한 대처가 가능하도록 설계되어야 한다.
- 연계성·연동성: 발전플랜트 소프트웨어 모듈 사이의 이력 및 실시간 데이터 공유, 이벤트 메시지 송수신, 서비스 연동이 가능하여야 한다. 단, 데이터 및 메시지의 용량에 따라 네트워크 및 서버·PC 등의 자원에 안정성을 저해하지 않도록 설계되어야 한다.
- 신뢰성: 연계하는 소프트웨어 간에 주고받는 데이터 및 이벤트 메시지뿐만 아니라 상호간 계약되고 합의된 서비스의 정합성, 무결성이 보장되는 체계를 갖추어야 한다.
- 개방성: 구축되는 플랫폼은 보일러, 터빈, 발전기 등 설비 제조업자, 계측기, 탐지장치 등의 기기 조립업자 등 하드웨어 임베디드 소프트웨어 개발자 및 시뮬레이터, 구조해석, 유체해석 등의 응용 소프트웨어 등 관계된 모든 개발 기관 및 개발자가 쉽게 해당 소프트웨어를 개발할 수 있도록 API, 개발도구(SDK) 등이 제공되어야 할 뿐만 아

나라, 표준 플랫폼의 설계 및 개발에도 참여하고 개선할 수 있는 기회가 제공되어야 한다.

- 재사용성·이식성: 플랫폼 기반으로 개발된 소프트웨어가 서버, PC, 모바일기기, 임베디드 등 다양한 하드웨어 환경에서 구동 가능해야 하며, 개발된 소프트웨어의 일부 모듈 또는 컴포넌트가 다른 소프트웨어에서도 활용될 수 있어야 한다.
- 무결성: 화력 및 원자력 발전플랜트의 제가동에 오랜 시간과 막대한 비용이 소요되는 특성상 새로운 소프트웨어의 추가, 삭제, 업데이트 등의 작업이 서버 시스템의 중단이나 재시작 없이 이루어지고 연동 실행이 가능해야 한다.
- 사업성: 더욱 많은 소프트웨어 개발자의 자발적 참여를 유도하기 위해 소프트웨어 간 교환되는 데이터, 이벤트 메시지, 서비스를 통해 비용 및 수익을 처리할 수 있어야 한다.

전술한 사항 외에도 일반적으로 요구되는 특성인 보안성, 확장성, 유연성, 편의성, 경량성 등의 요구사항을 만족해야 한다.



〈그림 2〉 플랫폼기반 소프트웨어 통신 및 데이터 공유 방식

### 2.3 발전플랜트 소프트웨어 플랫폼 개념 구조

분석된 요구사항을 만족할 수 있도록 발전플랜트 소프트웨어 플랫폼의 개념적인 구조를 <그림 1>과 같이 제안한다. 본 플랫폼의 구조는 <그림 1>의 운영체제 및 하드웨어 드라이버를 제외한 6개 Manager로 구성되어 있다. 본 플랫폼에서 구현되는 소프트웨어는 Java 언어를 기반으로 하며, JVM(Java Virtual Machine) 상에서 동작하기 때문에 하드웨어 및 운영체제에 독립적이다. 플랫폼을 구성하는 각 Manager의 기능 및 특성은 다음과 같다.

- Communication Manager: 같은 서버 내에 있거나 서로 다른 서버에서 실행중인 플랫폼 기반 소프트웨어 간 통신 서비스. 플랫폼 기반 소프트웨어는 실제 동작되는 통신방식(OPC(Object Linking and Embedding for Process Control), TCP/IP, UDP, Fieldbus 등)에 관계없이 플랫폼에서 제공되는 표준 방식으로 통신 가능
- Data Manager: 플랫폼 기반 소프트웨어 내부 또는 소프트웨어 간 데이터 관리 및 공유 서비스. 데이터 종류, 용량, 생성 및 조회 주기 등의 특성에 따라 공유 메모리, 데이터베이스, 파일 등의 방식으로 관리 및 공유
- Service Manager: 플랫폼 기반 소프트웨어의 기능 및 서비스(입출력 정보, 제약사항 등) 관리. 계약 사항 등에 의한 서비스 접속 권한에 따른 통신 및 데이터 서비스 연계 관리. 소프트웨어 구동, 정지, 업데이트 등 서비스 라이프사이클 관리
- Business Manager: 소프트웨어 간 서비스 계약, 과금 등 서비스. 계약 기간, 용량(메시지, 데이터 등), 동시접속수 등의 계약 사항에 따른 비용 및 수수료 관리 서비스
- System Manager: 발전플랜트 소프트웨어 플랫폼 운영자를 위한 관리 서비스. 플랫폼 변경, 업데이트, 소프트웨어 권한 관리 등 서비스
- Security Manager: 소프트웨어의 보안 서비스. 소프트웨어 간 계약 사항 또는 보안 등급에 따라 허가되지 않은 접속 차단.

