

계산 과학 응용을 위한 과학 워크플로우 통합 수행 환경 설계[☆]

A Design of Integrated Scientific Workflow Execution Environment for A Computational Scientific Application

김 서 영* 윤 경 아** 김 윤 회***
Seoyoung Kim Kyoung-a Yoon Yoonhee Kim

요 약

빠르게 발전하는 컴퓨팅 기술에 비례해서 컴퓨팅 기기의 이용이 더욱 편리해짐에 따라 계산과학 분야의 연구자들은 점점 더 컴퓨팅 기술에 의존하고 있으며 더욱 계산 집약적인 연구를 진행하고 있다. 이러한 계산 과학 연구자들의 연구 환경 지원을 위하여 전 세계적으로 e-사이언스 환경에 대한 연구가 활발하게 진행되고 있으며, 여전히 연구자의 실험 환경과 컴퓨팅 자원 간의 별도의 설정과 설치 과정이 필요 없는 자유로운 연구 환경에 대한 보장이 요구되고 있다.

본 논문에서는 계산과학 분야의 연구자들의 실험 환경을 분산된 고성능 컴퓨팅 인프라와 함께 웹 브라우저를 통해 쉽게 접근 가능한 과학 워크플로우 통합 수행 환경을 제시한다. 이 환경에서는 워크플로우 설계 툴을 통해 정의된 순서와 작업 특성을 고려하여 분산된 그리드 자원에 제출함으로써 작업들에 대한 자동화된 수행을 제공한다. 포털을 통해 제공되는 워크플로우 설계 툴은 사용하기 쉬운 인터페이스를 제공하며 실시간으로 워크플로우의 실행 상태를 모니터링 서비스를 통해 제공한다. 사용자는 워크플로우의 실행에서의 중간 결과를 확인하고 수행의 흐름을 결정할 수 있다. 이로써 복잡한 HTC 기반의 실험 수행을 효과적으로 진행할 수 있으며 연구의 생산성을 높일 수 있다.

ABSTRACT

Numerous scientists who are engaged in compute-intensive researches require more computing facilities than before, while the computing resource and techniques are increasingly becoming more advanced. For this reason, many works for e-Science environment have been actively invested and established around the world, but still the scientists look for an intuitive experimental environment, which is guaranteed the improved environmental facilities without additional configurations or installations.

In this paper, we present an integrated scientific workflow execution environment for Scientific applications supporting workflow design with high performance computing infrastructure and accessibility for web browser. This portal supports automated consecutive execution of computation jobs in order of the form defined by workflow design tool and execution service concerning characteristics of each job to batch over distributed grid resources. Workflow editor of the portal presents a high-level frontend and easy-to-use interface with monitoring service, which shows the status of workflow execution in real time so that user can check the intermediate data during experiments. Therefore, the scientists can take advantages of the environment to improve the productivity of study based on HTC.

☞ keyword : Scientific Workflow(과학 워크플로우), e-Science(e-사이언스), Computational Science application(계산 과학응용), Problem Solving Environment(문제풀이 환경)

1. 서 론

계산과학 분야는 국가의 과학 경쟁력을 높이고 산업

에의 적용을 통하여 경제적인 경쟁력 강화에 중요한 역할을 하므로 전 세계적으로 매우 중요한 연구 분야로서 잘 알려져 있다. 하지만 대규모 고성능 계산 자원의 확충 및 복잡한 컴퓨팅 설정에서 발생하는 어려움, 또는 공유하여 사용하는 상용 소프트웨어로의 접근 제한성으로 타 연구 분야에 비해 상대적으로 미흡한 발전을 보이고 있다. 이러한 문제점을 극복하고자 과학 응용을 위한 통합 실험 환경인 e-사이언스[11,13] 연구가 최근 몇 년간 국내외로 활발하게 진행되어왔다. 국내 대표적인 연구로는 계산과학 분야 중 항공우주 연구를 위한 실험 환경인 e-AIRS[1]가 존재한다. e-AIRS는 교육용 사이버 연구 환

* 준 회 원: 숙명여자대학교 대학원 컴퓨터과학부
컴퓨터과학과 석사과정 sssyy77@sookmyung.ac.kr
** 준 회 원 : 숙명여자대학교 대학원 컴퓨터과학부
컴퓨터과학과 석사과정 yoonka@sookmyung.ac.kr
*** 종신회원 : 숙명여자대학교 컴퓨터과학부 교수
yulan@sookmyung.ac.kr(교신저자)

[2011/08/30 투고 - 2011/09/08 심사 -2011/12/20 심사완료]

