

계산과학공학 플랫폼의 워크벤치 기반 동적 서비스 설계 및 구축에 관한 연구[☆]

A Study on Workbench-based Dynamic Service De-sign and Construction of Computational Science Engineering Platform

권 예진¹ 전 인 호¹ 마 진¹ 이 식¹ 조 금 원¹ 서 정 현^{1*}
Yejin Kwon Inho Jeon Jin Ma Sik Lee Kum Won Cho Jerry Seo

요 약

계산과학공학 소프트웨어 플랫폼인 EDISON(Education-research Integration through Simulation On the Net)은 클라우드 컴퓨팅 기반 웹서비스로서, 해석 소프트웨어를 이용한 웹 기반 시뮬레이션 서비스이다. EDISON에서는 사용자의 로컬 자원으로는 해석이 어렵거나 불가능한 계산과학공학 해석 문제를 웹 서비스를 통해 사용자에게 제공하며 미리 구축되어 있는 인프라 및 다양한 계산 노드를 통해 사용자에게 해석 결과를 제공한다. 그 결과로 사용자의 디바이스, 운영체제에 관계 없이 EDISON에 접근하여 계산과학 공학 해석 시뮬레이션을 수행할 수 있도록 웹 포털 환경에서 시뮬레이션 실행 환경을 제공한다.

본 연구의 목적은 현재 사용자에게 제공되고 있는 계산과학 공학 시뮬레이션 및 해석 결과 분석 플랫폼인 EDSION 시스템에 통합적인 유저 인터페이스를 제공하기 위한 워크벤치 기반 실시간 동적 서비스 설계 및 구축에 있다. 또한 로컬에서 사용하는 계산과학 공학 시뮬레이션 소프트웨어 환경과 유사한 사용자 유저 인터페이스를 가지며 다양한 해석기와 전처리기, 해석 시뮬레이션 소프트웨어 등의 동적인 웹 환경을 구성할 수 있는 워크벤치 기반 사용자 시뮬레이션 서비스 환경 구성에 대해 고안하였다. 이러한 웹 서비스를 제공하기 위해 사용자에게 필요한 서비스를 포틀릿 단위로 구성하였으며, 그 결과로 워크벤치를 이용한 시뮬레이션 서비스를 구축하였다.

☞ 주제어 : 클라우드 컴퓨팅, 서비스 포틀릿, 워크벤치, 프로토콜, 이벤트 처리

ABSTRACT

EDISON (Education-research Integration through Simulation On the Net) is a web simulation service based on cloud computing. EDISON provides that web service and provide analysis result to users through pre-built infrastructure and various calculation nodes computational science engineering problems that are difficult or impossible to analysis as user's personal resources to users. As a result, a simulation execution environment is provided in a web portal environment so that EDISON can be accessed regardless of a user's device or operating system to perform computational science engineering analysis simulation.

The purpose of this research is to design and construct a workbench based real - time dynamic service to provide an integrated user interface to the EDSION system, which is a computational science engineering simulation and analysis platform, which is currently provided to users. We also devised a workbench-based user simulation service environment configuration. That has a user interface that is similar to the computational science engineering simulation software environment used locally. It can configure a dynamic web environment such as various analyzers, preprocessors, and simulation software. In order to provide these web services, the service required by the user is configured in portlet units, and as a result, the simulation service using the workbench is constructed.

☞ keyword : cloud computing, service portlet, workbench, protocol, event processing

1. 서 론

클라우드 컴퓨팅 기술이 활성화되고 보편화되면서 사용자에게 소프트웨어를 웹 기반 서비스로 제공할 수 있는 기반이 마련되었다. 현재 사용되는 대부분의 계산과학 공학 소프트웨어는 사용자 로컬 환경에서 각 사용자의 컴퓨팅 구성에 특성화되어 있다. 이러한 계산과학 공학 소프트웨어를 웹 환경에서 제공하기 위해서는 클라우드

¹ Dept. of Computational Science Platform Center, Korea Institute of Science and Technology Information (KISTI), 305-806, Daejeon, 34141, Republic of Korea

* Corresponding author (jerry@kisti.or.kr)

[Received 8 January 2018, Reviewed 9 January 2018(R2 22 March 2018), Accepted 11 April 2018]

☆ 본 논문은 2017년도 정부(미래창조과학부)의 재원으로 한국연구재단 첨단사이언스·교육허브개발사업의 지원을 받아 수행된 연구임(No. NRF-2011-0020576).

기반의 플랫폼의 서비스가 필요하다. 클라우드 컴퓨팅 기반에서 소프트웨어 보급의 장점은 사용자의 환경 또는 운영체제에 관계 없이 해석 시뮬레이션 소프트웨어를 제공할 수 있다는 것이다. 또한 사용자에게 익숙한 소프트웨어 서비스를 클라우드 컴퓨팅 기반 서비스로 활성화하기 위해서는 사용자에게 익숙한 유저 인터페이스를 제공함과 동시에 적절한 서비스를 제공하여 접근 가능하도록 구성하여 제공하는 것이 필수적이다. [1,2].

최근 연구에 따르면 소프트웨어 플랫폼을 제공하는 것뿐만 아니라 사용자의 해석 시뮬레이션 결과를 공유하고 다양한 사용자와 교류하여 연구 할 수 있는 서비스의 필요성이 증대되었다. 계산과학공학 플랫폼을 통한 개인적인 연구를 지원하는 것이 아니라 다양한 연구자가 서로의 프로젝트 및 시뮬레이션 해석을 이용하여 융복합된 새로운 특화 연구분야를 재창조해낼 수 있는 서비스 지원 인프라의 요구가 증대되고 있다. 또한 다양한 연구 분야의 융합 및 복합된 새로운 연구 분야는 산업, 교육, 인프라 등에 새롭게 적용 될 수 있으며, 이러한 전문적인 연구자의 계산과학공학 해석 시뮬레이션 연구를 위해 클라우드 컴퓨팅 기반 사용자 지원 서비스의 연구가 진행되어 왔다[3-5]. 서로 다른 연구 분야의 계산과학공학 해석 결과를 공유하기 위해서는 통합적인 사용자 유저 인터페이스가 필요하다. 사용자에게 익숙하며 다양한 연구분야에 적합한 소프트웨어 구성 환경이 웹 서비스로 제공되어야 한다.

