

## Grid에서의 서비스 기반 RTI 구현

최기영<sup>0</sup>, 이태동, 정혜선, 유승훈, 장재형, 임중호, 정승환, 정창성  
고려대학교 전자컴퓨터공학과

{2xx195<sup>0</sup>, lyadlove, friendlyu, jjh3368, jhlim}@snoopy.korea.ac.kr, [csjeong@charlie.korea.ac.kr](mailto:csjeong@charlie.korea.ac.kr)

### Service-oriented RunTime Infrastructure on Grid

Ki-young Choi<sup>0</sup> Tae-Dong Lee, Hye-Sun Jung, Seung-Hun Yoo, Jae-Huyng Jang, Jung-Ho Lim,  
Seung-Hwan Jung, Chang-Sung Jung  
Dept. of Electronics Engineering Graduate School, Korea University

#### Abstract

HLA (High Level Architecture) 는 모델링과 시뮬레이션 분야에서 상호 이질적인 서비스에 따른 어플리케이션의 상호운용성(interoperability)과 재사용성(reusability)을 증진시키기 위해 필요한 infrastructure를 개발하기 위한 청사진이다. RTI(RunTime Infrastructure)는 HLA 의 소프트웨어적 도구로써 libRTI, FedExec, RTIExec 의 세가지 컴포넌트로 구성되어 있다. RTI는 분산환경에서 동적으로 다수대다수간의 통신을 제공하는 미들웨어(middleware)이다. 대규모의 분산 시뮬레이션의 실행은 지리적으로 퍼져있는 많은 양의 컴퓨팅 자원을 필요로 할 수 있는데 이러한 환경이 가지는 보안과 동적 자원 할당이라는 문제점을 Globus Toolkit3(GT3)안의 Open Grid Service Architecture(OGSA)를 사용해서 해결할 수 있는 RTI-G를 개발해 왔고 이를 service-oriented RTI-G라 부른다. 이 논문에서는 service-oriented RTI on Grid 의 구조와 앞에 제시된 문제점들의 해결방안을 제시하고 실험한다.

#### 1. Introduction

본 논문은 Grid 상에서 HLA[1]를 구현하기 위해 그 소프트웨어 도구인 RTI를 OGSA에 기반한 Grid service 로 만들기 위한 디자인과 구현, 실험을 다룬다.

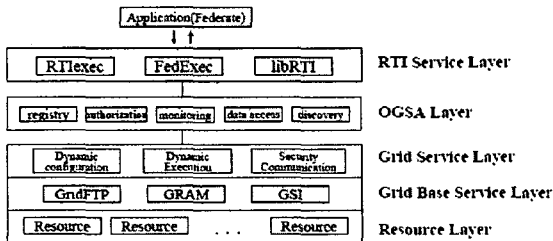
Networked Virtual Environment(NVE)는 지리적으로 떨어진 다수의 사용자들이 컴퓨터에 의해 생성된 가상 공간에서 실시간으로 상호작용을 할 수 있는 환경을 말하는데 NVE 시스템 간의 상호 이질적인 서비스에 따른 어플리케이션간의 상호운용성(interoperability)과 재사용성(reusability)을 높이기 위해 공통적인 개발 프레임워크(framework)로 미 국방성에서 개발한 HLA를 사용한다. 그러나 HLA의 디자인은 다른 조직에 의해 관리되는 instruments, displays, computational and information resources 들을 통합하기 위해 필요한 소프트웨어 어플리케이션을 제공하지 않는다. HLA의 소프트웨어 도구로써 존재하는 RTI는 분산 시뮬레이션을 위한 자원의 관리와 조정을 고려하지 않는다. 한편 Grid 는 이러한 이슈들을 정밀하게 다루기 위해 참신하게 디자인 되었다. Grid 사용을 위한 미들웨어인 Globus toolkit3 (GT3)[5] 은 RTI를 위해 유용한 특성들을 제공하는데 RTI의 시작을 위해 Grid Resource Broker, Globus Resource Allocation

Manager (GRAM)와 Metacomputing Directory Service 가 사용된다. 이전에 RTI의 실행은 RTIexec를 수동으로 설치해야 하는 불편하고 힘든 작업이었다. 그러나 Globus 를 이용함으로써 RTIexec를 remote로 원격 전송, 실행 시킬 수 있게 된다. Monitoring and Discovering Service(MDS)는 자원의 상태와 설정에 관한 정보를 discovery 하고 publish하는 표준 메커니즘을 제공하는데 MDS가 없으면 RTIexec 전송을 위해 지리적으로 다른 위치에 있는 많은 양의 자원의 정보들을 미리 알고 있어야만 한다. 또한 HLA가 federate들의 join을 위한 인증을 제공하지 않는데 따른 보안문제를 GT3의 Grid Security Infrastructure (GSI) service 를 이용해서 해결할 수 있다. 본 논문에서는 위에서 제시한 HLA 와 GT3의 결합을 service-oriented RTI-G 라 부르고 이를 GT3의 factory service를 이용하여 구현하고 실험한다.