☆ 본 연구는 숙명여자대학교 2010학년도 교내연구비 지원에 의해 수행되었음

경으로, 연구자 그룹 간의 효과적인 정보 및 협의 서비스를 제공해왔다. 그러나 주로 항공 우주 분야의 기초 교육 활용을 위해 설계되었기 때문에 사용자가 초보 연구자에 한정되어 있다. 상대적으로 수행이 복잡하고, 다수의 값 비싼 소프트웨어를 사용하는 심화 연구자들은 오히려 직접 환경을 구축하여 실험을 해야 하는 상황이며 이들을 위하여 사용이 편리한 통합형 실험 환경의 지원이 필요하다. 본 논문에서는 계산과학 분야 연구 중 하나인 항공 우주 연구를 프로토타입으로 하여 과학 워크플로우 중심 통합 수행 환경을 제안한다. 과학 워크플로우 중심 시스템[12]은 과학 연구자 들을 위해 복잡한 과학 응용의 실행을 사용자가 정의한 흐름대로 자동화하여 효율적인 실행이 가능하도록 도우며, 광범위하게 분산된 그리드 자원과의 접근을 통해 고성능 컴퓨팅 자원을 사용함으로써 단시간 내에 실험 결과를 제공받을 수 있다.

본 논문에서는 2장에서 관련 연구를 논하고, 3장에서는 전체 워크플로우 서비스를 제공하는 포털의 기본 구조에 대해 알아본다. 4장에서는 과학 워크플로우의 수행 처리를 위한 프레임워크를 설명하고, 5장에서 워크플로우 설계 환경의 구현에 대해 보이며 끝으로 결론 및 향후 과제를 기술한다.

2. 관련 연구

계산과학 응용이 점점 더 복잡해짐에 따라, 이에 포함된 복잡한 과정들을 단순화하고 자동화하기 위하여 과학 워크플로우 대한 연구들이 여러 계산과학 분야에 걸쳐서 진행되어 왔다[2,3,4]. 본 장에서는 과학 워크플로우에 대한 연구 및 이를 포함한 통합 실험 환경 연구들을 소개한다. 본 연구에서 대상으로 하는 항공우주 연구 분야 외에도 생물학, 천문학, 물리학 등을 위한 워크플로우의 설계 연구로 Pegasus-WMS[2]와 통합 수행 환경을 함께 제공한 연구인 Taverna[3], Triana[4]가 존재한다.

Pegasus-WMS(Pegasus Workflow Management Service) [2]는 천문학, 생물학 등 다양한 과학 분야에 적용되어 사용되고 있는 워크플로우 관리 시스템으로, 분산되어 존재하는 대규모의 컴퓨팅 자원 상에서 복잡한 워크플로우의 실행을 관리 한다. 워크플로우가 구성되면 Pegasus-WMS는 워크플로우를 이용 가능한 컴퓨팅 자원에서 동작 할 수 있도록 연결 과정을 거쳐 사용자에게 의해 정의된 적절한 순서로 워크플로우에 포함된 작업들이 실행 되도록 한다. 또한 포함된 컴포넌트는 사용자에게 의해 요구된 워크플로우를 실행 가능한 형태로 구성하여 적절한 자원

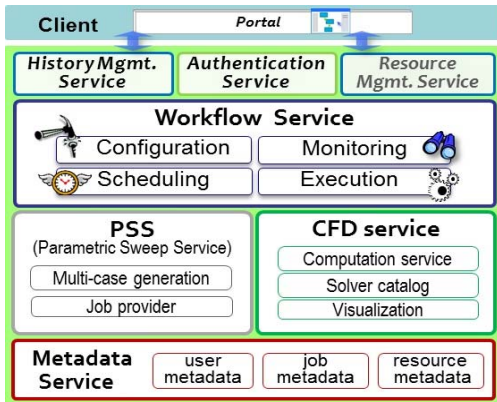
을 찾아 연결 시켜주는 기능도 제공한다. 이러한 Pegasus-WMS는 워크플로우 수행과 관리에 효율적인 관리 서비스를 제공하지만, 실제로 다양한 과학 분야의 연구자들이 사용하기 위해서는 현재 연구자들의 컴퓨팅 환경에 별도의 설치가 요구되며, 연구자들이 사용하는 클러스터 환경에서 Pegasus-WMS와의 통신이 가능하도록 특정 작업 스케줄러인 Condor를 이용하여야 한다. 따라서 과학 분야의 연구자들이 이를 잘 활용할 수 있는 추가적인 인터페이스의 제공이 요구된다.

Taverna[3]는 생물학 중심의 과학 워크플로우 관리 툴이다. 이는 웹 서비스 기반의 수행을 지원하여 이와 함께 제공하는 워크벤치(Workbench)를 통해 직접 웹 서비스에 연결하거나, 웹 서비스가 연결된 사이트를 통해 워크플로우 사용을 가능하게 해준다. 또한 사용자는 가시적으로 표현된 패널 인터페이스를 이용하여 각각의 프로세서와 링크들을 구성함으로써 워크플로우를 설계할 수 있고, 그 수행 결과를 텍스트 파일 형태로 사용자에게 제공하거나 또는 가시화 처리 소프트웨어에 직접 연결하여 제공한다.