계산과학공학 소프트웨어 플랫폼인 EDISON(Education-research Integration through Simulation On the Net)은 클라우드 컴퓨팅 기반 웹서비스로서, 해석 소프트웨어를 이용한 웹 기반 시뮬레이션 서비스이다. EDISON에서는 사용자의 로컬 자원으로는 해석이 어렵거나 불가능한 계산과학공학 해석 문제를 웹 서비스를 통해 사용자에게 제공하며 미리 구축되어 있는 인프라 및 다양한 계산 노드를 통해 사용자에게 해석 결과를 제공한다. 그 결과로 사용자의 디바이스, 운영체제에 관계 없이 EDISON에 접근하여 계산과학 공학 해석 시뮬레이션을 수행할 수 있도록 웹 포털 환경에서 시뮬레이션 실행 환경을 제공한다.

본 연구의 목적은 현재 사용자에게 제공되고 있는 계산과학 공학 시뮬레이션 및 해석 결과 분석 플랫폼인 EDISON 시스템에 통합적인 유저 인터페이스를 제공하기 위한 워크벤치 기반 이벤트 모듈 커뮤니케이션 프로토콜의 설계 및 구축에 있다. 서비스를 제공하는 모듈간의 이벤트 커뮤니케이션을 통해 실시간으로 서비스를 조합 및 통합하여 사용자에게 시뮬레이션 소프트웨어 통합

환경을 제공할 수 있다. 또한 사용자가 로컬에서 사용하는 계산과학 공학 시뮬레이션 소프트웨어 환경과 유사한 사용자 유저 인터페이스를 가지며 다양한 해석기와 전처리, 해석 시뮬레이션 소프트웨어 등의 동적인 웹 환경을 구성할 수 있는 워크벤치 기반 사용자 시뮬레이션 서비스 환경 구성에 대해 고안하였다. 이러한 웹 서비스를 제공하기 위해 사용자에게 필요한 서비스를 포털 단위로 구성하였다.

본 논문에서 설계 및 구축한 워크벤치 기반 시뮬레이션 소프트웨어를 실행하면 실시간으로 필요한 서비스를 분석하고 포털 단위로 모듈화된 서비스를 조합하여 동적으로 구성되게 된다. 또한 이러한 해석 결과를 분석하고 다양한 사용자에게 시뮬레이션 환경을 공유할 수 있는 커뮤니티를 활성화하는 방안의 한 갈래로서 워크벤치를 이용한 시뮬레이션 서비스를 제공할 수 있다.

EDISON 시스템에서 웹 기반 워크벤치 서비스는 현재 EDISON에서 제공중인 다양한 계산과학 공학 해석 애플리케이션에서 제공하는 시뮬레이션 결과 및 전처리 결과를 효과적으로 사용자에게 전달 할 수 있으며, 전문가가 아닌 일반 유저라도 웹 서비스에 쉽게 접근하여 서비스를 사용할 수 있도록 사용자 인터페이스 환경을 구성하였다. 또한 기존에 사용되는 시뮬레이션 시스템에서의 리소스나 분산된 서비스를 통합적으로 관리하여 웹 서비스의 효율성을 증대하였다.

본 논문의 구성은 다음과 같다. 2장에서는 클라우드 컴퓨팅의 기본 개념과 EDISON 시뮬레이션 시스템의 전반적인 아키텍처에 대해 서술하였다. 3장에서는 EDISON에서 제공하는 전처리, 해석 시뮬레이션 소프트웨어, 해석기의 서비스 모듈화를 위한 이벤트 프로토콜에 대해 정의하였으며, 4장에서는 기존 EDISON 시뮬레이션 시스템의 메모리 사용량과 워크벤치 기반 이벤트 모듈의 메모리 사용량에 대해 비교 분석 하였다. 5장에서는 결론 및 향후연구를 서술하였다.

2. 관련 연구

2.1 클라우드 컴퓨팅

클라우드 컴퓨팅은 유비쿼터스, 편의성, 사용자 요구에 따른 네트워크 접근을 가능하게 만드는 컴퓨팅 리소스(네트워크, 서버, 저장소, 애플리케이션, 서비스 등)를 제공하는 공유 구성 모듈이다. 또한 클라우드 컴퓨팅은 최소한의 관리 또는 서비스 제공자와 상호 작용을 할 수 있는 모

텔이다. 클라우드 컴퓨팅은 다섯 가지의 필수 특징, 세 가지의 서비스 모델, 네 가지의 배포 모델로 구분될 수 있다 [1,4,5].

(표 1) 클라우드 컴퓨팅 필수 특징
(Table 1) Essential Characteristics of Cloud Computing

Characteristics	Definition
On-demand self-service	A consumer can unilaterally provision computing capabilities
Broad network access	Capabilities are available over the network and accessed through standard mechanisms
Resource Pooling	The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model
Rapid elasticity	Capabilities can be elastically provisioned and released
Measured service	Capabilities can be elastically provisioned and released

(표 2) 클라우드 컴퓨팅 서비스 모델
(Table 2) Service Models of Cloud Computing

Service Model	Definition
SaaS	Provides the capability of running providers' applications on a cloud infrastructure
PaaS	Provides ability to deploy consumer-created applications created using programming languages, libraries, services, and tools
IaaS	Provides computer processing, storage, networks, and other fundamental computing resources

(표 3) 클라우드 컴퓨팅 배포 모델
(Table 3) Deployment Models of Cloud Computing

Deployment Model	Definition
Private cloud	The cloud infrastructure is provisioned for exclusive use by a single organization comprising multiple consumers
Community cloud	The cloud infrastructure is provisioned for exclusive use by a specific community of consumers from organizations that have shared concerns
Public cloud	The cloud infrastructure is provisioned for open use by the general public
Hybrid cloud	A composition of two or more distinct cloud infrastructures

NIST(National Institute of Standards and Technology)에서는 클라우드 컴퓨팅에 대해 특징, 서비스 모델, 배포 모델에 대한 정의하였으며, 해당 내용은 표 1-3과 같다. 클라우드 컴퓨팅은 컴퓨팅 자원을 공유하거나 네트워크 자원을 통해 전달할 수 있는 시스템 서비스를 의미하며, 다양한 형태의 클라우드 컴퓨팅 모델이 존재 할 수 있다. 현재 대부분의 소프트웨어 서비스는 사용자 로컬 환경에서 제공되는 것이 아닌 사용자에게 서비스 단위로 제공되고 결과를 공유할 수 있는 커뮤니티를 형성하기 위한 연구가 진행되고 있다.

