

액세스그리드 프레임워크

백종권^o 이태동 정창성

고려대학교 병렬분산처리 연구실

googolfx@korea.ac.kr^o, lyadlove@snoopy.korea.ac.kr, csjeong@charlie.korea.ac.kr

AccessGrid Framework

Jong-kwun Baek^o, Tae-dong Lee, Chang-sung Jeong

Parallel Distributed Processing Lab, Korea University

요 약

액세스그리드 프레임워크(AccessGrid Framework)는 지리적인 제한에 관계없이 가상적인 협업 환경을 제공하는 도구이다. 액세스그리드 프레임워크는 기존의 시스템이 갖추지 못한 사용자의 이동성 지원을 추가하고, 유비쿼터스 환경에 적절한 자동화 기능을 제공함으로써 액세스그리드 환경을 확장한다. 이들은 웹 서비스 기반의 기존 구현물인 액세스그리드 툴킷(AccessGrid Toolkit)을 적극적으로 활용하여 개발되었으며, 향후에 불안정한 종속성을 제거하여 개수될 예정이다.

1. 서 론

전 세계를 아우르는 컴퓨터 네트워크인 인터넷과 그 구성원들을 연결하는 고속망의 보급에 힘입어 원격지에 있는 다수의 컴퓨터 사용자와 얼굴을 마주보고 이야기할 수 있는 화상회의 시스템도 그 편의성으로 인해 많이 사용되고 있다.

액세스그리드 툴킷(AccessGrid Toolkit)⁽¹⁾은 오픈소스 계열의 화상회의 소프트웨어의 하나로 학계에서 연구 협업을 위해 사용되고 있다. 그리드 미들웨어가 바탕이 되어 개발된 액세스그리드 툴킷은 보안, 웹 서비스 기반, GUI 등 많은 면에서 우수성과 편의성을 가지고 있다.

그러나 액세스그리드 툴킷은 차세대 분산 컴퓨팅의 목표 지점이라 할 수 있는 유비쿼터스 환경에서는 적절하지 못한 요소를 구조적인 면에서 가지고 있다.

본 논문에서는 이러한 액세스그리드 툴킷의 단점을 극복하고 액세스그리드가 유비쿼터스 환경에서도 사용될 수 있게 개선된 액세스그리드 프레임워크(AccessGrid Framework)를 소개한다.

2. 액세스그리드 (AccessGrid)

액세스그리드 툴킷⁽²⁾은 미 Argonne National Laboratory의 Futures Lab.에서 개발한 협업환경이다. GSI(Grid Security Infrastructure)⁽³⁾가 제공하는 보안 프로토콜을 기반으로 한 웹 서비스 형태로 구현이 이루어졌으며, 사용 언어는 파이썬(Python)이다.⁽⁴⁾

기능적으로 가상의 협업 공간인 Venue에 참가자

(Participants)가 모여 공유된 데이터(Shared Data, Shared Application)를 가지고 화상 회의를 진행할 수 있도록 되어 있다.

액세스그리드는 다중 플랫폼을 지원하고, GSI 보안 메커니즘에 기반한 전송체제로 신뢰성 높은 보안성을 유지하며, 다중음성/화상 전송과 더불어 파일 데이터/어플리케이션 공유 등과 같은 다양한 기능들로 완성도 높은 협업 환경을 제공한다.

그러나 액세스그리드는 노드(Node)라는 사용자 혹은 단체 단위로 고정된 하드웨어 시설을 가정하고 구현되어 있다.

이렇게 사용자가 이동하는 상황을 배제한 설계는 유비쿼터스 환경에서 시스템이 지원해야 한다고 생각되는 편재성(Pervasiveness)을 제한한다. 따라서 지금의 액세스그리드는 사용자가 충분한 배경지식을 가지고 시스템을 설정해줘야 하는 기계적인 면을 벗어나지 못한다.

3. 설계중점

액세스그리드 프레임워크(AccessGrid Framework)는 그 자체로는 고정된 액세스그리드 노드 설비에 앉아있는 사용자간의 상호작용을 지원하는 틀에 지나지 않는 액세스그리드 툴킷의 능력을 확장하고, 유비쿼터스 환경에 적절한 능력을 제공하는 소프트웨어이다.