Triana [4]는 천문학과 물리학 등을 대상으로 과학 워크플로우 구성을 지원하는 오픈 소스 기반 문제풀이 환경 소프트웨어이다. 이는 그리드 서비스 연결을 가능하게 하고, 과학 연구자들이 편리한 사용을 보장하는 가시적 인터페이스를 기반으로 조립-설치 가능한 컴포넌트들을 제공한다. 하지만 앞서 살펴본 두 연구는 사용시 추가적인 설정이 요구되어 비 전문가들의 사용이 어렵다는 단점이 존재한다.

3. 기본 포털 구조

본 논문에서는 [5]를 선행 연구로 하여 확장된 형태의 응용을 적용하는 경우에도 사용자가 쉽게 다룰 수 있는 형태를 갖는, 인터페이스 위주의 통합 환경으로 확장하고 개선하였다. [5]에서 제시한 포털 시스템의 실험 서비스는 연구자에게 주로 고정된 실험 단계를 제공하고, 소수의 기능만을 갖는 워크플로우 편집 서비스를 제공하였다. 하지만 더욱 확장된 응용을 적용하는 경우, 전반적인 실험 흐름의 변경이 잦아 워크플로우 편집 서비스를 더욱 선호하게 되는데, 이 워크플로우 편집 서비스에서 제공하는 모듈에서의 속성 값 설정이 극히 제한적이라는 한계점이 존재하였다. 이를 개선하여 구체적인 실험 설정이 가능하며 사용자의 동적인 요구를 반영하고, 그 수행을 지원할 수 있는 워크플로우 설계 서비스를 구성하



(그림 1) 포탈 시스템의 서비스 구성

었다.

전체 포탈 시스템은 크게 워크플로우 서비스, 어플리케이션 서비스, 그리고 메타데이터 서비스와 기타 서비스로 구성되며, (그림 1)은 포탈 시스템의 전체 서비스 구성을 나타낸다.

3.1 워크플로우 서비스

워크플로우 서비스는 사용자에게 의해 설계된 작업 흐름을 내부적으로 처리할 수 있는 형태로 생성해 주고, 그 흐름에 따라 수행된 결과를 일괄적으로 사용자가 얻을 수 있도록 돕는다. 사용자에게 의해 설계된 워크플로우의 작업 순서와 속성 값 같은 메타

정보들은 특정 시점에 포탈 인터페이스로부터 내부 처리 시스템으로 전달 된다. 전달된 데이터를 기반으로, ‘워크플로우 서비스’에서는 시스템에서 해석과 관리가 가능한 형태의 스크립트로 구성(Configuration)해주며, 구성된 워크플로우 명세 스크립트에 사용자의 결정을 포함하여 스케줄링 (Scheduling)이 가능한 형태로 구성해준다. 스케줄링 서비스에서는 사용자에게 의해 정의된 순서와 워크 플로우를 구성하는 모듈의 타입, 그리고 사용자의 의사 결정을 반영하여 스케줄링을 적용하며, 최종 수행 후에는 워크플로우에 속하는 다수의 작업들의 실행 결과를 다시 모아서 하나의 파일로 구성 (Orchestration) 및 통합해준다. 실행 (Execution) 서비스에서는 최종적인 작업 실행 준비를 한 후 미들웨어와의 인터페이스를 통해 작업들을 실행해준다. 실행되고 있는 작업들은 워크플로우 모니터링(Monitoring) 서비스에 의해 각 작업의 실행 상태 정보와 워크플로우의 진행 사항을 사용자에게 제공하

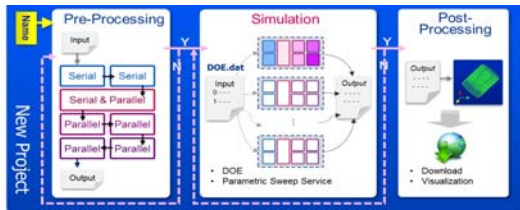
며 워크플로우를 구성하는 작업들의 실행 정보까지 일괄적으로 관리한다.

3.2 어플리케이션 서비스

어플리케이션 서비스는 수치 해석 및 실험 서비스를 제공하기 위해 지원되는 서비스로 파라 메트릭 스위프 서비스(PSS, Parametric Sweep Service)와 CFD 서비스가 여기에 속한다. 먼저 파라메트릭 스위프 서비스(PSS)는 HTC (High Throughput Computing) 응용의 수행을 위해 제공되는 서비스로, 다양한 파라미터 값을 동일한 실행 파일에 적용하여 파라미터 값에 따른 실험 결과를 비교하고 분석할 수 있도록 하며, 사용자에게 적용할 파라미터 값의 범위를 입력 받아 자동으로 문제(Case)들을 생성하고 수행해준다. CFD 서비스는 수치 해석을 위한 계산 서비스 (Computation Service)와 분석을 위한 그래프 생성 및 가시화 처리(Visualization) 서비스, 해석자 목록(Solver Catalog) 서비스를 포함한다. 먼저 계산 서비스는 모든 수치 해석 수행을 담당하며, 해석자 목록 서비스는 사용자가 직접 설계한 해석자(Solver)를 실험에 사용할 수 있도록 다양한 해석자의 목록을 일괄적으로 관리해 준다. 실험 도중의 중간 값을 실시간으로 반영한 그래프 생성 및 실험 후의 결과에 대한 가시화 처리를 가시화 서비스로 제공한다.