#### 2. Related Works

현재의 RTI는 Department of Defense(DoD) 에 의해 개발된 RTI1.3NG[2]와 기업에서 개발한 pRTI[3]와 MAK RTI[4], 대학에서 개발한 RTI-Kit가 각각의 특징들을 가지고 발전해 왔다. 이러한 HLA community들의 노력에도

불구하고 collaboration, 데이터 공유, 그리고 다른 새로운 모델과의 interaction은 분산 컴퓨팅이 해결해야 할 고민거리였다. 그 결과, 내부와 외부 시스템간의 interconnection에 대한 관심이 증가했고 이러한 진화는 새로운 분산 어플리케이션 개발과 배치 라는 요구를 창출했다. 분산된 소프트웨어, 하드웨어 그리고 human 리소스를 바람직한 QoS를 보장하면서 동적으로 사용하고 싶은 것이 그것이다. 이것의 해결책이 OGSA[7] 이다. OGSA는 어플리케이션들에게 넓은 지역에 퍼져있는 분산된 서비스들과 자원들의 공유와 접근을 허용한다[6].

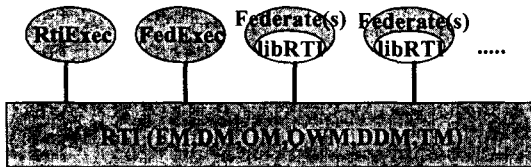


[그림 1] Architecture of service-oriented RTI on Grid

3. Design of service-oriented RTI on Grid

3.1 RTI service layer

RTI는 시뮬레이션 진행에 필요한 시간 관리 및 객체 관리 등의 시뮬레이션 서비스들을 제공하는 소프트웨어 도구로써 RTI에서 제공될 서비스의 종류 및 기능은 HLA 접속 사양(Interface Specification)에 정의되어 있다.

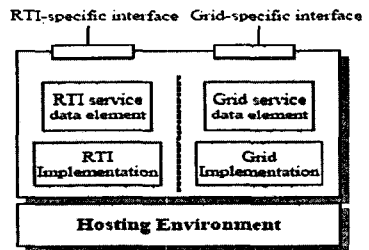


[그림 2] Components of RTI

RTIExec는 FedExec의 생성과 소멸을 담당한다. FedExec는 Federate들의 참가와 탈퇴를 담당한다. Federate들은 libRTI를 통해서 RTI의 여섯 가지 서비스를 이용한다. 이들 서비스들은 federation의 생성과 federate의 참가, class 선언과 객체 생성, data filtering, 시간 전진 등의 기능을 담당한다.

3.2 OGSA layer

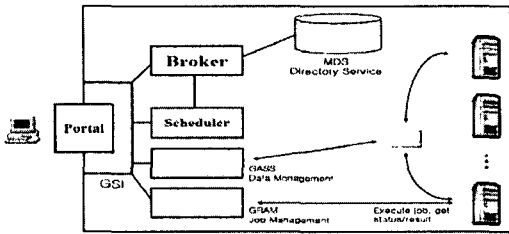
OGSA의 목적은 Grid 기반의 어플리케이션을 위한 새로운 공동의 표준 architecture를 정의하는데 있다. 이 새로운 architecture의 중심에는 Grid service의 컨셉이 있다. OGSA는 Grid service가 무엇인지, 기반으로 하는 테크놀로지의 타입이 무엇인지에 대해 정의한다. Grid service를 실행하기 위해 필요한 기술과 자세한 명세는 Open Grid Service Infrastructure(OGSI)에 주어진다. OGSA의 정의에 따라 Grid service가 제공하는 Grid core service(GridFTP, GRAM, GSI)를 사용하여 실제의 federate data와 RTI components(RTIExec, FedExec, libRTI)를 전송하고 실행하기 위한 방안으로 RTI-specific part 와 Grid-specific part 의 두 부분으로 OGSA를 디자인한다. 다른 Grid service들과 마찬가지로 WSDL(Web Service Description language)로 RTI의 specification을 기술하고 WSD(Web Service Deployment Descriptor)를 사용하여 factory에 service로 등록함으로써 client가 하나의 hosting 환경에서 multiple binding을 통해 RTI service를 invocation 할 수 있게 디자인 한다.



