

특집

유비쿼터스 서비스를 위한 롬 기반 홈 네트워크 관리 미들웨어

박준호¹⁾ 오주용²⁾ 박성호¹⁾ 강순주³⁾

목 차

- 1. 서 론
- 2. 실용적인 홈 네트워크 구축을 위한 요구 사항
- 3. 계층적 구조를 바탕으로 설계된 홈 네트워크의 개념적 모델
- 4. 롬 기반의 홈 네트워크 관리 미들웨어
- 5. Case Study: 홈 네트워크 관리 에이전트
- 6. 관련연구
- 7. 결론 및 향후 과제

1. 서 론

최근 홈 네트워크 환경은 가전기기의 단순 제어가 아닌 통합적인 관리와 상호연동을 필요로 하는 네트워크 환경으로 발전하고 있으며, 가전기기의 통신 기능 및 컴퓨팅 능력 향상으로 유비쿼터스 환경으로 변화되고 있다. 또한 사용자에 따른 서비스의 구별 및 에이전트의 학습에 따라 차별화된 서비스를 제공하는 것 또한 중요한 이슈가 되고 있다[1]. 예를 들어 사용자를 인식하고 사용자의 기호에 따라 TV의 채널을 선택한다거나 방의 온도를 조절하는 것이 가능하며 방법 시스템에서 외부인의 침입을 감지했을 경우 자동으로 위험을 외부에 알리는 서비스가 가능할 것이다. 이러한 서비스들은 가전기기들의 상호연동에 의해 가능하며 홈 네트워크를 관리하는 미들웨어에 의해 지원되어야만 할 것이다. 따라서 홈 네트워크를 관리하고 모니터링 하는 미들웨어는 가정의 모든 가전

기기 상태 정보를 관리하여 홈 네트워크 서비스를 제공하는 에이전트들이 동작할 수 있는 프레임워크를 제공해야 한다. 실질적인 홈 네트워크 구축을 위해 기존의 중앙 집중식 구조[11]나 완전 분산 관리 구조[9,10]는 홈 네트워크의 안정성과 에이전트 동작 환경에 문제점을 가진다. 중앙 집중식 관리의 경우 서버의 다운은 홈 네트워크 전체 서비스의 중단을 가져오게 되며, 완전 분산 관리의 네트워크 부하를 증가시킬뿐만 아니라 미들웨어를 기반으로 동작하는 어플리케이션 에이전트들이 디바이스의 상태 정보를 수집하고 제어하는데 어려움을 줄 수 있다. 따라서 이러한 미들웨어의 문제점을 해결하고 실질적인 홈 네트워크 환경을 구축하기 위해 본 논문은 롬 기반의 홈 네트워크 관리 미들웨어를 제안하고자 한다. 효율적인 홈 네트워크의 제어 및 모니터링을 위해 계층적 홈 네트워크 구조를 제시하고 이를 바탕으로 분산 및 중첩 자원 저장소를 설계하여 어플리케이션 에이전트들이 가전기기들을 쉽게 제어 및 모니터링 할 수 있는 롬 기반의 관리 미들웨어를 제시할 것이다. 또한 구현 예로 제안된 미들웨어를 기반으로 동작하는 홈 네트워크 관리 에이전트를 소개할 것이다.

1) 경북대학교 전자공학과 박사과정 수료
 2) 경북대학교 전자공학과 정보통신전공 박사과정
 3) 경북대학교 전자전기공학부 정보통신전공 부교수

본 논문은 다음과 같이 구성된다. 2장에서 룸 기반의 홈 네트워크 관리 미들웨어를 설계하는데 필요한 요구사항에 대해 알아보고, 3장에서 미들웨어를 설계하는데 바탕이 되는 계층적 홈 네트워크의 개념적 구조에 대해 살펴볼 것이다. 4장에서는 제안하는 룸 기반의 관리 미들웨어의 소프트웨어 구조를 상세히 설명하고, 5장에서는 미들웨어를 기반으로 동작하는 홈 네트워크 관리 에이전트 시스템을 소개할 것이다. 6장에서는 관련 연구를 설명하고 7장에서 결론을 맺도록 한다.

2. 실용적인 홈 네트워크 구축을 위한 요구 사항

홈 네트워크 환경은 일반적인 컴퓨터 네트워크 환경과 다른 특성을 가지고 있다. 따라서 실용적인 홈 네트워크 환경을 구축하기 위해 다음과 같은 특징들을 고려할 필요가 있다.

2.1 다양한 프로토콜의 지원

홈 네트워크에서 다양한 서비스를 제공하기 위해 여러 프로토콜들이 제안되고 있지만, 모든 프로토콜들이 실용화되어 사용되지는 않을 것이다. 하지만, 홈 네트워크 서비스들은 각각 다른 요구조건을 가지고 있기 때문에 하나의 프로토콜만으로 모든 서비스들을 지원할 수 없다. 따라서 몇 개의 하부 네트워크가 홈 네트워크에 공존할 것이며, 미들웨어는 어떤 프로토콜이 사용되더라도 수용할 수 있는 구조를 가져야 할 것이다. 현재 제안되고 있는 미들웨어들은 다른 프로토콜과의 연동을 고려하고 있긴 하지만, 백본 네트워크가 아닌 다른 프로토콜들은 미들웨어의 기능을 사용하기 어렵게 설계되어 있다. 따라서 홈 네트워크 관리 미들웨어는 하부의 다양한 네트워크 프로토콜을 지원할 수 있는 확장성 있는 구조를 고려하여 설계되어야 한다.