2.2 EDISON 시뮬레이션 시스템

웹 기반 모델링 및 시뮬레이션 시스템은 계산 과학 공학 전공 학생들을 위해 개발되고 웹을 통해서 서비스 되고 있다. 웹 기반 모델링 및 시뮬레이션 시스템은 학생에게 고성능 컴퓨팅 또는 높은 처리량 컴퓨팅 리소스를 제공하여 웹을 통해 문제를 해결하고, 계산 알고리즘을 통해 사용자에게 시뮬레이션을 수행하고 결과를 분석할 수 있는 서비스를 제공한다.

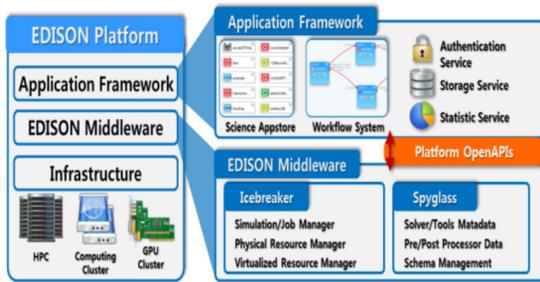
계산 과학 공학 분야의 교육은 수학적 모델[6]의 컴퓨터 응용 프로그램을 통해 시뮬레이션을 수행하고 해당 결과를 분석하며 과학에 대한 이해를 제공 한다. 또한 컴퓨팅 자원 및 전산 도구의 발전으로 인터넷 교육 기회가 다양화 되었기 때문에 계산 과학 공학 시뮬레이션 시스템을 전공하는 많은 사람들이 인터넷을 통해 이러한 시뮬레이션 시스템을 접근하여 무료로 학습할 수 있는 다양한 웹 기반 시뮬레이션 서비스 플랫폼이 증가하였다[10-12].

EDISON(Education-research Integration through Simulation On the Net)은 웹 환경에서 사용자에게 계산과학공학 분야 해석을 위한 시뮬레이션 환경을 제공하는 플랫폼이다. EDISON에서는 현재 전산열유체(Computational Fluid Dynamics, CFD), 나노물리(Computational Nanop-hysics, NANO), 계산 화학(Computational Chemistry, CHEM), 구조동역학(Computational Structural Dynamics, CSD), 전산설계(Computer Aided Opti-mal Design, DESIGN), 전산의학(Computational Medicine, CMED)의 6개 분야를 서비스 하고 있으며, 각 분야별로 계산과학 공학 분야 시뮬레이션 환경을 제공한다[7-10].

2.1.1 EDISON 시스템 아키텍처

그림 1은 EDISON 시스템 프로세스의 전체 시스템 아키텍처를 표현한 것이다. EDISON 계산과학 공학 플랫폼은 해

석 시뮬레이션을 지원하는 계산 자원(Infrastructure), 계산 자원을 관리하고 적절한 노드를 분배하는 미들웨어(EDISON Middleware), 해석 알고리즘을 수행하는 소프트웨어(Application Framework)로 나뉘질 수 있다. 계산과학공학 시뮬레이션을 수행하기 위해 EDISON 플랫폼에서는 미들웨어 API를 통해 계산 노드에 접근하여 해석 시뮬레이션을 수행할 수 있는 서비스 프레임워크를 제공한다.



(그림 1) EDISON 아키텍처(8)
(Figure 1) EDISON Architecture(8)

계산 과학 공학 응용 프로그램은 고성능 컴퓨팅 환경에서 시뮬레이션을 수행하며 웹 포털을 기반으로 계산 과학 공학 문제를 해결하는 역할을한다. 시뮬레이션 응용 프로그램을 통해 시뮬레이션을 실행하기 전에 사용자는 파일을 선택하거나 시뮬레이션 입력 포트에 대한 일부 값을 입력해야 하며 두 번째로 사용자의 입력 포트의 입력 및 전처리 과정이 완료되면 시뮬레이션을 실행할 수 있다. 마지막으로 시뮬레이션 결과를 분석하고 시각화 할 수 있는 다양한 후처리기를 통해 시뮬레이션 결과를 분석하고 유의미한 결과를 도출해 낼 수 있다.

3. 이벤트 기반 워크벤치

웹 포털을 통해 해석 시뮬레이션 소프트웨어를 실행하기 위한 실행 환경은 다양한 서비스를 기반으로 실행된다. 해석 시뮬레이션 소프트웨어 실행 환경은 사용자의 입력 또는 입력 값을 정의할 수 있는 전처리 프로세스, 사용자의 입력을 통해 해석 시뮬레이션을 실행하는 시뮬레이션 실행, 사용자의 입력을 통해 해석된 결과를 분석하는 후처리 프로세스를 포함한다. 또한 사용자에게 적절한 프로세스와 해당하는 서비스를 실시간으로 구성할 수 있도록 제공해야 한다.

워크벤치는 사용자의 실시간 요구에 따른 서비스 리소스를 적절하게 배치하고 제공해야 하며, 효과적인 시뮬레이션

실행 환경을 구성해야 한다. 또한 적절한 시뮬레이션 환경을 구성하고, 다양한 사용자의 요구에 상호 작용하는 실시간 서비스 구성 방법이 고안되어야 한다. 실시간으로 사용자에게 요구에 맞게 서비스를 처리하고 수행하기 위해서는 유저 인터페이스를 통한 이벤트 기반 서비스 구성 및 프로토콜을 설계 하고 각 모듈간의 커뮤니케이션이 가능하도록 해야 한다.

결과적으로, 이벤트를 기반으로 하는 워크벤치는 사용자에게 제공되는 다양한 서비스 모듈의 조합으로 구성된다. 각 모듈 사이에는 이벤트를 기반으로 하는 서비스 데이터 통신을 지원하며, 사용자의 요구에 맞게 실시간으로 동작할 수 있다.

