


 특집  
04

## e-Science 워크벤치를 위한 그리드 서비스 통합 기술

서영균 · 김병상  
(한국과학기술정보연구원)

1. 서론
2. 본론
3. 관련 연구
4. 결론 및 미래 과제

### 1. 서론

e-Science는 연구자들이 언제, 어디서나 협업할 수 있도록 그리드 기술 및 글로벌 환경에서의 계산 자원 및 데이터 자원 그리고 유사 분야의 연구자들이 협력을 통하여 기존의 환경에서는 해결할 수 없었던 보다 복잡하고 거대한 문제를 해결할 수 있도록 지원하는 전산 및 시스템적 패러다임이다[1]. e-Science를 통해, 고에너지물리학, 생물학, 신약 개발, 항공우주공학 등 다양한 과학 및 공학 분야의 연구자들은 전에는 해결할 수 없었던 복잡한 문제를 함께 논의하고 시뮬레이션하고 실험할 수 있게 된다. e-Science 서비스 통합 워크벤치는 연구자 간의 협업을 위한 가상 실험 환경을 구축하여, 그리드 기반의 고성능 컴퓨팅 및 고출력 컴퓨팅, 다양한 매개변수의 설정을 통한 매개변수 기반 연구, 시각화를 통한 문제 토의 등을 지원한다. 하지만 이와 같은 다양한 연구 활동은 일회적이며 단일한 과정이 아닐 뿐만 아니라 과거의 정보를 통하여 분석하고 유추하여 새로운 정보를 생산할 수 있는 상호 연관성이 매우 높은 과정이라 할 수 있다. 따라서, e-Science 환

경에서의 응용 연구 활동은 보다 정형적으로 전, 후 관계를 정의하고 그러한 관계 속에서의 응용 연구에서 생산된 데이터를 관리함으로써, 보다 효율적으로 다른 연구 활동에 활용될 수 있도록 해야 한다. e-Science 워크벤치는 다양한 응용 연구 활동을 전산기술을 기반으로 단일화된 서비스로 정의를 하고 응용 연구자들에서 서비스를 인지할 수 있는 매커니즘을 제공하고 가용한 서비스를 호출할 수 있는 표준화된 인터페이스를 제공함으로써 보다 모듈화 되고 자동화된 문제 해결 환경을 제공하고자 한다. 하지만 이러한 연구 활동들이 서비스로 정의될 때 기존의 웹서비스 기반의 구조는 비즈니스 영역에 초점을 두고 설계되어졌기 때문에 과학 영역의 문제를 다루기에는 부족한 부분이 있다. 과학 영역의 문제는 단순한 서비스의 정의 이외에 장시간의 실행시간동안의 서비스의 상태관리 및 어플리케이션 의존적인 데이터관리가 함께 이루어져야 하기 때문이다. 특히 분산된 자원을 고려해야 하는 그리드 환경에서는 자원에 분포되어 있는 서비스를 획득하고 호출하는 기술과 더불어 서비스의 실행 중 상태 및 데이터의 관리를 함께 고려해야 한다.

따라서, 본 논문에서는 이러한 이슈를 고려하여, 워크벤치가 이기종의 분산된 자원에 분포된 그리드 서비스를 인지하고 호출하고 실행하며 상태를 관리할 수 있도록 설계된 구현 아키텍처를 제안한다. 2장에서는 본 논문이 다루게 될 문제를 정의하고, 3장에서는 간단한 접근 방법을 제시한다. 4장에서는 제안된 구현하고자 하는 시스템 구조를 설계하고 5장에서는 세부적인 구현 내용을 다루며, 5장에서는 결론 및 미래 과제에 대해서 언급한다.

## 2. 본 론

### 2.1 문제 설명

본 장에서는 e-AIRS(e-Aerospace Integrated Research System)[1] 시나리오를 기반으로 우리가 해결해야 할 문제를 설명한다.

e-AIRS는 항공우주 공학 연구자들이 e-Science 인프라를 기반으로 계산 및 실험 연구를 포함하는 항공우주 연구 프로세스를 지원하기 위해 설계된 가상 조직 시스템이다. e-AIRS에서는 (그림 1)과 같은 연구 흐름을 갖는다.