3.1 컴포넌트(Component) 기반

협업 환경을 제공하기 위하여 구성된 액세스그리드의 각 기능들은 노드간 연결성을 제공하는 프로토콜 종속성을 제외하면 서로 독립적이다. 예를 들어 파일 데이터의

공유 기능은 화상 전송 서비스와 직접적인 관계는 없다.

액세스그리드 프레임워크는 각자 독립된 기능을 제공하는 컴포넌트의 집합으로 설계되어 필요로 하는 능력만을 응용계층 프로그램이 이용할 수 있도록 지원한다. 물론 컴포넌트간에 필요할지 모르는 상호운용성도 고려되어야 한다.

3.2 개인 사용자 중심

액세스그리드 툴킷에서 Venue 내에 존재하는 ‘참가자’(Participant)는 개인일 수도 있고, 그룹 노드일 수도 있다. 데스크탑 환경에서의 액세스그리드 툴킷 사용자는 ‘Personal Node’로 불린다. 반면 대형 강의실 내에 설치된 노드에서 참가자는 다수일 수 있다(Group Node).

액세스그리드 프레임워크에서 다수의 사용자 노드는 고려대상이 아니다. 고정된 설비의 노드를 제외하면 개인별 노드는 크기가 작아 이동성이 고려되어야 하며, 움직이는 개체는 개인으로 구분되어야 하기 때문에, Venue내의 참가자는 물리적으로 개인이라고 가정한다. 이는 액세스그리드 프레임워크의 노드 이동성 지원과 밀접한 관계가 있다.

3.3 이동성 지원

액세스그리드 툴킷에서 사용자 개개인 혹은 그룹을 인식(Identify)하는 수단은 GSI에서 사용하는 인증서(Certificate)이다. 이 인증서는 사용자가 이용하는 각 노드 장비마다 하나씩 설치되어야 한다.

사용자가 사용 노드를 바꾸는 상황에서 인증서를 매번 설치하는 것은 바람직하지 않으므로, 그리드가 사용하는 인증 체계인 ‘MyProxy’ 서비스를 이용할 수도 있다. 그러나 이것도 매번 사용자의 인증이 필요하다.

액세스그리드 프레임워크는 사용자가 물리적으로 이동하는 상황에서의 자동 인증을 지원한다. 사용하는 노드의 위치가 달라지거나, 이동형 장비(Mobile devices, 보통 PDA나 휴대폰 등)에서의 사용시에 액세스그리드 프레임워크는 위치독립적으로 사용자 정보를 관리한다. 따라서 GSI에게 제시되는 인증서는 지역 노드에 저장되는 것이 아니라, 이동성 지원 기반 시스템(유비쿼터스 시스템 혹은 OGS⁽⁵⁾기반)에 의해 사용자 개인별로 관리된다.

4. 구조 및 기능

4.1 매니저(Managers)

액세스그리드 프레임워크는 크게 4가지 컴포넌트로 이루어져 있다. 이 컴포넌트들은 ‘Manager’로 불린다. 여기에는 사용자가 접속할 Venue의 주소 리스트를 기억하고 접속을 자동화하는 ‘Session Manager’, 접속한 Venue에 참가중인 참가자 정보와 공유된 데이터 및 메시지를 관리하는 ‘Collaboration Manager’, 참가자들의 영상과 대화내용의 취득 및 전송, 출력을 관리하는 ‘Multimedia Manager’, 참가자간에 열람 및 조작이 공유되는, Shared Application을 관리하는 ‘Application Manager’가 있다.

4.2 Session Manager

Session Manager는 사용자를 인식하고 사용자가 이용할 수 있는 Venue의 주소 리스트를 제공한다. 사용자가 인식되면 사용자가 접속 가능한 가상 회의 공간(Venues)을 제시하고 이미 사용하던 Venue가 존재하는 경우 자동 접속을 지원한다.

