

Globus를 이용한 Run-Time Infrastructure 설계 및 구현

진정훈⁰ 이태동 유승훈 장재형 임중호 정창성
고려대학교 전자공학과

(daein⁰, lyadlove, friendlyu, jjh3368, jhlim)⁰@snoopy.korea.ac.kr, csjeong@charlie.korea.ac.kr

Run-Time Infrastructure on GRID(RTI-G)

Jung-Hun Jin⁰, Tae-Dong Lee, Seung-Hun Yoo, Jae-Hyung Jang
Jung-Ho Lim Chang-Sung Jeong
Dept. of Electronics Engineering Graduate School, Korea University

요 약

본 논문은 GRID 상에서 HLA(High Level Architecture)를 기반으로 한 분산 객체 지향 simulation을 위한 네트워크 하부구조인 Run-Time Infrastructure (이하 RTI)의 디자인과 구현에 관해 기술한다. RTI는 지리적으로 떨어진 응용 프로그램 간의 정보 교환 외에도, 접속 사양에 정의된 다양한 서비스를 제공하는 middleware이다. GRID는 전세계에 펼쳐져 있는 자원들에 대한 관리와 접근, 사용을 위한 다양한 기능과 안전하고 편리한 security를 보장한다. 본 논문에서는 globus toolkit에서 지원하는 security와 자원 접근 방법을 사용해서 RTI에 보안과 자원의 동적 할당을 부여해서 RTI를 이용한 simulation에 뛰어난 상호 연동 능력과 대규모의 프로젝트를 위한 광범위한 자원을 보다 안전하고 효율적으로 사용할 수 있도록 하는 환경을 구현하였다.

1. Introduction

본 논문은 GRID 상에서 NVE 환경 구축을 위해 중요 사항 중 하나인 보안 통신과 동적인 환경 설정을 위한 RTI구현 및 디자인에 대해서 소개한다.

Networked Virtual Environment (이하 NVE)는 지리적으로 떨어진 다수의 사용자들이 컴퓨터에 의해 생성된 가상 공간에서 실시간으로 상호작용을 할 수 있는 환경을 말한다. NVE의 응용 분야로는 가상 소평물, 회의 시스템, 국방 분야에서의 훈련 시스템, 네트워크 게임, 분산 협력(Collaborative) 디자인 시스템 등이 있을 수 있다. 위와 같은 응용 분야를 지원하는 NVE 개발에는 네트워크 프로토콜, 분산/병렬 시스템, 데이터베이스, 보안 개발 등과 같은 다양한 전문지식이 활용될 수 있다.

NVE 시스템들의 공통적인 특징은, 참여하는 응용 프로그램들이 공간(space), 실재(presence), 시간(time), 가상 환경과 상호작용을 하는 방법 등을 공유해야 한다. NVE 시스템들은 위와 같은 내용을 지원하기 위해 응용 프로그램에게 보안 서비스와 동적인 가상 환경 구축을 위한 서비스를 제공해야 한다. 그러나, 다양한 NVE 시스템 간의 상호 이질적인 서비스 제공은 어플리케이션 간의 상호 운용성(interoperability)과 재사용성(reusability)을 떨어뜨리게 된다. 응용 프로그램간의 상호 운용성, 재사용성을 높이기 위해서는, 공통적인 개발 프레임워크/framework가 필요하다.

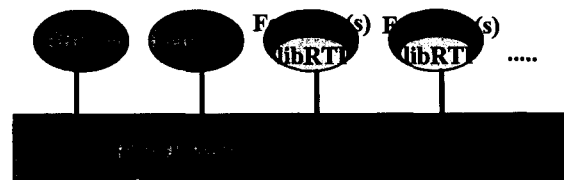
2장에서는 HLA와 Globus에 대한 전체적인 소개를 하고, 3장부터 6장에서는 RTI-G의 구조와 구현에 대해 다룬다.7장에서는 결론과 앞으로 할 일을 언급한다.