3.3 메타데이터 서비스 및 기타 서비스

그 밖의 제공되는 주요 서비스로는 메타데이터 서비스가 존재한다. 메타데이터(Metadata) 서비스는 포탈 서버에서 다루는 간결한 데이터 정보와 데이터 스토리지의 실제 데이터 정보 사이에 데이터 인터페이스를 위하여 수행 중에 사용되는 다양한 정보를 관리하는 서비스이다. 사용자 관리를 위한 사용자 메타 데이터(User Metadata)가 존재하며, 또한 계산 자원에 제출된 작업 런타임 정보 및 설계 정보 등의 작업 이력을 관리하는 작업 메타 데이터가 있다. 자원 메타데이터는 작업 제출 대상이 되는 컴퓨팅 자원의 정보를 의미하며 컴퓨팅 자원의 호스트 이름, 해당 자원을 소유하고 있는 기관 명, 전체 CPU 개수와 가용 CPU 수, 메모리, 디스크 용량 및 컴퓨팅 노드 정보와 OS 종류, 로컬 스케줄러 종류 등과 같은 정보를 포함한다. 이러한 메타 데이터를 바탕으로 사용자는 작업 제출을 위한 컴퓨팅 자원을 선택이 가능하며, 작업 이력 관리를 통해 실험의 비교 및 분석이 가능하다. 이 외에도 포탈 시스템을 통하여 직접적으로 사용자에게 제공되는



(그림 2) 과학 시뮬레이션 실험의 수행 흐름

서비스들이 다음과 같이 존재한다. 시뮬레이션 이력 관리 서비스(History Management Service)는 사용자의 과거 실험 이력 데이터를 제공하여 이를 기반으로 새로운 실험에 재사용하고 분석할 수 있도록 제공하는 서비스이다. 다양한 컴퓨팅 자원들을 사용 가능하도록 가용 자원에 대한 목록을 사용자에게 제공하며, 관리를 제공하는 자원 관리 서비스 (Resource Management Service)가 메타 데이터 서비스를 기반으로 함께 제공된다.

4. 과학 워크플로우 수행 프레임워크

4.1 실험 수행 시나리오

본 연구에서는 주로 유체 흐름과 연관된 문제에 대해 컴퓨터를 이용하여 대량의 연산을 수행하는 전산유체역학(CFD)[10]을 대상으로 한다. CFD는 유체의 동적인 움직임을 해석하는데 수치해석적인 방법으로 계산을 수행하는 연구 분야이다. 포털 서비스를 통해 제공하는 CFD 실험의 일반적 수행 과정은 크게 전처리(Pre-processing), 시뮬레이션(Simulation), 후처리(Post-processing)의 세 단계로 구성되며 그 수행 흐름은 (그림 2)와 같다. 각 모듈은 다수의 반복적인 계산 작업을 포함하며 이들 간의 순차적 수행이 보장되어야 한다.

전처리 모듈(Pre-processing module)은 CFD 계산 수행을 위한 해석 대상의 모델링 및 입력 파일의 최적화 수행을 제공한다. 이를 위해서는 사용자의 입력 파일을 입력 값으로 하여 직렬(Serial) 및 병렬(Parallel) 프로그램들로 구성된 일곱 단계의 순차적인 계산 수행이 수동 및 자동으로 제공되어야 한다.

전처리 모듈의 수행 후에는 사용자의 결정에 의해 다음 단계인 시뮬레이션 모듈(Simulation module)을 수행하게 된다. 시뮬레이션 모듈은 HTC(High Throughput Computing) 수행을 제공하는 단계이다. 사용자는 이 모듈에서 다양한 파라미터 값에 대해 동일한 계산이 수행할 수 있는데,

이는 다양한 파라미터 값 또는 입력에 대해 실험 및 시뮬레이션에 미치는 영향을 알아보기 위해 가설 설정에 따라 실험 계획(DOE: Design of Experiment)을 가능하게 하기 위함이다. 실행될 계산 응용은 전처리 모듈과 동일하게, 직렬 또는 병렬 프로그램 형태의 일곱 가지 응용의 순차적 실행이 적용되며 사용자로부터 입력 받은 파라미터의 범위 값 내의 다양한 값들을 이에 적용하여 다수의 문제들을 생성하고 대용량 계산을 동시에 수행하게 된다.