[그림3]OGSA division: RTI-specific part,Grid-specific part

3.3 Grid service layer

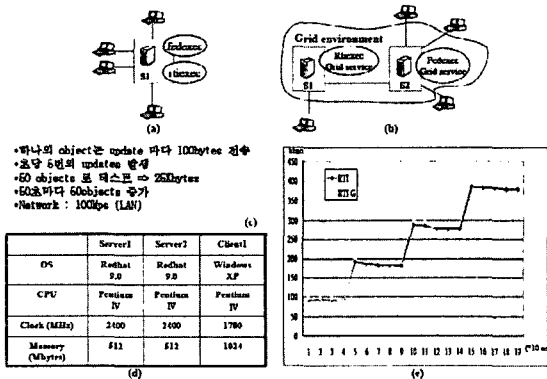
Grid service layer 에서는 보안 통신을 위한 GSI와 데이터의 전송을 위한 GridFTP와 GASS를 사용하여 RTI service layer를 위한 보안 통신과 동적 coordination을 제공한다. Grid service layer는 dynamic configuration과 execution, security components로 구성되어 있고 dynamic configuration과 execution은 Resource Broker(RB), Simple Transfer Agent(STA) 와 Remote Execution(RE)로 구성이 되어있다. RB는 MDS를 통해서 받은 정보를 기반으로 해서 어떤 리소스에 RTIExec를 실행 시켜야 하는가를 결정하게 된다. STA는 리소스 브로커를 통해서 얻은 최적의 리소스에 RTIExec를 전송하고 전송한 파일의 속성 변경을 담당하게 된다. RE는 원격지에 전송된 Rtiexec의 수행과 실행시에 필요한 환경을 설정하고 관리하는 역할을 한다. [그림 4]는 Client의 요구에 의해 Grid service가 수행되는 과정을 보여준다.



[그림4] Process of Grid service

4. Experiments

Service-oriented RTI-G는 C++언어를 사용하여 Linux에서 실험한다. 실험의 시나리오는 아래와 같다. 테스트 조건은 그림 5(c)에 있고, 5(d)는 실험에 사용되는 장비의 사양이다. 5(e)는 실험의 결과이다. 첫 번째 실험에서 네 개의 client가 server1에 접속하면 하나의 servers내에 RTIexec를 생성하고 FedExec를 만든다. 두 번째 실험에서 첫 번째와 마찬가지로 네 개의 client가 server1에 접속하면 RTIexec service를 생성하고 GT3 컨테이너로부터 fork 되어 FedExec service를 실행한다. RTI-G에서 RB, STA, RE service에 의해 제일 좋은 서버(실험에서의 Server2)에서 FedExec를 실행한다. FedExec 실행후 50초 마다 50개씩 오브젝트를 증가시키며 각각의 전송률을 테스트한다. 각 실험에서 FedExec의 실행후 (c)의 테스트 조건에 따른 데이터 전송률을 측정 한 결과 RTI-G의 성능이 RTI보다 더 나은 결과를 보였다.



[그림 5] (a)RTI 실험 (b)RTI-G 실험 (c)테스트 조건 (d)server/client 사양 (e)실험 결과

5. Conclusion

HLA의 소프트웨어 도구인 RTI는 RTIexec, FedExec 그리고 libRTI의 세가지 컴포넌트로 구성되어 있다. RTI는 분산 환경에서 다수대다수간의 동적인 통신을 제공하는 미들웨어이다. 큰 스케일의 분산 시뮬레이션을 실행

함에 있어 지리적으로 떨어져 있는 많은 양의 컴퓨팅 자원들을 필요로 할 수가 있는데 그러한 환경에서 고려해야 할 점이 보안 문제와 어떻게 동적으로 자원을 할당하고 조정할 것 인가 이다. 본 논문에서는 Globus Toolkit3(GT3)를 사용하여 이러한 문제들을 극복할 수 있는 RTI를 개발하고 이를 service-oriented RTI-G라 부른다. 그리고 실험을 통해 기존의 RTI 보다 향상되었음을 증명했다.

6. References

1. IEEE Standard for Modeling and Simulation (M&S), "High Level Architecture(HLA) Federate Interface Specification," IEEE Std 1516.1-2000
2. T.B. Stephen, J.R. Noseworthy, J.H. Frank, "Implementation of the Next Generation RTI," 1999 Spring Simulation Interoperability Workshop.
3. M. Karlsson, L. Olsson, "pRTI 1516 - Relation and Design," 2001 fall Simulation Interoperability Workshop.
4. D.W. Douglas, "Rationale and Design of MAK Real-Time RTI," 2001 Spring Simulation Interoperability Workshop.
5. I. Foster, C. Kesselman, "Globus: A Metacomputing Infrastructure Toolkit," Intl J. Supercomputer Applications, 11(2):115-128, 1997.
6. I. Foster, C. Kesselman, J. Nick, S. Tuecke, "The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration," June 22, 2002.
7. D.Talia, "The Open Grid Services Architecture: where the grid meets the Web," Internet Computing, IEEE, Volume: 6, Issue: 6, Nov.-Dec. 2002 ages:67 - 71
9. 진정훈, "Globus를 이용한 Run Time Infrastructure의 설계 및 구현"