4.3 Collaboration Manager

사용자가 Venue에 접속하게 되면, 같은 Venue내의 참가자와 공유된 데이터와 응용프로그램, 다른 참가자의 화상 정보를 얻을 수 있다. Collaboration Manager는 해당 Venue의 참가자와 공유 데이터 정보를 담당한다. 데이터들은 필요한 경우 자동으로 적합한 응용 프로그램으로 로드되어 사용될 수 있다.

4.4 Multimedia Manager

액세스그리드는 화상 회의 공간을 위해서 RAT(Robust Audio Tool)과 VIC(Video Conferencing tool)⁽⁶⁾을 이용한다. 이들은 ‘Service’로서 지역 노드별로 ‘Service Manager’에 의해 관리된다.

액세스그리드 프레임워크의 Multimedia Manager는 노드별 설정이 아닌 개인별 설정으로 RAT와 VIC를 제어한다. 따라서 사용자 개인은 어디에 있는 액세스그리드 노드를 사용하더라도 개인별로 선호되는 설정의 화상회의 환경을 제공받을 수 있다.

4.5 Shared Application Manager

Venue내에서 참가자들은 많은 응용 프로그램을 공유하여 사용한다. 액세스그리드 툴킷 사용자는 이 응용 프로그램을 사용하기 위해 설치 및 실행을 노드마다 수동으로 행해야 한다.

액세스그리드 프레임워크는 사용자 중심적인 시스템이므로 사용자가 어느 위치의 노드를 쓰던 사용하던 응용 프로그램을 설치하는 등의 번거로움을 허용하지 않는다. 사용자가 노드를 사용함과 동시에 자동으로 사용하던 응용 프로그램을 탐지하고, 자동으로 설치한 뒤 실행한다.

5. 시험 구현

위와 같은 설계 구조에 따라 액세스그리드 프레임워크의 시험 구현물(Prototype)은 Java언어⁽⁷⁾를 사용하여 개발되었다. 인터프리터 언어에 가까운 파이썬(Python)으로 액세스그리드 툴킷이 개발된 점을 감안할 때, 이러한 언어의 선택은 개발상의 난제를 가져다 줄 수 있다. 이 점을 해결하기 위해 JEP(Java Embedded Python)⁽⁸⁾이라는 라이브러리가 사용되었다. JEP은 자바 언어 내에서 파이썬 인터프리터를 호출할 수 있게 해준다. 이것으로 액세스그리드 프레임워크의 구성 컴포넌트가 액세스그리드 툴킷의 클래스들의 기능들을 사용할 수 있다.

액세스그리드 프레임워크의 구성 컴포넌트들은 필요한 데이터를 저장하고 관리하는 데에 많은 부분에서 서로 공통점이 있다. 따라서 모든 컴포넌트들은 ‘Manager’라는 공통 클래스를 상속하여 구현된다. Manager 자체는

전형적인 MVC 디자인 패턴(Model-View-Controller Design Pattern, 그림 1 참조)을 따라 고안되었다.

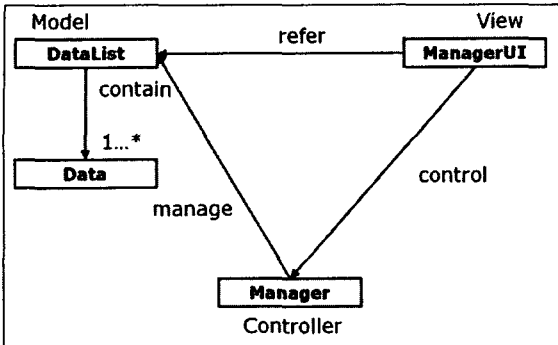


그림 1. 액세스그리드 프레임워크의 MVC디자인 패턴

5.1 Session Manager의 구현

사용자가 접속할 수 있는 Venue를 의미하는 Session은 초기 데이터를 UserConfig라는 클래스로부터 얻는다. 이것은 DataList를 상속한 SessionList에 의해 관리된다.