(그림 1)에서, 연구자는 Mesh 애플리케이션을 실행하여 Mesh 설계에 필요한 다양한 입력 조건을 갖는 매개변수 조합을 토대로 Mesh 데이터 파일을 생성하고, 이를 CFD Solver 서비스에 전달한다. CFD Solver서비스는 또 다른 매개변수 조합-Mach 숫자, Reynolds 숫자, 흐름 내 압력/온도 등-을 입력 받아, 준비된 Mesh 데이터 파일과 함께 결과를 생성해 낸다. 이 결과 중, 실험이 필요하다고 생각되는 중요한 부분은 실제로 수치 풍동 실험 요청을 하고, 나머지 부분은 Visualization 서비스를 이용해 결과에 대한 토의를 진행한다.

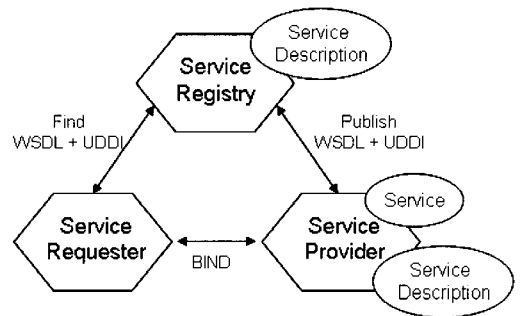
이 흐름에서, 그리드 서비스는 CFD Solver에서 Visualization까지 굵은 선으로 표시된 부분에서 활용된다. 즉 연구자는 CFD Solver와 같은 응

용 코드(legacy code)나 Visualization 도구를 이용하여 대용량의 데이터 처리와 계산을 기반으로 하는 작업을 실행하기 위해, PC 기반의 클러스터 계산 환경이 아닌 그리드 환경을 이용하게 되는 것이다. 이 코드나 도구들은 표준 인터페이스를 제공하는 웹 서비스의 형태로 그리드 상에 배포되어 있다.

따라서, 우리 워크벤치에서 위의 그리드 서비스를 활용하기 위해서는 작업 요청을 하고자 하는 그리드 서비스를 발견하고, 이것을 호출하기 위한 클라이언트를 워크벤치에 등록하고, 실제로 이 클라이언트를 이용해 작업을 실행할 수 있는 구조를 택해야 한다.

### 2.2 해결전략

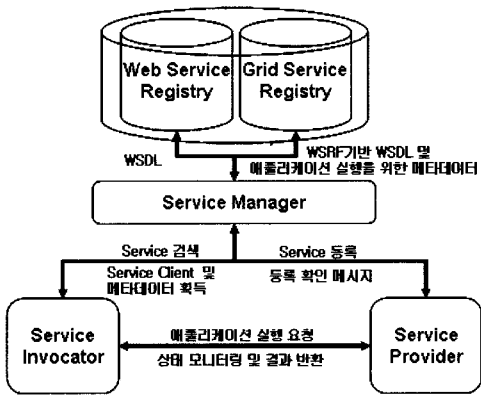
물론, 우리 워크벤치는 웹 서비스의 호출 및 실행이 근간이 된 연구 흐름을 작성하고 실행하므로, 일반 Publish/Subscribe 시스템에서 통용되고 있는 (그림 2)와 같은 구조를 취할 수도 있다.



(그림 2) Publish/Subscribe 시스템 구조

그러나, 클라이언트를 통해 작업을 제출하고 실행하기 위해서 필요한 해당 서비스를 발견하더라도 다음의 고려 사항으로 인해 우리의 워크벤치는 그것과는 좀 더 다른 구조를 취해야 한다. 첫째, 그 서비스는 그리드 자원을 기반으로 한 응용 코드를 포장한(wrapped) 것이기 때문에, 입력 받아야 할 매개 변수가 많고 복잡한 스크립트를 요구할 수 있다. 둘째, 실제로 많은 계