3.1 이벤트 구조 및 프로토콜

계산과학 공학 계산 문제 해석을 위한 시뮬레이션이 실행되기 위해서는 각 해석 소프트웨어별로 필요한 모듈이 다르게 구성된다. 따라서 각 해석 소프트웨어별로 필요한 구성정보, 사용자 정보, 실행환경 정보 등이 다르기 때문에 해석 소프트웨어의 실행을 위해 필요한 각 모듈간의 커뮤니케이션이 필요하다. 이러한 모듈간의 커뮤니케이션 실행 환경에 서로 이벤트를 발생시키고 각 이벤트의 프로토콜에 따라 시뮬레이션 실행 환경 구축 및 시뮬레이션 전체 프로세스 과정을 컨트롤 한다. 따라서 모듈간에 이벤트를 주고 받는 형태에 대한 사전에 정의된 프로토콜이 필요하며, 이러한 프로토콜은 JSON 형태의 이벤트 데이터를 포함한다. 다음 그림 2는 이러한 이벤트의 종류를 캡처한 화면이다.

```

OSP.Event = {
  OSP_RESIZE: 'OSP_REPORT',
  OSP_FILE_SELECTED: 'OSP_FILE_SELECTED',
  OSP_FILE_SAVED_AS: 'OSP_FILE_SAVED_AS',
  OSP_FILE_DESELECTED: 'OSP_FILE_DESELECTED',
  OSP_LOAD_FILE: 'OSP_LOAD_FILE',
  OSP_LOAD_DATA: 'OSP_LOAD_DATA',
  OSP_REQUEST_DATA: 'OSP_REQUEST_DATA',
  OSP_RESPONSE_DATA: 'OSP_RESPONSE_DATA',
  OSP_DATA_CHANGED: 'OSP_DATA_CHANGED',
  OSP_UPLOAD_FILE: 'OSP_UPLOAD_FILE',
  OSP_DOWNLOAD_FILE: 'OSP_DOWNLOAD_FILE',
  OSP_REPORT_NAMESPACE: 'OSP_REPORT_NAMESPACE',
  OSP_DATA_LOADED: 'OSP_DATA_LOADED',
  OSP_SAMPLE_SELECTED: 'OSP_SAMPLE_SELECTED',
  OSP_UPLOAD_SELECTED: 'OSP_UPLOAD_SELECTED',
  OSP_OK_CLICKED: 'OSP_OK_CLICKED',
  OSP_CANCEL_CLICKED: 'OSP_CANCEL_CLICKED',
  OSP_ERROR: 'OSP_ERROR',
  Constants: {
    BROADCAST: 'BROADCAST',
    CONTENT: 'CONTENT',
    FILE_NAME: 'FILE_NAME',
    URL: 'URL'
  }
},
    
```

(그림 2) 워크벤치 이벤트 종류
(Figure 2) Workbench Event Type

워크벤치 실행을 위한 각 모듈에서는 필요한 이벤트를 발생시키고 발생시킨 이벤트에 대한 응답을 기다리거나 사용자 환경을 업데이트 한다. 이러한 이벤트 구조는 자바 스크립트 라이브러리로 공유되며 일정한 서로 미리 정의된 프로토콜에 따라 이벤트 모듈이 업데이트 된다.

```
var eventData = {
    portletId: portletId,
    targetPortlet: targetId,
    data: data
};

Liferay.fire( OSP.Event.OSP_DATA_CHANGED, eventData );
```

(그림 3) 워크벤치 이벤트 데이터
(Figure 3) Workbench Event Data

각 이벤트 종류별로 상이할 수 있지만 기본적으로 워크벤치에서 사용되는 이벤트 데이터의 형태는 그림 3과 같다. 각 이벤트별로 이벤트를 발생시키는 모듈은 발생시키는 모듈의 아이디와 이벤트를 수용하여 해석하는 모듈의 아이디를 포함하며, 새로 업데이트 되는 내용이나 사용자 요구에 대한 내용이 이벤트 데이터의 'data'에 포함된다.

```
OSP_REGISTER_EVENTS: [SimulationDashboard_WAR_edisonsimulationportlet,
OSP_REQUEST_APP_INFO: [SimulationDashboard_WAR_edisonsimulationportlet,
OSP_REGISTER_EVENTS: [SimulationBreadcrumb_WAR_OSPWorkbenchportlet, Thu
OSP_REQUEST_APP_INFO: [SimulationBreadcrumb_WAR_OSPWorkbenchportlet, Th
OSP_REGISTER_EVENTS: [ScienceAppPort_WAR_OSPWorkbenchportlet, Thu Apr 1
OSP_REQUEST_APP_INFO: [ScienceAppPort_WAR_OSPWorkbenchportlet, Thu Apr
OSP_REQUEST_PORT_INFO: [ScienceAppPort_WAR_OSPWorkbenchportlet, Thu Apr
OSP_RESPONSE_PORT_INFO: [SimulationWorkbench_WAR_OSPWorkbenchportlet, T
_TextEditor_WAR_OSPTextEditorportlet_
OSP_REGISTER_EVENTS: [TextEditor_WAR_OSPTextEditorportlet, Thu Apr 12
Text Editor OSP_EVENTS_REGISTERED: [SimulationWorkbench_WAR_OSPWorkben
_OSPJSMol_WAR_OSPJSMolportlet_
[JSMol] file explorer id Check : FileExplorer_WAR_OSPFileExplorerportlet
[JSMol]OSP_HANDSHAKE: [SimulationWorkbench_WAR_OSPWorkbenchportlet, Thu
OSP_REGISTER_EVENTS: [OSPJSMol_WAR_OSPJSMolportlet, Thu Apr 12 2018 09:
[JSMOL]RegisteredSimulationWorkbench_WAR_OSPWorkbenchportlet activated.
_OSPTextViewer_WAR_OSPTextViewerportlet_
OSP_HANDSHAKE: [SimulationWorkbench_WAR_OSPWorkbenchportlet] ▶ a.EventF
OSP_REGISTER_EVENTS: [OSPTextViewer_WAR_OSPTextViewerportlet, Thu Apr 1
OSP_EVENTS_REGISTERED: [SimulationWorkbench_WAR_OSPWorkbenchportlet] ▶
OSP_JOB_SELECTED: [SimulationDashboard_WAR_edisonsimulationportlet, Thu
Text Editor OSP_LOAD_DATA: [SimulationWorkbench_WAR_OSPWorkbenchportlet
▶ OSP.InputData
OSPTextViewer_WAR_OSPTextViewerportlet>>OSP_LOAD_DATA: [SimulationWork
▶ OSP.InputData
```