2. Overview of HLA and Globus

2.1 HLA

미국 국방성에서는 병렬/분산 시뮬레이션 개발의 효율 향상을 위한 공통 프레임워크로서 High-Level Architecture (HLA)를 제안하였으며, 아래와 같은 세 가지 구성요소로 이루어져 있다. : HLA 운용 규칙 (compliance Rules); 객체 모델 템플릿(Object Model Template); HLA 하부 구조 (Run-Time Infrastructure). HLA 운용 규칙은 HLA 기반 시뮬레이션들이 지켜야 할 논리적 규범을 서술한 것이다. HLA는 페더레이션 객체 모델(Federation Object Model: FOM)과 시뮬레이션 객체 모델(Simulation Object Model: SOM)의 이원화된 객체 모델을 사용하며, 객체 모델 템플릿은 FOM 및 SOM에 따라 정의된 객체들을 기술하는데 사용되는 객체 서술 도구이다. 마지막으로, HLA 하부구조(RTI)는 시뮬레이션 진행에 필요한 시간 관리 및 객체 관리 등의 시뮬레이션 서비스들을 제공하는 네트워크 하부구조이며, RTI에서 제공될 서비스의 종류 및 기능은 HLA 접속 사양(Interface Specification)에 정의되어 있다.

[그림 1]은 RTI의 구성 요소들을 나타낸다.



[그림 1] 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.

2.2 Globus

Globus는 GRID에서 필요로 하는 다양한 서비스들을 틀과 API로 제공하고 있다. 다음은 Globus에서 제공하는 기능들이다.

-MDS(Metacomputing Directory Service) : LDAP(Lightweight Directory Access Protocol)을 사용해서 자원의 위치와 자원에 대한 메타데이터를 제공한다.

-GRAM(Grid Resource Allocation Management) : Globus에서 자원의 할당과 관리를 담당한다. GRAM을 통해서 사용자나 원격지의 자원들을 사용할 수가 있다.

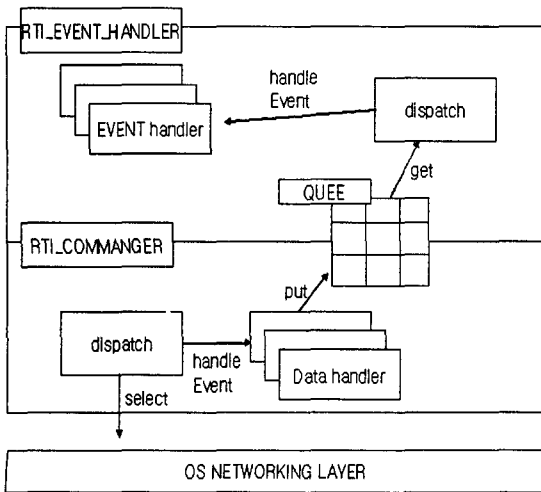
-GSI(Grid Security Infrastructure) : GSI는 Globus의 보안을 담당한다. PKI와 SSL에 기반을 두고 Single sign-on 기능을 지원한다.

-GASS(Globus Access to Secondary Storage) : 원격지에 있는 파일의 처리나 데이터의 분산 저장을 담당한다.

3. Architecture of RTI-G

RTI-G는 Globus Toolkit의 Grid Security Infrastructure(GSI)을 이용해서 RTI에서 상호 동작 하는 가상 공간에 보안 요소를 추가하고 MDS를 통해서 얻은 정보를 이용해서 동적인 FedExec의 생성을 할 수 있도록 설계 하였다.

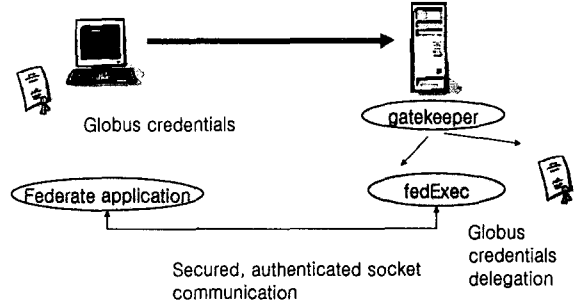
RTI-G의 전체적인 구조는 통신을 담당하는 부분인 커뮤니케이션 관리자와 커뮤니케이션 관리자가 받은 데이터를 처리하는 이벤트 핸들러 두 부분으로 크게 나누어져 있으며 두 부분은 효율성을 위해서 메시지 큐를 경계로 해서 나누어져 있다. [그림2]는 두 부분을 보여 준다.