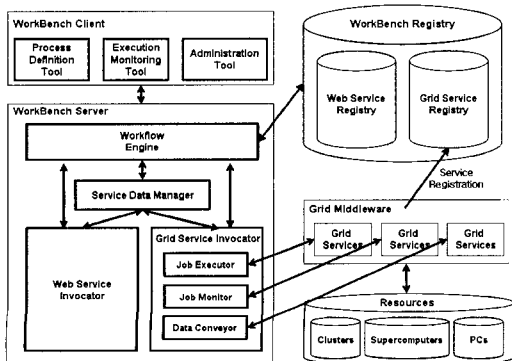
산과 대용량의 데이터 처리가 이루어 질 수 있으므로 서비스의 실행 결과를 돌려받는데 걸리는 시간이 길어질 수 있다. 셋째, 연구자는 자신이 해당 서비스에 제출한 작업의 실행 상태를 모니터링 받을 수 있어야 한다. 따라서, 워크벤치의 그리드 서비스 실행은 (그림 3)과 같은 요약된 (abstract) 구조를 통해 이루어져야 한다. 다음 장에서 그리드 미들웨어의 그리드 서비스를 지원하는 워크벤치의 좀 더 상세한 구현 아키텍처를 설명한다.



(그림 3) 요약된 그리드 서비스 실행 구조

### 2.3 그리드 서비스 연동 구조 설계

(그림 4)는 워크벤치를 중심으로 그리드 미들웨어 자원과의 연동 관계를 상세히 나타낸 전체 아키텍처이다.



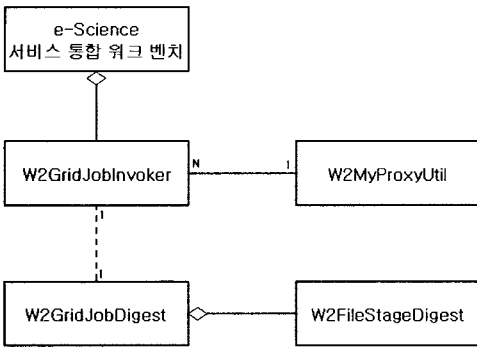
(그림 4) 그리드 서비스 연동 아키텍처

워크벤치는 크게 클라이언트, 레지스트리, 서버로 구성된다. 클라이언트는 연구 흐름을 설계하는 프로세스 정의 도구, 설계된 흐름을 모니터링하는 실행 모니터링 도구, 관리자를 위한 관리 도구로 구성된다. 레지스트리는 (그림 3)의 구조와 동일하게 웹 서비스 레지스트리와 그리드 서비스 레지스트리로 이루어져 있는데, 그리드 자원은 자신의 그리드 서비스와 그것에 메타데이터를 이곳에 등록해야 한다. 서버는 흐름 엔진, 서비스 데이터 매니저, 웹 서비스 실행자, 그리드 서비스 실행자로 구성된다. 흐름 엔진은 설계된 연구 흐름의 각 활동의 순차적인 흐름에 대한 제어를 담당한다. 활동은 크게 서비스의 실행과 같은 자동 활동과 실험 요청과 같은 수동 활동으로 나누어진다. 흐름 엔진이 자동 활동을 수행하기 위해 웹 서비스나 그리드 서비스의 실행이 요청될 경우, 워크벤치 레지스트리에 저장된 각 서비스 레지스트리에 질의하여 서비스의 URL 및 서비스 실행을 위한 메타데이터 등을 가져온다. 서비스 데이터 매니저는 흐름 엔진의 명령에 따라 웹 서비스 실행자와 그리드 서비스 실행자에 의해 생성되는 결과 데이터를 저장, 관리한다. 웹 서비스 실행자는 자동 활동에 해당하는 웹 서비스를 실행한다. 그리드 서비스 실행자는 작업 실행자, 작업 모니터, 데이터 전달자로 구성되어 있다. 각 구성요소는 PC, 클러스터, 슈퍼컴퓨터 등의 자원과 연동하는 그리드 미들웨어와 통신한다. 작업 실행자는 그리드 미들웨어가 지원하는 특정 그리드 서비스의 실행을 담당하며, 그 서비스의 실행을 위해 서비스 데이터 관리자로부터 받은 메타데이터를 입력 매개변수를 설정하여 이를 전달한다. 작업 모니터는 실행되고 있는 그리드 서비스의 작업 상태를 모니터링한다. 데이터 전달자는 작업이 완료된 그리드 서비스가 생성한 결과 파일이나 데이터를 넘겨받아 서비스 데이터 관리자에 전달한다.