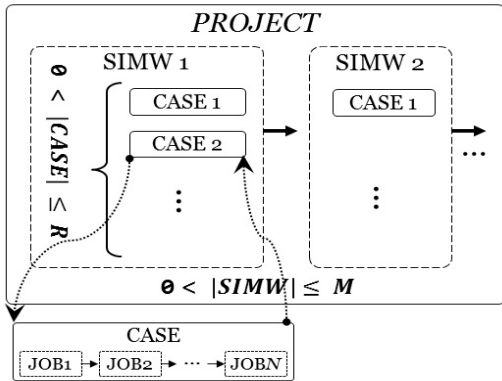
마지막으로, 시뮬레이션 모듈에서 HTC 수행이 완료되면 사용자는 그 결과를 다운로드 받거나 분석을 위해 가시화 툴을 통한 가시화 서비스를 제공받을 수 있다. 이 과정을 후처리 모듈(Post-processing module)이라고 하고, 이 모듈의 수행이 끝나면 사용자는 하나의 실험을 완성하게 된다.

4.2 워크플로우 수행 관리

앞서 살펴본 실험의 수행 흐름을 바탕으로 포털 상에서 워크플로우 설계 서비스가 제공되며, 사용자는 자신만의 실험 과정을 포털의 편집 툴을 통해 재정의 할 수 있다. 사용자가 설계한 실험은 그 순서와 입력 값과 함께 다수의 응용들의 실행으로 이어지는데, 실험 설계 상의 복잡성과 반복 정도에 제한 없이 동일한 실험 서비스를 지원할 수 있어야 한다. 따라서 복잡한 수행 과정으로 구성된 수행 체계를 계층적으로 분리하여 관리되도록 하였으며, 관리 단위의 정의로 사용자에게 유연한 설계 서비스를 제공할 수 있다.

관리 단위는 각각 PROJECT, SIMW(Workflow for Simulation), CASE, JOB으로 정의하였다. 이러한 단위들로 수행을 관리한다는 것은, 정의된 단위 별로 식별 번호(id)를 부여하여 수행 체계에서 사용하는 것을 의미하며, 작업 스케줄링 시 실행 순서 관리에도 활용된다. 가장 포괄적인 관리 단위로는 PROJECT가 사용되고, 계산이 처리되는 가장 작은 관리 단위로 JOB이 사용된다. (그림 3)은 정의된 단위들간의 관계를 나타내며 각 단위들이 나타내는 바는 다음과 같다.

먼저 PROJECT는 전체 실험 과정을 포함하는 가장 상위 단계의 단위로, 사용자가 포털 서비스를 통해 워크플로우를 설계할 때 하나의 PROJECT가 생성되며 그에 따른 식별 번호가 부여된다. SIMW는 전처리, 시뮬레이션, 후처리 모듈에 대응되는 단위로, 하나 이상의 SIMW가 PROJECT에 포함될 수 있다. (그림 3)에서 보이는 것처럼 SIMW의 생성은 사용자가 워크플로우 편집기를 통하여



(그림 3) 수행 관리 단위 관계도

각 세 가지 모듈의 사용에 따라 결정되며, 사용자가 화면 상에서 사용한 모듈 개수인 M 개에 제한된다. 예를 들어, 사용자가 전처리 모듈을 두 번 반복하려고 한다면 내부적으로는 $SIMW$ 에 해당하는 2개의 식별 번호가 생성되며, 사용자가 전처리 과정을 한 번 수행 후에 시뮬레이션 과정이 수행되도록 설계하여 제출한 경우도 역시 2개의 $SIMW$ 식별 번호가 부여된다. 단일 $PROJECT$ 식별 번호에 속하는 $SIMW$ 들은 동시 수행이 불가능하며, 각 $SIMW$ 사이에 정의된 의존성을 통해 반드시 순차적으로 수행되도록 관리한다. 하나의 $SIMW$ 내에는 다수의 $CASE$ 들이 포함될 수가 있는데 $CASE$ 의 개수(R)는 HTC 에서 동일 실행 파일에 다른 파라미터 값이 적용되는 경우이므로, 사용자가 입력할 파라미터 값의 범위에 따라 다양해질 수 있다.

