유비쿼터스 환경에서의 구조 기반의 동적 적응형 원격 화상 회의 시스템

이상희°, 정철호, 이은석
성균관대학교 컴퓨터공학과
{neomine°, jesus98, eslee}@selab.skku.ac.kr

Dynamic Adaptive Architecture based VideoConferencing System in Ubiquitous Environment

Sanghee Lee°, Chulho Jung, Eunseok Lee
Dept. of Computer Engineering, Sungkyunkwan University

요 약

유비쿼터스 환경에서의 원격화상회의 시스템은 그 시스템을 이용하는 사용자에 따라서 시스템의 실행환경이나 사용자의 디바이스 환경 등 많은 변화 요소를 가지고 있다. 본 논문에서는, 사용자가 가질 수 있는 서로 다른 환경 등에 스스로 적응하여 효과적이고 최적화된 원격화상회의를 진행 할 수 있는 시스템 구조를 제안하며, 이를 이종의 플랫폼에서 구동할 수 있도록 각각의 플랫폼 환경에 최적화된 적응 구조를 동적으로 재구성할 수 있는 방법과 응용프로그램을 제안한다. 또한, 제안시스템의 프로토타입을 실제 구현하여 그 유효성을 확인하였다.

1. 서 론

오늘날 다양해진 네트워크 환경과 사용자 환경으로 인하여 소프트웨어는 점점 더 복잡하게 구현되었으며 다양한 환경에 적응하기 위한 기술을 필요로 해 왔다. 인터넷 기반의 원격화상회의시스템 또한 다양한 네트워크 환경과 사용자 환경에 따라서 서로 다른 구조와 구현 언어를 필요로 해 왔으며 이를 구현하기 위하여 많은 기술들이 개발 되어 왔다. Self-Adaptive Software[1] 는 이러한 필요에 의해 제안된 모델로써, 유비쿼터스 환경의 많은 응용 프로그램 개발에 사용되어 왔으며 원격화상 회의 시스템에도 적용되어 많은 연구가 진행되어 왔으며[2,5], 이 기술을 이용한 원격 화상회의 시스템도 많은 연구가 이루어 지고 있다.

하지만 기존의 대부분의 연구들은 원격 화상 회의 시스템에 Self-Adaptive 기술을 적용하는 방법 또는 온전한 적용에 초점을 맞추어 진행이 되어 왔을 뿐, Self-Adaptive 기술을 수행하는 디바이스(데스크탑 컴퓨터 또는 PDA)들의 Resource 에 대한 고려는 하지 않고 있다. 따라서, 원할한 원격화상회의를 위하여 시스템은, 참여하는 사용자의 시스템 운용 환경에 따라 영상을 수신하는 방법이나 영상의 질, 크기 등을 조절 가능해야 한다. 또한, 시스템 스스로가 회의에 참여하는 사용자에 따라서 스스로의 시스템을 적응시켜 나가는 작업을 위해 충분한 자원의 제공 여부도 시스템 스스로 파악할 수 있어야 한다.

본 논문에서는 유비쿼터스 환경에서 동작하는 원격화상회의 시스템에 Self-Adaptive 기술을 적용하여 설계하였으며, 이종의 사용자 환경을 가진 사용자간의 원활한 화상 회의를 진행 할 수 있도록 Multi-Agent 기반의 Adaptive Architecture를 설계하였다. 이에 덧붙여, 자가 적응을 위한 충분한 자원이 부족한 PDA와 같은 장치 환경에서도 효과적인 적응을 지원하도록 기존의 Adaptive Architecture를 동적으로 재구성하는 Architecture Manager를 설계하고 구현하였다.

본 논문은 다음과 같은 구성으로 되어있다. 2장에서는 관련연구를 소개하며, 3장에서는 제안시스템을 기술하였고 4장에서는 구현 및 평가, 그리고 5장에서는 결론과 향후과제로 구성하였다.

2. 관련 연구

2.1 Self-Adaptive Software

Self-Adaptive Software[3]는 프로그램 스스로 변화를 이해하고, 모니터하고 수정 가능하게 하는 기술을 가진 소프트웨어로써 프로그램이 스스로 무엇을 해야 하고, 어떻게 그것을 하고, 어떻게 그 자신의 수행 성능을 평가하고, 그래서 어떻게 상태 변화에 응답하는지에 대한 내용을 이해하고 있어야 한다. 또한, Self-Adaptive Software는 소프트웨어가 실행되고 있는 도중에 적응하기 위해 정확한 상태와 입력을 알게 되었을 때 그 자신의 행동을 수정 하고 요구사항과 입력 값, 내외부의 상태정보의 변화를 알아채는 것이 가능해야 한다[1].