[그림 2] Architecture of RTI

4. Security Communication of RTI-G

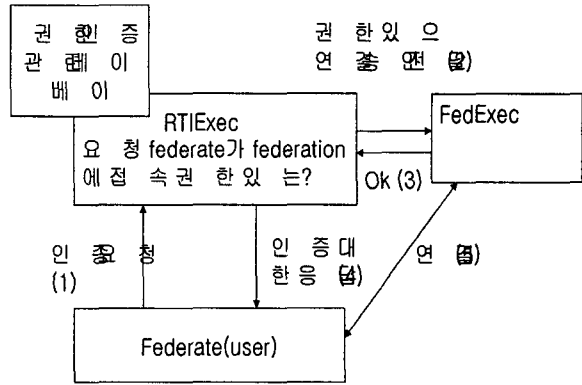
기존 RTI의 커뮤니케이션 관리자[5]는 보안과 관련된 부분이 결여가 되어 있었다. RTI-G는 GSI 이용해서 보안 커뮤니케이션을 지원을 한다. RTIExec는 보안과 관련된 정책을 수립하고 임의의 페더레이트들이 페더레이션에 참여 하지 않도록 권한과 인증을 할 수가 있다. RTI-G의 보안 통신은 globus toolkit이 제공하는 gatekeeper 을 통해서 인증이 이루어 진다. [그림3]는 이 과정을 보여 준다.



[그림 3] Security Communication of RTI-G

4.1 Community Authorization

federate가 federation에 가입하기 위해서는 자기의 인증서로 인증을 받고 자기 자신이 federation에 가입 할 권한이 있는지 RTIExec로부터 확인을 요청하고 확인을 받아야 한다. RTIExec는 권한이 있으면 FedExec에게 연결 할 federate의 정보를 전달 해준다.



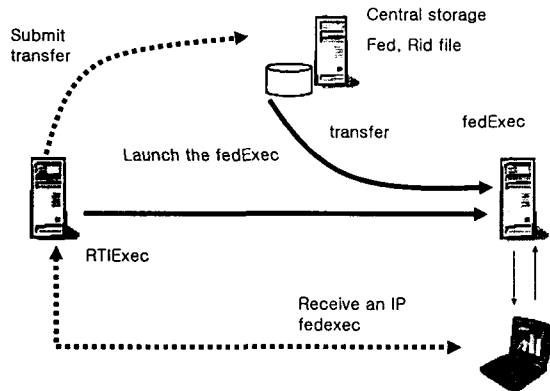
[그림 4] Process of Community Authorization

5. Dynamic Configuration of RTI-G

RTI-G는 동적인 자원 할당과 데이터 처리의 분배를 위해서 세 가지의 컴포넌트를 가지고 있다. 각각의 컴포넌트는 다음과 같다.

첫 번째는 리소스 브로커이다. 리소스 브로커는 MDS를 통해서 받은 정보를 기반으로 해서 어떤 리소스에 fedexec를 실행 시키야 하는가를 결정하게 된다. 두 번째는 전송 에이전트(transfer agent)이다. 전송 에이전트는 리소스 브로커를 통해서 얻은 최적의 리소스에 fedexec를 전송하고 전송에 파일의 속성의 변경을 담당하게 된다. 마지막으로 원격 실행 컴포넌트가 있다. 원격 실행 컴포넌트는 원격지에 전송된 fedexec의 수행과 실행시에 필요한 환경을 설정하고 관리하는 역할을 한다.