$CASE$ 는 가장 하위 단위인 JOB 의 집합, 즉 여러 응용들의 순차적인 수행이 포함되며, 여러 JOB 들이 정의된 순서대로 수행된다. $CASE$ 는 HTC 과정에서 PSS 서비스에 의해 생성되므로 모든 $CASE$ 가 병렬로 수행 가능하다. 이 때, 하나의 $CASE$ 에는 사용자로부터 정의된 병렬 수행을 위한 응용의 개수(N)만큼의 JOB 이 포함되며, 본 시스템에서는 7개로 정의하였다. JOB 은 하나의 응용에 대한 수행을 의미 하는데, 사용자가 원하면 선택적으로 그리드 자원 상에 제출될 수 있도록 설계되었으며 이러한 계층적 관리 체계는 복잡한 워크플로우의 전체 수행에서 발생 할 수 있는 오버헤드 생성을 막고 사용자에게 실험 흐름에 대한 제어를 가능하게 한다. 더불어 효율적인 작업들의 수행을 유지하는 데에 도움을 줄 뿐만 아니라 전체 수행의 완성적인 흐름에 기여한다.

```
<job id="ID1001" namespace="Pre_f" name="def_prob" >
  <argument> 'AMACH: 0.73d0, RE: 6.5d6, AOA: 2.79d0,
  TOL:1.0d-4, TINF: 290.0d0,INTWRT2: 200, ITMAX2: 200,
  STEADINESS: 0'</argument> </job>
<job id="ID1002" namespace="Pre_f" name="upload_mesh">
  <argument>-o /storage/.../mesh/mesh.grd <filename
  file="mesh.grd"/> </argument>
  <uses file="mesh.grd" link="output" type="mesh"/></job>
<job id="ID1003" namespace="Pre_e" name="execute_pre">
  <argument> auto <filename file="mesh.grd"/></argument>
  <uses file="mesh.grd" link="input" type="mesh"/></job>
<child ref="ID1002">
  <parent ref="ID1001"/> </child>
  <child ref="ID1003">
  <parent ref="ID1002"/> </child>
</adag>
```

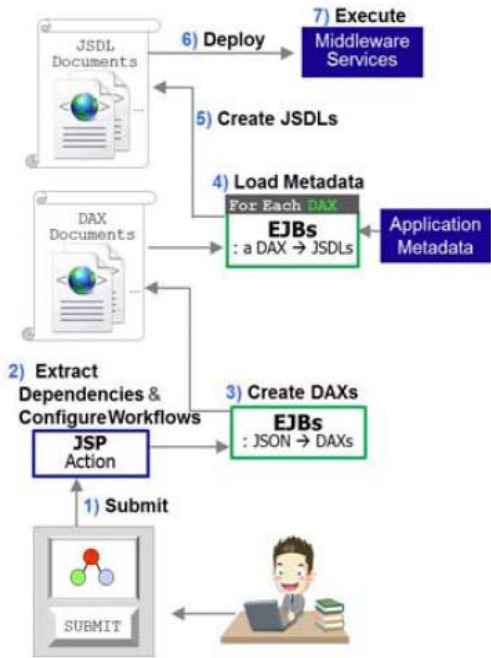
(그림 4) 워크플로우 명세의 예

4.3 워크플로우 수행 절차

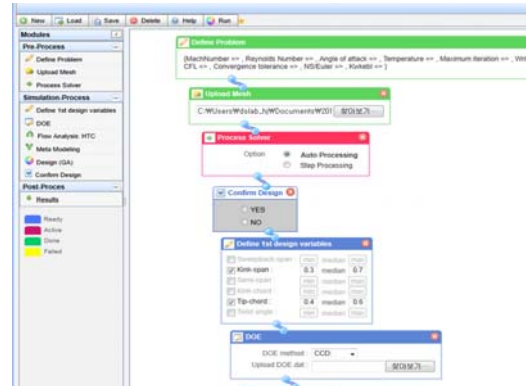
사용자가 포탈을 통해 설계한 워크플로우를 제출 하면 포탈 인터페이스를 통해 모듈의 순서와 입력 값들을 포함한 파일이 생성된다. 포탈 시스템에서는 전 처리에 해당하는 부분과 시뮬레이션, 그리고 후처리 과정에 해당하는 부분을 분리하여 워크플로우의 명세 형태인 DAX (Directed Acyclic graph in XML)[7]를 생성하도록 구성되었다. 이 이유는 각 3 단계마다 사용자의 의사 결정이 반영 되어 그 흐름을 제어할 수 있어야 하므로 모듈 간의 독립적인 수행을 보장하고, 수행 결과에 따라 반복 또는 재 수행이 가능하도록 하기 위함이다. 따라서 워크플로우 설계 서비스로부터 3 단계에 따라 JSON (Java Script Object Notation) 명세를 각 작업들의 의존성을 고려하여 DAX로 변환해준다. (그림 4)는 생성된 워크플로우 명세의 일부이며 워크플로우의 전처리 모듈 부분을 나타낸다. 따라서 명세의 속성들 중 namespace의 값이 'Pre'로 시작되어 표현된다. 시뮬레이션 또는 후 처리에 속한다면 'Simul', 'Post'로 각각 표현된다. 계산 작업들의 경우 자원에서 처리 가능한 형태로 변환되어야 하므로 구별되어 표현 되는데, (그림 4)에서 볼 수 있듯이 전 처리 과정을 의미하는 'Pre'에 실행 파일임을 식별하기 위한 'e'를 덧붙인 'Pre_e'로 표현 된다. 'f'가 붙은 경우는 파일 업로드나 입력파일 생성 같이 로컬 자원에서 처리될 수 있는 모듈을 의미한다. 표현된 DAX는 계산 작업에 한해서 JSDL (Job Sub-mission Description Language)[8] 형태로 변환되며 미들웨어를 통해 계산 자원에 제출 됨으로써 처리된다. (그림 5)는 이러한 전체 처리 흐름을 보인다.