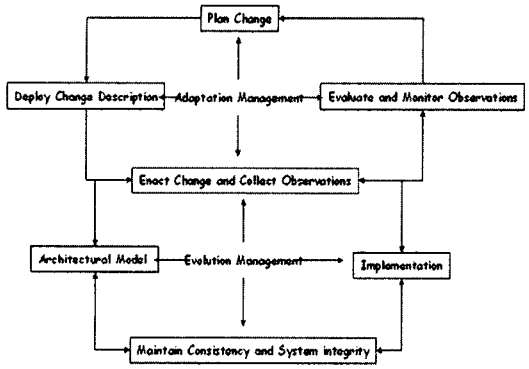
2.2. JQOS 시스템

JQOS 시스템은 세션 수신자들이 수신 받은 스트림을 그들의 개인적인 취향에 적응하는 것을 가능하게 하는 시스템으로 사용자의 QoS 요구나 시스템의 능력 그리고 Connection의 다양함이 문제를 야기한다고 판단하여 제안한 시스템이다. Self-Adaptive 요소를 사용자의 스트림 Quality에 한정하고 있으며, 그에 대한 제어를 JMF의 RTP(Real-Time Transport Protocol)에서 보내주는 RTPR을 이용하고 있어서 자가 적응의 의미가 매우 작다고 할 수 있다[2].

3. 제안 시스템

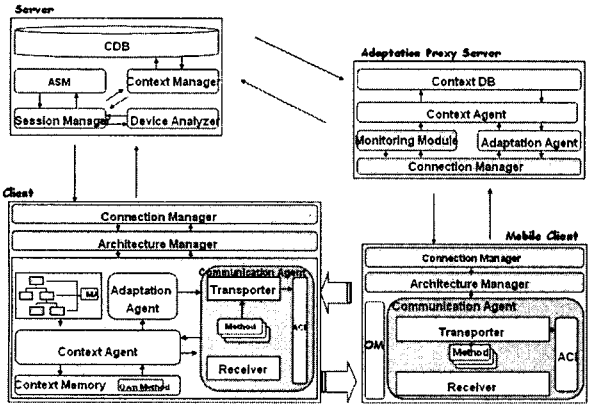
본 논문에서 제안하는 원격화상회의 시스템은 소프트웨어 구조 기반의 Self-Adaptive Software[1]를 시스템에 적용하여 설계하였다. 소프트웨어 구조 기반의 Self-Adaptive Software는 소프트웨어의 컴포넌트를 사용하여 컴포넌트의 추가, 삭제, 교체, 그리고 재설정 등

의 소프트웨어 구조를 기반으로 하여 실행중의 소프트웨어 구조를 변경하는 기능을 가능하게 한다. 소프트웨어 구조 기반의 Self-Adaptive Software의 프로세스는 그림[3]와 같다.



그림[1] 소프트웨어 구조 기반 자가 적응 시스템의 프로세스.

이 프로세스를 기반으로 하여, 우리는 시스템 스스로가 사용자의 환경에 적응하도록 하는 원격화상회의 시스템을 설계하였으며, 그 구조는 그림[2]와 같다.



그림[2] 시스템 아키텍쳐

본 논문의 원격화상회의 시스템은 다음과 같은 주요 기능을 가지고 있다.

1) 다양한 운영체제를 가진 사용자 간의 화상 통신 기능
2) 이종의 플랫폼(PC또는 PDA)을 가진 사용자 간의 화상 통신
3) 사용자의 CPU, Memory, HDD, 그리고 Network Bandwidth 등에 맞추어 화상 정보의 재구성 하는 적응 기능
4) 적응 기능을 수행하기 위한 자원이 부족한 모바일 장치 사용자를 위한 Adaptive Proxy Server 의 구성 및 Lightweight Client 구성 기능

이를 위하여 우리는 동적으로 Client의 Architecture를 재구성해주는 Architecture Manager와 모바일 사용자를 위한 적응 메카니즘을 대신 수행해 주는 Adaptive Proxy Server를 설계하였다.

3.1 시스템 구성

3.1.1 Server

서버는 원격화상회의를 위한 Client의 관리를 하는 매니저의 역할을 수행한다. 서버의 구성과 기능은 다음과 같다.

1) Context DB : 접속한 Client의 Context 저장 DB.
2) Adaptive Proxy Server Manger : Mobile Device를 위한 Adaptive Proxy Server를 관리(추가, 설정, 삭제).
3) Session Manager : 사용자 Session을 관리
4) Context Manager : Context를 관리.
5) Device Analyzer : 접속한 사용자의 장치를 분석.