5.2 Collaboration Manager의 구현

Venue에 접속하게 되면 사용자는 Venue로부터 참가자와 공유된 데이터에 대한 정보를 얻을 수 있다. 각각은 ParticipantList와 SharedDataList에 의해 관리된다. 이들은 Jep 클래스를 통하여 파이썬 코드로 이루어진 ClientIW의 메소드를 호출하는 것으로 스스로의 데이터를 가져온다. ClientIW는 Jep을 중심으로 Java와 Python 양측에 존재한다. 이는 Jep 파이썬 인터프리터를 통해 필요한 데이터를 교환하도록 고안된 것이다.

5.3 Multimedia Manager의 구현

Multimedia Manager는 Audio와 Video Service의 사용자별 설정을 ServiceConfig에 저장하지만 Venue별로 어떤 Service가 활성화되어 있는지는 ServiceManagerIW 클래스를 통해 확인한다. 이 또한 액세스그리드 툴킷에 존재하는 것을 이용한다. MultiManager 클래스는 ServiceConfig로부터 사용자가 사용하던 Service를 파악하여 자동으로 Service를 활성화시킨다.

5.4 Shared Application Manager의 구현

SharedAppManager는 MultiManager와 같이 사용할 수 있는 공유 응용 프로그램의 목록을 파이썬 쪽 클래스를 통해 얻어오나, 응용 프로그램의 설치와 실행방식은 플랫폼에 따라 다를 수 있기 때문에 Delegator 클래스들에게 이 동작을 위임한다. Delegator 클래스는 액세스그리드 프레임워크의 사용 환경에 따라 플랫폼 중속적으로 구현된 자식 클래스들을 호출하여 응용 프로그램의 설치와 실행을 행한다.

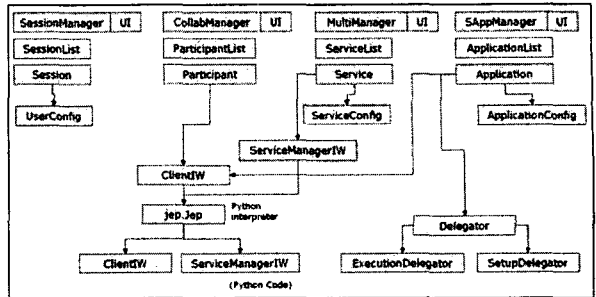


그림 2. Class Diagram

6. 결 론

위와 같은 방법으로 시험 구현된 액세스그리드 프레임워크는 액세스그리드가 구조적으로 가지고 있는 확장성의 부채를 개발 언어에 독립적으로 극복될 수 있다는 점을 보여준다. 따라서 액세스그리드가 제공하는 협업 환경은 사용자 중심의 분산 환경에서도 응용이 가능하다고 할 수 있다.

그러나 액세스그리드 프레임워크의 시험 구현물은 여전히 기존의 액세스그리드 툴킷에 매우 의존적이고, 액세스그리드를 비롯한 모든 그리드 분산 환경이 지향하는 플랫폼 독립적인 웹 서비스가 지원되지 않는다는 점은 미결로 남아있다.

액세스그리드 프레임워크는 앞으로 액세스 그리드 툴킷에 중속적인 부위를 제거하고 웹 서비스를 지원하여, 액세스그리드 호환이면서도 플랫폼 독립적이며 유비쿼터스 지향적으로 협업 환경을 지원하는 시스템으로 발전할 것이다.

7. 참고문헌/사이트

1. Rick Stevens et al., "Prototyping the Workspaces of the Future" Internet Computing, IEEE Vol. 7, Issue 4, July-Aug., p.51 - 58, 2003
2. AccessGrid Web Site <http://www.accessgrid.org/>
3. Globus Alliance Web Site <http://www.globus.org/>
4. Python programming language Homepage <http://www.python.org/>
5. OGSi Alliance Homepage <http://www.osgi.org/>
6. Mbone Conferencing Applications <http://www-mice.cs.ucl.ac.uk/multimedia/software/>
7. Sun Java Technology Web Site <http://java.sun.com/>
8. JEP (Java Embedded Python) Homepage <http://jep.sourceforge.net/>