5. 과학 워크플로우 설계 환경 구현

과학 워크플로우 수행 환경은 포탈의 그래픽 기반 인터페이스를 통해 접근 가능한 형태로 연결됨으로 전반적



(그림 5) 워크플로우의 처리 흐름



(그림 6) 워크플로우 설계 서비스를 통한 실험 설계

Process Workflow				
Pre-Processing				
Analysis	Start Time	End Time	Status	Result
Surfmod	0011:00:12	0012:45:22	Done	Download
Gridcut	0013:02:03	0013:55:38	Done	Download
HoleSearch	0014:44:14	0015:22:48	Done	Download
DonorSearch	0015:22:54	0016:13:52	Done	Download
Donorrem	0016:13:59		Active	
Flow Solver			Ready	

(그림 7) 작업 모니터링 서비스

인 과학 워크플로우 설계 환경이 완성된다. 과학 워크플로우 설계 환경의 접근을 제공하는 사용자 인터페이스 구현에는 YUI[9] 기반의 워크플로우 형태의 인터페이스를 다루는 오픈 소스 라이브러리 Wire-It[6] API를 이용하였으며, 실험 구성에 필요한 단계는 수행을 위한 세 가지 모듈 내의 세부 모듈로 구성하였다. 사용자는 모듈마다 실험 설정을 위한 파라미터 값 정의 및 여러 속성 값에 대한 구체적인 설계가 가능하며, 모듈 간의 입/출력 터미널을 와이어 연결을 통해 하나의 데이터 흐름으로 구성하여 워크플로우를 설계 할 수 있다. 동시에 워크플로우 설계에 필요한 흐름을 사용자가 쉽게 연결하고 삭제할 수 있으며, 연속적인 실험 과정의 편집을 하나의 인터페이스 내에서 편리하게 조작 할 수 있다. 구성된 실험은 내부적으로 직렬(Serial), 병렬(Parallel) 작업의 수행과 작업의 분기(Fork), 제어 흐름 등의 기본적인 워크플로우 수행이 모듈로서 정의돼 사용 가능하고, 사용자는 이러한 내부적인 실행에 신경 쓰지 않아도 워크플로우를 쉽게 설계해나갈 수 있다. 워크플로우 설계 도중에 언제든지 작업 정보를 저장하거나 기존 설계 히스토리에 대한 로드가 가능하며, 최종 완성된 워크플로우는 'Run' 버튼을 클릭함으로써 제출된다.

(그림 6)은 실제 구현된 워크플로우 설계 서비스를 통해 워크플로우를 설계하고 수행한 모습이다. 그림의 좌측 부분은 실험의 수행 모듈의 구분에 따라 세부 모듈들이 정의되어 원하는 모듈을 드래그하여 캔버스에 위치시킨다. 캔버스의 우측에는 사용자가 설계한 워크플로우에 대한 전체 흐름을 추상화한 미니 맵과 수행 중인 워크플로우의 구체적인 상태 정보를 보여주는 창이 위치된다. 계산 작업이 실행되는 경우에는 진행 흐름에 단계 별 중간 결과물과 수행 로그를 검사하여 연구 진행 방향에 대한 상세한 검증 작업을 수행해야 하는 연구의 특성과 다량의 작업을 수행한다는 점을 반영하여 진행 되는 작업과 사용자가 원하는 작업 위주의 모니터링 서비스를 제공한다. 따라서 전체 시스템에서 제공 되는 모니터링 서비스는 두 가지 형태로 제공 한다. 먼저 계산작업들을 위한 모니터링 서비스는, 수치 해석 모듈의 수행 시작과 함께 주기적으로 포함된 작업 상태를 체크하며, PBS (Portable Batch Scheduler) 혹은 SGE(Sun Grid Engine) 같은 로컬 작업 매니저를 통해 작업의 상태 정보가 데이터베이스 시스템에 저장되고 업데이트 값을 포탈 상에 (그림 7)과 같이 표현된다. HTC의 수행은 다수의 문제(Case)

들이 생성되고 각 상태 정보를 효율적으로 제공해야 하므로 Case마다 간략한 파라미터 정보와 결과 값의 일부를 요약하여 제공한다. 전체 워크플로우에 대한 모니터링은 작업이 진행 됨에 따라 각 모듈마다 색의 변화를 통해 상태 변화를 직관적으로 확인 할 수 있도록 하며 수행 작업에 대한 수렴 그래프를 실시간으로 제공받을 수 있다.