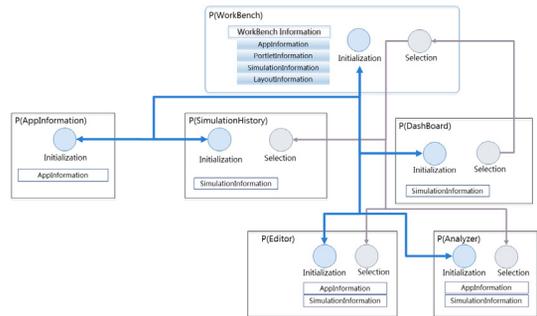
(그림 4) 워크벤치 실행 커뮤니케이션
(Figure 4) Execution Communication of Workbench

그림 4는 실제 워크벤치가 실행됨과 동시에 각 이벤트를 처리하는 모듈간의 커뮤니케이션의 내용을 캡처한 것이다. 지속적으로 각 워크벤치를 구성하는 모듈은 이벤트를 발생시키고 사용자에게 새로운 정보나 데이터를 업데이트 하여 공유한다.

3.2 실시간 서비스 구성 방법을 위한 워크벤치 초기화

실시간으로 사용자에게 필요한 해석 시뮬레이션을 수행하는 워크벤치 환경을 구성하기 위해서는 처음 사용자가 선택한 시뮬레이션 소프트웨어에 관련된 서비스들의 구성이 초기화 되어야 한다. 사용자가 해석 시뮬레이션을 수행하기 위해서 처음으로 사용자가 선택한 시뮬레이션 소프트웨어의 기본적인 정보, 시뮬레이션 소프트웨어를 수행하기 위한 전처리 서비스, 시뮬레이션 실행 서비스, 후처리 서비스가 필요하며 해당정보는 웹 포털 내부에 저장되어 있는 해석 시뮬레이션 소프트웨어 실행 정보에 포함된다. 또한 사용자의 시뮬레이션 실행 환경을 실시간으로 구성하기 위한 데이터 초기화 및 사용자의 기본 정보를 적절하게 분석하여 시뮬레이션 워크벤치 환경에 적용한다.

다음 그림 5는 실제 워크벤치가 실행되는 과정에서 사용자에게 제공하는 다양한 서비스를 초기화 하는 작업에 대하여 도식화하였다.



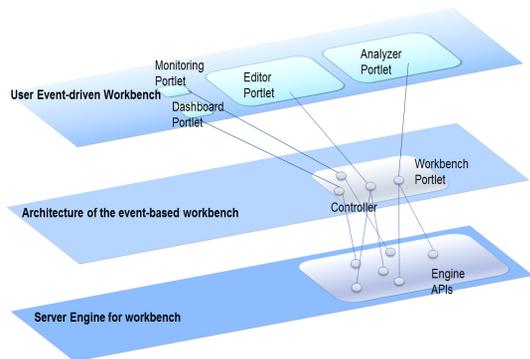
(그림 5) 웹 기반 시뮬레이션을 위한 워크벤치 초기화
(Figure 5) Web based Simulation Workbench Initialization

처음 워크벤치가 사용자에게 의해 호출되면 해석 시뮬레이션 소프트웨어 실행을 위한 기본 정보 및 해당 해석자가 필요로 하는 기본 서비스를 기반으로 하여 실시간으로 사용자에게 필요한 서비스를 구성한다. 워크벤치가 실행되면서 기본적인 정보를 포함하는 해석 시뮬레이션 소프트웨어의 정보를 표현하는 소프트웨어 정보 확인을 위한 포틀릿, 해당 소프트웨어를 사용했던 사용자의 시뮬레이션 히스토리를 호출하며, 시뮬레이션 히스토리 정보를 통해 시뮬레이션 및 시뮬레이션 잠을 구성하는 대시 보드 포틀릿 서비스를 호출한다. 이러한 과정은 워크벤치 기반으로 실행되는 환경에서 사용자의 호출과 동시에 웹 포털 내부적으로 진행되며, 사용자는 해당 서비스를 호출하고 화면이 구성되는 내부 프로

세스에 대한 이해 없이 워크벤치 시뮬레이션 시스템을 통해 시뮬레이션 기본 환경을 구성할 수 있다. 각 해석 소프트웨어의 시뮬레이션을 실행하기 위해서는 앞서 정의된 바와 같이 전처리기, 시뮬레이션 실행, 후처리기의 세 단계의 프로세스를 거치게 된다. 서비스를 지원하기 위해 워크벤치에서는 사용자가 시뮬레이션 수행을 위해 실행에 필요한 전처리기 에디터, 후처리기 분석기를 호출하며, 각 사용자의 선택에 따라 실시간으로 서비스를 호출하여 워크벤치 구성 환경에 반영된다. 시뮬레이션 실행을 위한 워크벤치 초기화는 첫째로 사용자의 현재 세션 정보를 통해 구성되며, 사용자의 시뮬레이션 히스토리, 사용자의 시뮬레이션 실행 정보를 반영하여 초기화 된다.

3.3 이벤트 기반 사용자 UI 구성

계산과학 공학 시뮬레이션 소프트웨어를 실행하기 위한 워크벤치는 사용자의 유저 인터페이스를 통해 웹 애플리케이션 기반으로 실행된다. 처음 시뮬레이션을 실행하게 되면 웹 내부적으로 사용자의 세션 정보 및 시뮬레이션 실행 히스토리 정보를 해석하며, 앞서 정의한 바와 같이 사용자의 환경 및 서비스 구성 정보를 초기화 한다. 이러한 해석 시뮬레이션 구성 환경을 초기화 하여 애플리케이션을 실행하는 과정은 다음 그림 6과 같이 도식화할 수 있다.