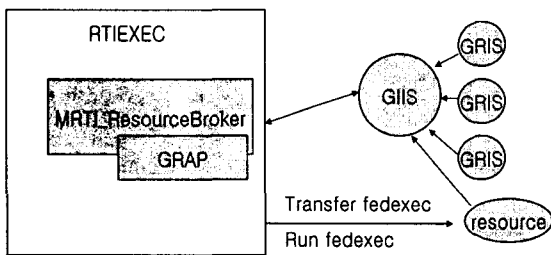
[그림 5]는 RTI-G가 Grid 위에서 fedexec의 설정 파일을 전송하고 리소스 브로커로부터 얻은 정보를 이용해서 해당 리소스에 fedexec를 전송하고 실행하는 과정을 보여 주고 있다.



[그림 5] Process of Dynamic configuration

5.1 Resource Broker

리소스 브로커는 MDS를 이용해서 얻은 메모리, cpu 정보를 바탕으로 RTIExec가 요구한 리소스를 전달해 준다. 리소스 브로커는 GRAP(grid resource allocation policy)을 이용해서 사용자가 정의한 정책에 따라서 리소스를 전달하게 된다. GRAP는 간단한 인터페이스를 가지는 클래스이고 리소스 브로커의 인자로 사용되게 된다. [그림 5]는 리소스 브로커의 구성 모양이다.



[그림 5] Process of Dynamic configuration

5.2 Transfer Agent

RTI-G에서 구현이 된 전송 에이전트는 간단하게 구현이 되어 있다. 리소스 브로커로부터 받은 리소스 정보를 이용해서 해당 리소스에 데이터 및 실행 프로그램을 전송한다. 전송된 데이터 및 프로그램은 그 파일에 사용 특성에 맞게 파일의 속성 및 사용자가 결정이 되어야 한다. 전송 에이전트는 파일의 사용 목적에 맞게 원격지 파일의 속성을 바꾸는 기능도 지원을 한다.

5.3 Remote Execution

원격 실행은 원격에 있는 실행 가능한 프로그램을 실행시키는 역할을 한다. 실행에 필요한 인자가 있다면 인자를 전달해 주고 실행을 시킨다.

6. Implementation

RTI-G의 구성 요소인 RTIExec, FedExec와 libRTI는 C++로 구현 되어 있으며 리소스 브로커는 ldap API를 이용해서 구현 되었으며 전송 에이전트는 globus 모듈인 globus ftp client와 globus gram client 모듈을 사용해서 구현 되었다. 또한 원격 실행도 globus gram client 모듈을 이용해서 구현이 되었다. 보안 통신은 globus io 모듈과 gsi 모듈을 이용해서 구현이 되었다.

7. Conclusion and Future Work

대규모의 가상 환경 구축을 위해서는 보안 통신은 중요한 요소이다. 가상 환경에 같이 참여 하지 않는 다른 사람에 의해서 가상 환경이 침해를 받는다면 그건 심각한 문제를 야기 할 수 있기 때문이다. RTI-G는 기존의 RTI가 가지고 있지 못한 보안 통신을 globus를 이용해서 grid 위에서 구현을 했으며 대규모 가상 환경 구축을 위해서 동적으로 환경을 설정 할 수 있도록 구성이 되었다. 하지만 RTI-G에는 아직 RTIExec의 위치 정보는 사용자가 알고 있어야만 한다. 그래서 우리는 차후에는 RTIExec의 위치도 디스커버리 할 수 있는 기능을 추가 해야 할 것이다.

8. 참고 문헌

- [1]. Laura Pearlman : " A Community Authorization Service for Group Collaboration "
- [2]. I. Foster, C. Kesselman, S. Tuecke , "The Anatomy of the Grid : Enabling Scalable Virtual Organizations" , International J. Supercomputer Applications, 15(3), 2001.
- [3]. DMSO, "HLA RTI-1.3NG Programmer's Guide Version 3.2"
- [4] Bart Jacob: Enabling Applications for Grid Computing with Globus
- [5]. HoKyung Kim: "Object-Oriented Design of RTI Communication Manager using Design Patterns "
- [6]. 김창훈: "GRID를 이용한 HLA 기반 객체 지향 분산 시뮬레이션 "