3.1.2 Client

Client는 원격화상회의의 주체가 되는 사용자 모듈이다. Server측의 Device Analyzer가 접속한 client의 디바이스의 자원 및 환경을 탐색한 후, 적응에 필요한 충분한 자원이 있다고 판단될 경우 Architecture Manager에 의해서 그림[2]의 좌측 하단처럼 구성된다. 주요 기능으로는 자신의 상황과 통신에 참여한 사용자들의 상황을 고려하여 화상 정보를 재설정하고 송수신 하는 기능이 있다. Client의 구성과 기능은 다음과 같다.

1) Connection Manager: Client의 Server 접속을 관리
2) Architecture Manager: Server에 접속했을 때 자원 상황에 따라서 Architecture Manager는 Client의 Architecture를 동적으로 구성.
3) Monitoring Agent: 자신의 context를 모니터링.
4) Adaptation Agent: 자신의 context정보와 통신에 참여하는 다른 client의 context정보에 따라 화상 정보를 재설정.
5) Context Agent: 다른 client의 context와 화상 전송 Method를 Agent Communication Interface를 통하여 주고 받음.
6) Context Memory: 자신의 context와 method를 저장
7) Communication Agent: 가져온 method에 따라 다른 client와의 화상을 주고받는 agent.

3.1.3 Mobile Client

Server측 Device Analyzer의 접속한 client들의 디바이스의 자원 탐색 후, 적응에 필요한 충분한 자원이 없는 client라고 결정을 내리게 되면 Architecture Manager에 의해서 그림[2]의 우측 하단과 같이 주요한 모듈이 Lightweight하게 구성된다. Mobile Client는 Client가 가지고 있는 각 모듈들 중에서 Monitoring Agent, Adaptation Agent 그리고 Context Agent를 제외하고는 거의 동일하다.

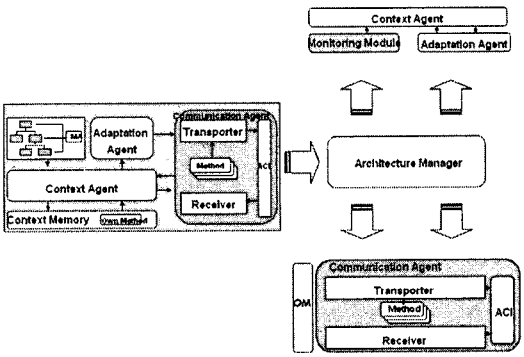
3.1.4 Adaptive Proxy Server(APS)

APS 는 Mobile Client가 하지 못하는 적응 기능을 대신 수행해 주는 일종의 proxy server이다. APS는 자원이 많은 Clinet측에 생성되며, Mobile Client가 접속하면 자동 생성된다. APS당 Mobile Client의 수는 Server측의 APSM가 담당한다. APS가 수행하는 기능은 Client 의 Monitoring Agent, Adaptation Agent, 그리고 Context Agent가 하는 기능과 동일하다.

3.2 Architecture Manager(AM)

Architecture Manager는 Client의 Architecture를 관리하는 컴포넌트이다. AM은 Client의 플랫폼 정보 및 자원 정보에 따라서 Client의 Architecture를 결정한다. 또한 원격화상회의 서버에 접속하는 사용자가 적응 메카니즘을 실행하기에 충분한 자원을 가지고 있는 않은 모바일

사용자라면, 모바일 환경에 적합한 Lightweight 한 Architecture를 구성하며, 또한 Adaptive Proxy Server를 구성하여 주요 기능을 대신하도록 해준다. 이와 같은 기능은 소프트웨어 컴포넌트의 추가 및 삭제, 교체, 재설정 등의 기능이 가능해야만 구현할 수 있는 기능이다. 이에 대한 연구는 이미 많은 곳에서 수행되어 왔으며 [4,6,7], 우리는 이들 연구중 Weaves[4]의 Reconfiguration of Component 개념을 사용하여 컴포넌트를 동적으로 재구성할 수 있도록 설계하였다.
본 논문에서 제안하는 시스템의 컴포넌트 재구성은 그림[3]과 같다.