(그림 6) 웹 기반 워크벤치 계층 구조
(Figure 6) Web based Workbench Layer Architecture

각각의 계산과학 공학 해석 시뮬레이션을 수행하기 위한 서비스는 동적으로 구성되며, 각각의 서비스는 포틀릿 단위로 워크벤치 환경을 지원한다. 사용자의 정보를 표현하기 위한 사용자의 실행 이력 및 시뮬레이션 잡의 관리를 위한 대시보드 포틀릿, 현재 해석 시뮬레이션의 상태

및 실행 정보를 확인할 수 있는 모니터링 포틀릿, 시뮬레이션 입력 및 전처리 과정을 수행하기 위한 에디터 포틀릿, 시뮬레이션 실행 결과를 해석하는 해석자 포틀릿이 기본으로 구성되며, 사용자의 입력 및 시뮬레이션을 해석하는 해석 시뮬레이션 소프트웨어의 특징에 따라 실시간으로 사용자 시뮬레이션 환경을 구성하게 된다. 각각의 포틀릿 단위의 서비스를 수행하는 과정은 사용자의 이벤트를 기반으로 수행되며, 워크벤치에서는 서비스 포틀릿의 이벤트 입력을 받아 내부 서비스 리소스를 처리하는 적절한 엔진 API와 연결된다. 사용자의 유저 인터페이스를 통해 사용자의 실시간적인 요구사항 및 시뮬레이션 수행 및 해석에 관련한 각각의 이벤트 컨트롤을 포함하는 워크벤치는 필요에 따라 서버 리소스를 호출하고 디스플레이하는 역할을 수행한다.

다양한 사용자들이 시뮬레이션을 수행할 수 있으며, 해석자나 분석기, 결과 분석기 등을 개발하는 개발자는 미리 정의된 워크벤치 컨트롤에 의해 시스템 애플리케이션의 서비스 리소스를 호출할 수 있으며, 사용자의 환경에 따라 서비스 워크벤치를 실시간으로 구성할 수 있다.

4. 시뮬레이션 시스템 분석

현재 EDISON 서비스에서 제공하고 있는 계산과학 공학 해석 시뮬레이션의 서비스 종류는 표 4와 같다. 시뮬레이션의 입력 및 전처리 기능을 수행하는 전처리기는 총 4가지를 제공하고 있으며, 시뮬레이션을 수행하고 결과를 도출해내는 해석기는 607종이 있다. 또한 해석된 결과를 분석하고 사용자에게 시각화 서비스를 제공하는 후처리기는 8종이 존재한다. 단순히 각 해석 시뮬레이션 별로 제공할 수 있는 시스템 워크벤치 구성으로는 각 솔버 별로 전처리기 4종 및 후처리기 8종을 포함할 수 있으며, 각 워크벤치 구성은 사용자 선호도 및 시뮬레이션 환경에 따라 자유도가 높은 형태로 구성되며 사용자에게 제공된다.

(표 4) EDISON 서비스 소프트웨어
(Table 4) Service Software on EDISON

Type	No
Solver	607
Pre Processor	4
Post-Processor	8

4.1 시뮬레이션 시스템 비교 분석

기존의 EDISON 계산과학 공학 시뮬레이션 수행방식은 시뮬레이션 해석을 위한 솔버, 전처리 과정, 후처리 과정이 모두 분리된 서비스 형태로 제공되었다. 그림 7과 같이 시뮬레이션 잡 실행을 위해서는 새로운 시뮬레이션 잡을 생성하고, 입력값을 입력하고 잡을 제출하게 된다. 성공적으로 수행이 끝난 시뮬레이션 잡은 각 해석 시뮬레이션 결과를 확인하는 모니터링 서비스를 통해 후처리 과정을 수행한다.

반면 그림 8에서 보는 바와 같이 워크벤치 기반 계산과학 공학 시뮬레이션은 사용자 환경을 실시간으로 구성

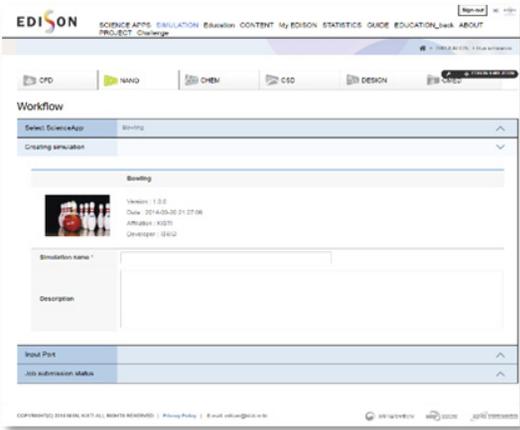
하며, 전처리 과정, 시뮬레이션 수행, 후처리 과정을 서비스의 이동 및 리소스 요청 재수행 없이 하나의 시뮬레이션 환경을 구성한 워크벤치 기반으로 수행할 수 있다.

이러한 두 가지 서비스의 가장 큰 차이점은 하나의 서비스로 모든 시뮬레이션 수행을 진행할 수 있는 것인지 아니면 여러 페이지 이동과 많은 서비스 요청을 진행하여 시뮬레이션을 수행할 것인지에 대한 것이다. 또한 워크벤치 기반 시뮬레이션 환경은 각 서비스 모듈 간의 이벤트 커뮤니케이션을 통해 사용자의 요구 및 시뮬레이션 결과를 동적으로 구성할 수 있다.

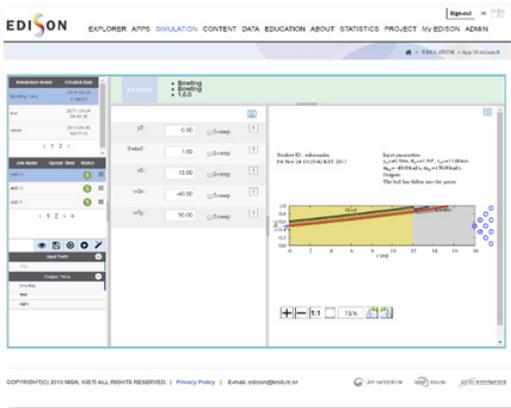
3.3 시뮬레이션 메모리 사용량 분석

기존 EDISON 계산과학 공학 시뮬레이션을 통한 해석 결과 과정은 여러 페이지 이동과 지속적인 서버 리소스 호출을 요구한다. 그러나 워크벤치 기반 시뮬레이션 수행은 하나의 통합된 웹 서비스 호출로 전체적인 프로세스를 단축할 수 있었다. 또한 워크벤치 내부적으로 서비스를 세분화 하여 이벤트 커뮤니케이션을 수행함으로써 사용자의 실시간적인 요구를 반영할 수 있는 시스템을 구축할 수 있었다.