6. 결론 및 향후 연구

본 논문에서는 계산과학 응용을 위한 과학 워크플로우 통합 수행 환경을 제안하였으며, 항공우주 응용에 특화된 워크플로우 중심의 문제풀이 환경을 제공하는 포탈 시스템을 구현하였다. 이러한 환경의 제공은 대규모 계산자원의 확충 및 접근의 어려움으로 타 연구 분야에 비해 상대적으로 미흡한 발전을 보인 계산과학 연구자들에게 더욱 활성화 된 연구 환경을 제공할 수 있고, 활용도를 높여 고부가 가치의 상품과 기술을 쉽고 빠르게 만들 수 있으며 더불어 국가 과학 기술 경쟁력을 높일 것으로 예상된다.

향후 연구로 본 연구팀에서는 워크플로우의 수행에서 다수의 작업들의 일괄 수행으로 발생될 수 있는 작업 부하를 줄이기 위한 효율적인 스케줄링 정책에 대한 연구와 함께 폭 넓은 컴퓨팅 자원 제공을 위하여 그리드 환경 또는 클라우드 환경에 선택적으로 작업 제출이 가능한 고성능 컴퓨팅 환경 구성에 대한 연구로 이러한 실험 환경과 함께 제공할 예정이다.

참 고 문 헌

- [1] 조정현, 허신영, 김윤희, 김종암, 조금원, "e-Science 기반 사이버 교육을 위한 유체 해석 연구 시스템", KNOM Review, Vol. 12, No. 1, June, 2009, pp. 42-50
- [2] Pegasus-WMS, <http://pegasus.isi.edu/>
- [3] Oinn T, et al. "Taverna: A tool for the composition and enactment of bioinformatics workflows", *Bioinformatics* 2004; 20 (17):3045-3054B.
- [4] Majithia S, et al, "Triana: A graphical web service composition and execution toolkit." *IEEE International conference on Web Services(ICWS '04)*, SanDiego, U.S.A., 6-9 July, 2004
- [5] S.Y.Kim, H.J.Kang, K.A.Yoon, Y.H.Kim, C.Y.Hur, C.A.Kim, "Scientific workflow support of multi-stage experiments on an e-Science environment", *ICONI* 2010, pp. 663-667, Dec. 16-20, 2010
- [6] Wire-It, <http://neyric.github.com/wireit/>
- [7] DAX, DAG (Directed Acyclic Graph) in XML, <http://pegasus.isi.edu/wms/docs/3.0/perl/Pegasus/DAX/ADAG.html#name>
- [8] JSDL, Job Submission Description Language, <http://forge.gridforum.org/projects/jsdl-wg>
- [9] YUI, Yahoo User Interface, <http://developer.yahoo.com/yui/>
- [10] Computational Fluid Dynamics, <http://www.cfd-online.com/>
- [11] e-Science, www.nesc.ac.uk/nesc/define.html
- [12] Bertram L., Mathias W., Timothy M., Shawn Bowers, "Scientific Workflows: Business as Usual?" *7th Intl. Conf. on Business Process Management (BPM)*, LNCS 5701
- [13] 강희중, e-Science 추진동향 및 시사점, 과학기술 정책 제21권 제 1호
- [1] 조정현, 허신영, 김윤희, 김종암, 조금원, "e-Science 기반 사이버 교육을 위한 유체 해석 연구

● 저 자 소개 ●

김 서 영



2010년 숙명여자대학교 컴퓨터학과 졸업(학사)
2010년~현재 숙명여자대학교 대학원 컴퓨터과학부 컴퓨터학과 석사과정
관심분야 : 그리드/클라우드 컴퓨팅, 메타 스케줄링, 워크플로우 제어 etc.
E-mail : sssyyy77@sookmyung.ac.kr

윤 경 아



2010년 제주대학교 컴퓨터공학과 졸업(학사)
2010년~현재 숙명여자대학교 대학원 컴퓨터과학부 컴퓨터학과 석사과정
관심분야 : 그리드 컴퓨팅 환경, 시맨틱 웹 서비스, 지능형 시스템 등
E-mail : yoonka@sookmyung.ac.kr

김 윤 희



1991년 숙명여자대학교 전산학과 졸업(학사)
1996년 Syracuse University 전산학과 졸업(석사)
2000년 Syracuse University 전산학과 졸업(박사)
1991년~1994년 한국전자통신연구소(ETRI) 연구원
2000년~2001년 Rochester Institute of Technology 컴퓨터공학과 조교수
2001년~2004년 숙명여자대학교 컴퓨터학과 조교수
2004년~2009년 숙명여자대학교 컴퓨터학과 부교수
2009년~현재 숙명여자대학교 컴퓨터과학부 교수
관심분야 : 그리드/클라우드 컴퓨팅, 오토노믹 컴퓨팅, 워크플로우 제어 etc.
E-mail : yulan@sookmyung.ac.kr