따라서, 본 아키텍처에 의해 개발되고 있는 위

크벤치는 이기종의 미들웨어에 대응하는 그리드 서비스 실행자를 구현함으로써, 다양한 그리드 서비스의 지원을 가능하게 하는 뛰어난 유연성을 갖게 될 것으로 기대된다. 4장에서, 우리 구현 아키텍처에서 가장 핵심이 되는 그리드 서비스 호출자의 세부 구현 내용을 알아본다.

### 2.4 세부 구현(Implementation Details)



(그림 5) 그리드 서비스 통합을 위한 클래스 다이어그램

본 논문에서 제안하는 (그림 4)의 Grid Service Invocator가 구현된 (그림 5)는 e-Science 워크벤치에서 그리드 서비스 통합을 지원하기 위해 설계되고 구현된 클래스 다이어그램을 나타낸다. 워크벤치에서 작성된 작업 실행 파일, 즉 JSDL(Job Submission Description Language)은 dom4j를 이용하는 W2GridJob Digest 클래스를 통해 생성된다. W2GridJob Digest 클래스는 작업 실행을 위해 StageIn 파일 정보와 StageOut 파일 정보를 내부에 포함하고 있는데, 이에 대한 정보를 다루는 클래스가 W2FileStageDigest이다. W2GridJobInvoker 클래스는 JSDL 문서를 전달받아 실제 그리드 서비스를 호출한다. W2GridJobInvoker 클래스는 IBM에서 공개한 GramJob 클래스를 기반으로 하며, 내부적으로 그리드 서비스 서버로부터 의뢰된 그리드 작업의 상태 변경 정보를 통보받기

위해 GramJobListener 인터페이스가 구현되어 있다. W2GridJobInvoker 클래스가 그리드 서비스를 호출하기 위해서는 그리드 서비스 호출자의 Proxy Credential을 얻어야 하는데 이때 W2MyProxyUtil 클래스를 이용하게 된다. W2MyProxyUtil 클래스는 MyProxy 서버와 연관되는 작업 전반을 지원하는 클래스로서 Proxy Credential의 생성, 등록, 사용 등의 서비스를 제공한다. 현재 삭제 서비스는 추후에 추가될 예정이다.

### 3. 관련 연구

e-Science 워크벤치는 매우 중요한 기능으로서 부각되고 있다. 특히 생명정보학분야에서는 산재되어 있는 생명정보 데이터베이스의 접근을 통한 데이터 획득, 데이터분석에 있어서 웹서비스 기반의 구현이 진행되고 있다. 특히 Taverna 워크벤치[3]는 산재한 관련 웹서비스를 플러그인으로 제공을 한다. 하지만 Taverna 워크벤치는 대용량 자원과 장시간이 요구되는 그리드 환경을 지원하고 있지는 않으며 순수한 웹서비스의 WSDL을 통해서만 서비스의 정보를 획득할 수 있다. 또한 Kepler workflow [4]는 그리드 환경에 적합하도록 계획되었다. 특히 보안, 작업 실행, 데이터 전송 등의 기능을 직접적으로 워크플로우상에 명시할 수가 있다. 미들웨어로서 현재 Kepler는 Globus를 사용함으로써 GRAM, GridFTP, Proxy Certificate Generator 등을 통해서 그리드 환경에 접근할수 있다. 하지만 Kepler 는 서비스의 관점에서 그리드 어플리케이션에 초점을 두지 않고 있다. 이것은 그리드 환경에 접근할수 있는 기능만을 부여할 뿐 그것을 통하여 어떠한 어플리케이션이 활용가능한가와 같은 메타정보를 제공하지 않는다. 따라서 사용자는 어플리케이션의 실행을 위해 필요한 모든 정보를 사전에 알고 있어야 하는 전제가 필요하다. GridLab 에서 진행되고 있는 Triana[5]는

그리드 환경의 통합을 위해서 GAT(Grid Application Toolkit) [6]의 인터페이스를 이용한다. GAT는 GSI(Grid Security Infrastructure) 기반의 작업 제출 및 데이터 전송 기능 지원해준다. 하지만 Triana 도 Kepler 와 마찬가지로 워크벤치에서는 단순히 그리드 자원에 접근할 수 있는 방법만을 제공해준다. 따라서 실행을 위해 필요한 어플리케이션에 대한 메타 정보는 사용자가 이미 알고 있다는 것을 전제하고 있다.