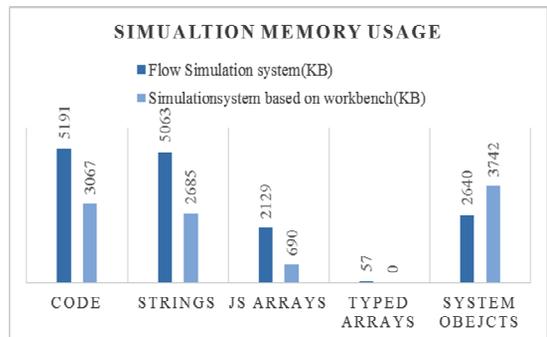
이러한 결과로 실제 같은 해석 시뮬레이션을 수행하였을 때, 두 시스템의 결과 및 메모리 사용량을 분석하여 비교하였다. 그림 9는 사용자가 두 가지 시스템을 사용한 경우 총 메모리 사용량에 대한 사용자 환경에서의 메모리 사용량을 나타낸 것이다. 사용자가 시뮬레이션을 실행하고 시뮬레이션 전처리 과정, 시뮬레이션 실행, 시뮬레이션 후처리 과정을 거치면서 사용되는 메모리 사용의 총량을 표현하며, KB 단위로 표현된다. 사용자에게 필요한 시스템의 메모리 사용량은 전반적으로 워크벤치를 기



(그림 7) 기존 시뮬레이션 시스템
(Figure 7) Existing Simulation System



(그림 8) 워크벤치 기반 시뮬레이션 시스템
(Figure 8) Workbench based Simulation System



(그림 9) 메모리 사용량 비교
(Figure 9) Memory Usage Comparison

반을 하는 계산과학 공학 시뮬레이션 환경에서 적은 추세를 보이며, 시뮬레이션을 수행하는 사용자의 로컬 환경에서 시뮬레이션을 수행하는데 있어 전체적인 코드, 자바스크립트 배열, 문자열 등의 메모리 사용량은 워크벤치를 기반으로 하는 시뮬레이션 시스템에서 약 60% 정도로 감소되는 추세를 보인다.

시스템 오브젝트는 워크벤치 기반 시뮬레이션 시스템에서 3742KB로 더 큰 메모리 사용량을 보이는데 이는 워크벤치 기반 시뮬레이션 시스템에서 사용자에게 시뮬레이션 환경을 제공할 때, 각 포틀릿 단위로 서로 다양한 이벤트를 주고 받으며 사용자의 입력, 컨트롤, 결과 값의 빈번한 데이터 교환으로 인한 것으로 분석된다. 동일한 계산과학 공학 문제를 해결하는 해석 소프트웨어를 수행하였을 때, 절차 지향 시뮬레이션 시스템은 15080KB의 메모리 사용량을 가지며, 워크벤치 기반 소프트웨어 시뮬레이션을 수행한 경우에는 10184KB의 메모리를 사용하였다. 이는 워크벤치 시뮬레이션에서 사용되는 전체 시뮬레이션 메모리 사용량이 기존 EDISON 시뮬레이션 시스템에 비해 32.46% 감소한 것을 의미한다. 해당 결과는 시뮬레이션을 수행하는데 있어 시뮬레이션을 생성하고, 입력 및 전처리 과정을 수행하고, 시뮬레이션을 실행 한 후, 해당 결과를 분석하기 위한 두 개의 후처리 과정을 포함한다. 기존 EDISON 시스템에서는 여러 페이지 이동과 서비스 호출이 필요하지만, 워크벤치 기반 계산과학 공학 시뮬레이션 시스템에서는 하나의 통합된 서비스를 제공하기 때문에 메모리 효율 및 리소스 사용량의 감소를 도출해 낼 수 있었다.

5. 결론 및 향후 연구

지금까지 본 논문에서는 계산과학공학 분야를 위한 시뮬레이션 실행 플랫폼인 EDISON 시스템에서 사용자의 실시간 서비스 요청과 서비스 구성 환경을 다양하게 구성하여 사용자의 실시간 요구사항에 맞게 서비스를 구성하고 제공하는 워크벤치 기반 시뮬레이션 시스템에 대하여 연구하였다. 또한 실제 서비스 되고 있는 해석 시뮬레이션 환경에서 사용자에게 최적화된 서비스 구성 요소를 조합하여 실시간으로 제공하는 통합 워크벤치 시뮬레이션을 제공함으로써 기존의 시뮬레이션 시뮬레이션 시스템에서 필요로 하는 시스템 프로세스를 단축시킬 수 있었다.

본 연구에서는 현재 사용자에게 제공되고 있는 계산과학 공학 시뮬레이션 및 해석 결과 분석 플랫폼인 EDISON

시스템에 통합적인 유저 인터페이스를 제공하기 위한 워크벤치 기반 이벤트 모듈 커뮤니케이션 프로토콜의 설계 및 구축하였다. 또한 사용자가 로컬에서 사용하는 계산과학 공학 시뮬레이션 소프트웨어 환경과 유사한 사용자 유저 인터페이스를 가지며 다양한 해석기와 전처리기, 해석 시뮬레이션 소프트웨어 등의 동적인 웹 환경을 구성할 수 있는 워크벤치 기반 사용자 시뮬레이션 서비스 환경 구성에 대해 고안하였다. 이러한 웹 서비스를 제공하기 위해 사용자에게 필요한 서비스를 포틀릿 단위로 구성하였다.

기존의 시스템에서 제공되는 다양한 계산과학 공학 해석 시뮬레이션은 사용자에게 시뮬레이션을 생성하고 결과를 분석하는데 있어 여러 과정을 거쳐야 했고 사용자에게 불필요한 서비스를 포함하는 서비스의 낭비도 존재했다. 또한 이미 정해져 있는 계산과학 애플리케이션 시뮬레이션의 실행 절차를 따라가며, 실시간으로 사용자의 니즈와 서비스 상태 등을 파악하는 과정이 포함되어 있지 않았다. 본 논문에서는 실시간으로 사용자의 요구를 분석하고 필요한 서비스를 호출하여 실시간으로 서비스를 조합해 사용자에게 제공하는 워크벤치 기반 해석 시뮬레이션 시스템을 구축하였고, 사용자에게 친숙한 워크벤치 기반 사용자 유저 인터페이스를 제공하였다. 또한 필요한 서비스 자원을 모듈화하여 서비스 자원의 효율성을 증가시켰다. 동시에 여러 사용자가 통합 포털에 접근하여 시뮬레이션을 수행한다 하더라도 각 사용자별로 사용하고자 하는 자원이 다양화되었기 때문에 서버의 과부하를 방지할 수 있는 방편으로 사용될 수 있다.