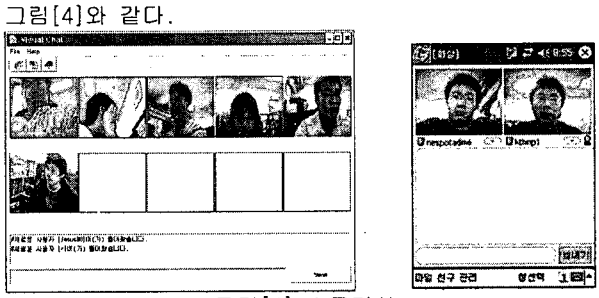


그림[3] 시스템의 컴포넌트 재구성

4. 구현 및 평가

4.1 구현

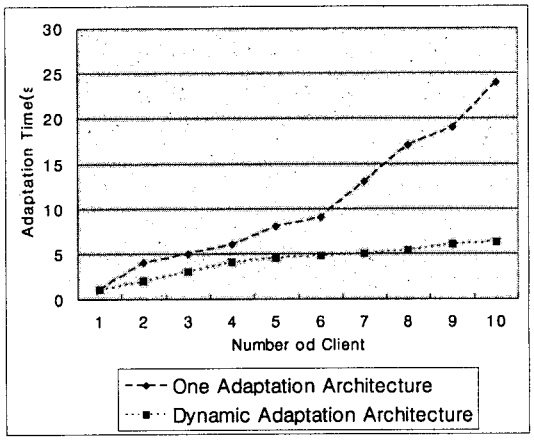
본 논문에서 제안한 원격화상회의 시스템은 Java 기반으로 설계되어 Java Media Framework(JMF)를 이용하여 구현되었다. 구현된 시스템의 스크린샷은 그림[4]와 같다.



그림[4] 스크린샷

4.2 평가

대부분의 Self-Adaptive Software는 시스템 스스로 적응 하기 위한 메카니즘을 Desktop PC에 최적화하여 설계하고 구현하였다. 본 논문에서는 이러한 점에 비추어 하나의 플랫폼에 최적화된 적응 구조를 사용한 시스템과 시스템의 자원 상황에 맞추어 동적인 적응 구조를 가진 시스템을 비교 분석 하였다. 그림[5]는 시스템 평가 그래프 이다. 그림[5]에서, 원격화상회의에 참가하는 모바일 장치를 이용하는 사용자가 점차적으로 늘어날 경우, 기존의 하나의 플랫폼에 최적화된 적응 구조를 사용한 시스템은 점점 적응 시간이 더 길어지는 것을 볼 수 있다. 반대로 동적인 적응 구조를 가진 시스템은 모바일 사용자가 늘어나더라도 적응 시간에는 큰 변화가 일어나지 않는다.



그림[5] 시스템 평가 그래프.

5. 결론 및 향후 과제

우리는 본 논문에서 사용자의 상황에 따라 화상정보를 재설정하여 원격화상회의를 수행하는 시스템을 설계, 구현 하였다. 또한 사용자의 상황에 적응하는 메카니즘을 좀 더 효과적으로 수행하도록 하기 위해 동적인 적응 구조를 설정하는 Architecture Manager를 설계하였다. 이를 통해서 이종의 사용자간의 원활한 원격화상회의를 진행할 수 있도록 하는 Self-Adaptive 원격화상회의 시스템을 구현할 수 있었다. 그러나 APSM의 효과적인 APS 생성과 관리부분에서 좀더 최적화된 알고리즘을 설계하는 것이 향후 과제이며 현재 연구중에 있다.

6. 참고문헌

[1] R. Laddaga, "Active Software",IWSAS2000,LNCS 1936,Springer-Verlag,2000,Pages:11-26,17-19 April 2000.
[2] Wenbiao Zhu, etc. al. 'JQOS: a QoS-based Internet videoconferencing system using the Java media framework (JMF)' ECE2001 Canadian ConferencePages:625 - 630 vol.1May 2001
[3] Peyman Oreizy, etc. al. "An Architecture-Based Approach to Self-Adaptive Software" IEEE Educational Activities Department, Volume 14, Pages: 54 - 62,May 1999,
[4] Michael M. Gorlick and Rami R. Razouk. The Aerospace Corporation. "Using Weaves for Software Construction and Analysis.
[5] Sung doke Lee, etc. al. " Multi-agent based Adaptive QoS Control Mechanism in -Flexible videoconferencing System. Advanced Communication Technology, 2004. The 6th International Conference on ,2 , Pages:745 - 750
[6] Quianxian Wang, etc al., CIMPSAC' 03 IEEE "Runtime Software Architecture Based Software Online Evolution"
[7] R.N. Taylor et al., "A Component- and Message-Based Architectural Style for GUI Software," IEEE Trans. Software Eng., Vol. 22, No. 6, 1996, Pages. 390-406.