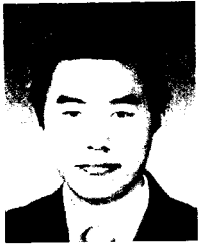
#### 4. 결론 및 미래 과제

우리는 본 논문에서 예제 시나리오를 바탕으로 e-Science 워크벤치의 그리드 서비스 지원 아키텍처에 대한 설계 및 구현 내용에 대해서 설명하였다. 본 아키텍처에 의해 구현된 워크벤치를 통해 연구자들은 다양한 그리드 미들웨어 자원에서 제공하는 서비스를 발견하고, 등록하고, 사용할 수 있을 것으로 기대된다. 현재 한국과학기술정보연구원 e-Science사업단 응용연구기술개발팀에서 개발 중인 워크벤치는 Globus 미들웨어 자원의 서비스만 지원하고 있으나, 향후 gLite 미들웨어 자원의 서비스 등도 사용할 수 있도록 이를 확장할 예정이다.

#### 참고문헌

- [1] Web Services Architecture: <http://www.research.ibm.com/journal/sj/412/gottschalk.html>
- [2] Byungsang Kim, Dukyun Nam, Young-Kyoon Suh, June Hawk Lee, Kumwon Cho, and Soonwook Hwang, "Application Parameter Description Scheme for Multiple Job Generation in Problem Solving Environment", IEEE International Conference on e-Science and Grid Computing, 2007(to be appeared)
- [3] T. Oinn, M. Addis, J. Ferris, D. Marvin, M. Senger, M. Greenwood, T. Carver and K. Glover, M.R. Pocock, A. Wipat, and P. Li. Taverna: a tool for the composition and enactment of bioinformatics workflows. *Bioinformatics*, 20(17):3045-3054, Oxford University Press, London, UK, 2004.
- [4] B. Ludascher, I. Altintas, C. Berkley, D. Higgins, E. Jaeger-Frank, M. Jones, E. Lee, J. Tao, Y. Zhao, Scientific Workflow Management and the Kepler System. *Concurrency and Computation: Practice & Experience*, 18(10), pp. 1039-1065, 2006.
- [5] I. Taylor, M. Shields, and I. Wang. Resource Management of Triana P2P Services. *Grid Resource Management*, Kluwer, Netherlands, June 2003.
- [6] G. Allen, K. Davis, K. N. Dolkas, N. D. Doulamis, T. Goodale, T. Kielmann, A. Merzky, J. Nabrzyski, J. Pukacki, T. Radke, M. Russell, E. Seidel, J. Shalf, and I. Taylor. Enabling Applications on the Grid : A GridLab Overview. *International Journal of High Performance Computing Applications (JHPCA)*, Special Issue on Grid Computing: Infrastructure and Applications, SAGE Publications Inc., London, UK, August 2003.

## 저자약력



**서영권**

2003년 경북대학교 컴퓨터과학과 학사  
2005년 KAIST 전자전산학과 전산학전공 석사  
2005년 3월~한국과학기술정보연구원(KISTI)  
e-Science사업단 응용연구개발팀 연구원  
관심분야 : e-Science, Grid, 워크플로우, XML, 정보 통합  
이 메 일 : yksuh@kisti.re.kr



**김병상**

2002년 동국대학교 산업공학과 학사  
2004년 한국정보통신대학교 전자통신학과 석사  
2004년 3월~한국정보통신대학교 전자통신학과 박사과정  
2006년 3월~한국과학기술정보연구원(KISTI)  
e-Science사업단 응용연구개발팀 연구원  
관심분야 : Grid Middleware and Job Scheduling, Problem  
Solving Environment (PSE)  
이 메 일 : bskim@kisti.re.kr