향후 연구로는 사용자에게 맞춤형 선택적인 서비스 조합 및 모듈 구성에 대한 분석을 다각적으로 진행할 예정이다. 사용자 행동 분석 시나리오를 통해 사용자 요구를 반영하는 시뮬레이션 시스템을 제공하며 동적인 웹 서비스를 제공하는 것에 목적을 두고 있다. 또한 각 사용자에게 제한된 서비스 접근 제한 및 조건, 권한 등을 조직화된 사용자 그룹에 따라 세분화하여 특정 사용자에게 최적화된 서비스를 제공할 예정이다.

참고문헌(Reference)

- [1] MELL, Peter, et al. "The NIST definition of cloud computing". 2011.
- [2] Ostermann, Simon, et al., "A performance analysis of EC2 cloud computing services for scientific computing.", International Conference on Cloud Computing. Springer,

- Berlin, Heidelberg, pp.115-131, 2009.
- [3] Byrne, David, Carol Corrado, and Daniel Sichel., "The Rise of Cloud Computing: Minding your P's and Q's.", Conference on Research in Income and Wealth (CRIW)/National Bureau of Economic Research (NBER) Conference "Measuring and Accounting for Innovation in the 21st Century," March, 2017.
- [4] Weintraub, Eli, and Yuval Cohen., "Multi Objective Optimization of Cloud Computing Services for Consumers.", INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS, Vol 8.2, pp. 139-147, 2017.
https://thesai.org/Downloads/Volume8No2/Paper_19-Multi_Objective_Optimization_of_Cloud_Computing_Services.pdf
- [5] Klimeck, Gerhard, et al., "nanoHUB. org-online simulation and more materials for semiconductors and nanoelectronics in education and research.", Nanotechnology, NANO'08. 8th IEEE Conference on. IEEE, pp. 401-404, 2008.
- [6] Yasar, Osman, and Rubin H. Landau. "Elements of computational science and engineering education." SIAM re-view, Vol. 45.4, pp. 787-805, 2003.
- [7] Du-Seok Jin, Youngjin Jung, Hoe-Kyung Jung, "EDISON Platform to Supporting Education and Integration Research in Computational Science", Journal of the Korea Institute of Information and Communication Engineering Vol. 16, no. 1, pp.176-182, 2012.
<https://doi.org/10.6109/jkiice.2012.16.1.176>
- [8] Ma, J., Lee, J. R., Cho, K., and Park, M. 2017., "Design and Implementation of Information Management Tools for the EDISON Open Platform", KSII Transactions on Internet and Information Systems, 11, 2, 2017, pp. 1089-1104. <https://doi.org/10.3837/tiis.2017.02.026>
- [9] Jin Ma, Jong Suk Ruth-Lee, "A Design of Storage Service on EDISON Platform", in Proc. Of KSII Fall Conference, pp. 38-47.
- [10] Y.-K. Suh, et. al., "EDISON: A Web-Based HPC Simulation Execution Framework for Large-Scale Scientific Computing Software", in Proceedings of the ACM/IEEE CCGrid'16, pp. 608-612, 2016.
- [11] <https://www.simscale.com/workbench?pid=6952061813321666514>
- [12] <https://nanohub.org/tools/CNDO/invoke/38>

● 저 자 소개 ●



권 예 진 (Yejin Kwon)

2007년 단국대학교 컴퓨터학과 졸업(학사)
2014년 단국대학교 대학원 전자계산학과 졸업(석사)
2016년~현재 한국과학기술정보연구원(KISTI) 계산과학플랫폼센터 연구원
관심분야 : 소프트웨어 공학, 소프트웨어 테스트, 신뢰성 분석
E-mail : yejinkwon@kisti.re.kr



전 인 호 (Inho Jeon)

2009년 광운대학교 전자공학과 졸업(학사)
2017년 광운대학교 전자공학과 졸업(박사)
2013년~현재 한국과학기술정보연구원(KISTI) 계산과학플랫폼센터 선임연구원
관심분야 : 스마트러닝, 오픈 사이언스, 웹포털 프레임워크
이메일 : inojeon@kisti.re.kr



마 진 (Jin Ma)

2010년 광운대학교 컴퓨터소프트웨어학과 졸업(학사)
2012년 광운대학교 대학원 컴퓨터학과 졸업(석사)
2012년~2015년 (주)비스텔 선임연구원
2015년~현재 한국과학기술정보연구원(KISTI) 계산과학플랫폼센터 연구원
관심분야 : 데이터 통합, 빅 데이터, 분석시스템, 분산컴퓨팅, 정보검색
E-mail : majin@kisti.re.kr



이 식 (Sik Lee)

1989년 서울대학교 화학과 졸업(학사)
1993년 포항공과대학교 화학과 졸업(석사)
1996년 포항공과대학교 화학과 졸업(박사)
2000년~현재 한국과학기술정보연구원 (KISTI) 슈퍼컴퓨팅클라우드센터 책임연구원 (센터장)
관심분야 : 계산과학, 바이오 인포매틱스, 오픈 사이언스, 이공계 교육·연구 융합
E-mail : siklee@kisti.re.kr



조 금 원 (Kum Won Cho)

1993년 인하대학교 항공우주공학과 졸업(학사)
1995년 KAIST 항공우주공학과 졸업(석사)
2000년 KAIST 항공우주공학과 졸업(박사)
2000년~2017년 한국과학기술정보연구원 (KISTI) 계산과학공학센터 책임연구원
2018년~현재 한국과학기술정보연구원 (KISTI) 부원장
관심분야 : 계산과학, 항공우주, 유체해석, 이공계 교육·연구 융합
E-mail : ckw@kisti.re.kr



서 정 현 (Jung-Hyun Jerry Seo)

1987년 한양대학교 수학과 졸업(학사)
2016년 연세대학교 대학원 졸업(박사)
1987년~현재 한국과학기술정보연구원(KISTI) 계산과학플랫폼센터 선임연구원
관심분야 : 웹서비스, 웹포털 프레임워크, 정보검색
E-mail : jerry@kisti.re.kr