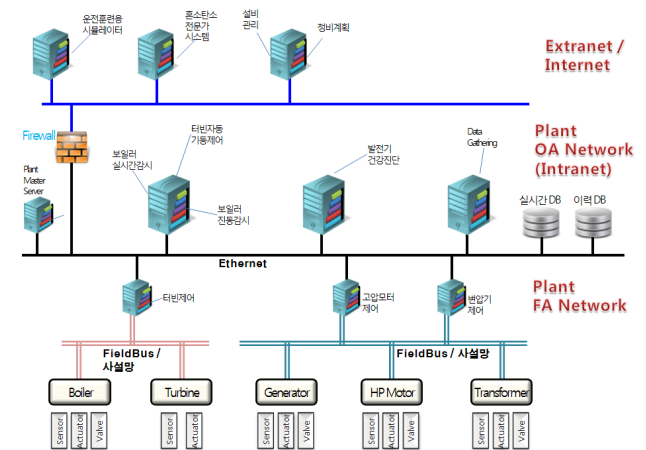
<그림 2>는 플랫폼 기반 소프트웨어의 통신 및 데이터 공유 방식을 설명한다. 발전플랜트 소프트웨어 플랫폼은 한 대의 서버(또는 PC)에 한 개씩 구동되며, 한 개의 플랫폼에서는 복 수의 소프트웨어가 동작될 수 있다. <그림 2>의 왼쪽 서버(Server#1)에는 두 개의 소프트웨어(S/W#1, S/W#2)가 실행 중이고, 오른쪽 서버(Server#2)에는 한 개의 소프트웨어(S/W#3)가 실행 중이다. S/W#1과 S/W#2는 같은 플랫폼의 Communication Server를 통해 통신하며, Data Manager를 통해 데이터를 공유한다. Server#1과 Server#2는 Ethernet 또는 Fieldbus에 연결되어 있고, 서로 다른 서버에 있는 S/W#1과 S/W#3은 각자의 플랫폼에 구현된 Data Manager와 Communication Manager를 통해 데이터 및 통신 메시지를 서비스한다.

<그림 3>은 발전플랜트 소프트웨어 플랫폼 기반 소프트웨어들이 탑재된 가상의 상황을 나타낸다. 발전플랜트 통신망은 하드웨어 센서 및 계측 정보가 전달되는 FA(Factory Automation) 망과 감시 및 통제, 통합 제어 및 운영최적 등의 여러 응용 소프트웨어의 통신 이벤트 및 데이터

가 전달되는 OA(Office Automation) 망으로 구분되며, 외부의 인터넷 망은 방화벽을 통해 연결되어 있다. FA 망에 있는 보일러, 터빈, 발전기 등의 제어 시스템 DCS(distributed control system)와 OA 망의 진동 감시, 데이터 취득, 분석 시스템은 Service Manager를 통해 각각 필요한 데이터와 이벤트 종류를 찾아 해당 서비스를 제공하는 시스템과 플랫폼의 Communication Manager 및 Data Manager를 통해 메시지와 데이터를 공유한다. 방화벽을 통해 발전플랜트 통신망과 연결되어 있는 인터넷의 여러 응용 소프트웨어는 원격지에서 발전플랜트의 운영 상황을 파악하고 관리할 수 있다.

### 3. 결 론

발전플랜트의 효율적이고 안정적인 운영을 위해서 소프트웨어 간 통신 이벤트 메시지 및 데이터 서비스의 통합연계가 가능해야 하며 완전한 통합을 위해서는 소프트웨어 플랫폼 구축이 필수적이다. 본 연구에서는 발전플랜트 소프트웨어 플랫폼의 요구사항을 분석하고 개념 구조를 제안하였다.



〈그림 2〉 플랫폼기반 소프트웨어 설치 구조

### [참 고 문 헌]

- [1] 황재필, 박태훈, 홍석규, “화력발전용 통합감시제어시스템의 개발 및 검증”, 정보 및 제어 학술대회 논문집, 213-214, Oct. 2010
- [2] 김승민, 윤석문, 홍성진, 최준혁, 이주현, 육신균, “화력발전용 통합감시제어시스템의 개발 및 기능 검증”, 대한기계학회 논문집, 1164-1169, 2010.11
- [3] 옥연호, 박지근, 곽원구, 이재홍, “발전통합운영시스템 최적제어 및 시스템 구성에 관한 연구”, 대한전기학회 논문집, 1852-1853, Jul. 2010
- [4] Mark Strembeck, Otto U. Pilhal, “VIDIHIP - A Web Service based Integration Platform for Power plant Control Systems”, Proceedings of IEEE International Conference on Service-Oriented Computing and Applications(SOCA'07), 207-214, Jun. 2007
- [5] 최병삼, “성장의 화두, 플랫폼”, SERI 경영노트(제 80호), 1-11, Nov. 2010
- [6] 박동욱, “소프트웨어 플랫폼 경쟁의 전면화와 대응 방안”, LG 경제연구원 Business Insight 리포트, 2-15, Aug. 2008
- [7] David S. Evans, Andrei Hagiu, Richard Schmalensee, “Invisible Engines: How Software Platforms Drive Innovation and Transform Industries”, MIT Press, Apr. 2008