2.2 계층적 네트워크 구조

홈 네트워크의 가전기기들은 여러 프로토콜을 사용하며 서비스 수행을 위해 다른 요구조건을 가지고 있다. 예를 들어 A/V 가전기기의 경우 넓은 대역폭과 높은 QoS(Quality of Service)를 요구하지만 단순 가전기기들은 신뢰성이 있는 데이터의 전송과 실시간 전송을 요구한다. 그러나 상위 레벨의 소프트웨어 에이전트들은 가전기기들의 다양한 요구사항과 하부 프로토콜에 관계없이 서비스를 수행해야만 한다. 또한 소프트웨어 에이전트들과 가정의 사용자들은 자신의 위치나 특성에 관계없이 가정의 모든 가전기기들을 제어할 수 있어야 할 것이다. 예를 들어, "주방에서 거실의 TV를 켜라" 혹은 "안방의 모든 가전기기의 전원을 꺼라"와 같은 제어가 가능해야 된다. 이러한 요구조건을 만족하기 위해 미들웨어는 계층적인 네트워크 구조를 바탕으로 설계될 필요가 있다.

2.3 분산 및 중첩 자원 관리 구조

가정의 모든 가전기기 정보를 관리하기 위해 미들웨어는 자원 저장소를 가져야만 한다. 하지만 Jini[11]와 같이 모든 소프트웨어 에이전트들이 중앙 집중식 자원 저장소에 접근하여 가전기기 정보를 얻는다면 트래픽은 집중될 것이며 네트워크 전체를 불안하게 만들 것이다. 따라서 분산되며 독립적인 자원 관리 구조는 홈 네트워크의 신뢰성을 높여줄 것이다.

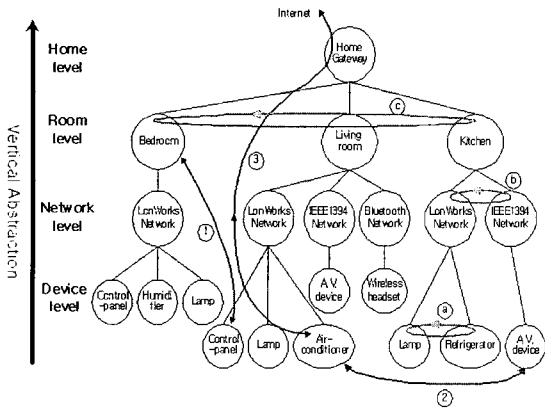
2.4 동적인 상태 정보 저장 및 변경

가정에서 가전기기들은 서로 다른 특성(사용시간, 사용빈도, 서비스 장소 등)을 가지고 있으며, 상태 정보 또한 자주 바뀐다. 따라서 홈 네트워크 관리 미들웨어는 가전기기의 상태나 위치의 변경에 관계없이 가전기기 정보의 유효성과 신뢰성을 제공해야만 한다. 이런 기능을 수행하기 위해 홈 네트워크 관리 미들웨어는 가전기기들의 등록과

삭제, 상태 정보 변경을 감시하여 소프트웨어 에이전트에게 알려줄 수 있는 구조를 가져야 할 것이다.

3. 계층적 구조를 바탕으로 설계된 홈 네트워크의 개념적 모델

2장에서 설명한 실용적인 홈 네트워크를 구현하기 위한 요구 사항을 바탕으로 하여, (그림 1)과 같은 계층적 홈 네트워크 관리 구조를 제안한다. 제안하는 계층적 관리 구조는 홈 네트워크 환경을 트리 구조의 개념적인 형태로 표현한다. 따라서 가정은 논리적으로 하부 네트워크(예: IEEE1394[6,8], LonTalk[7,8])를 가지는 몇 개의 방(예: 거실, 침실, 주방 등)단위로 구분되며, 모든 가전기기들은 상위의 객체인 네트워크 계층에 그룹으로 묶인다.



(그림 1) 계층적 홈 네트워크의 개념적 구조도

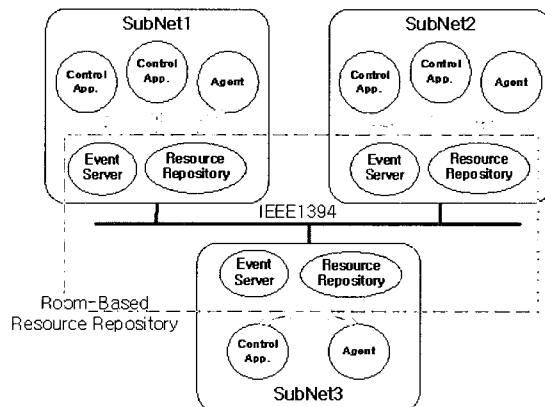
계층적 관리 구조는 수평적인 관계뿐만 아니라 수직적인 관계를 가지고 있다. 수평적인 관계(그림 1의 ㉓, ㉔, ㉕)는 같은 계층에 있는 객체들 사이에 상호 운용에 있어 다른 계층을 이용하지 않고 직접적인 통신이 가능한 관계를 의미한다. 동일한 네트워크 프로토콜을 사용하는 디바이스들은 상

위의 네트워크 계층을 통하지 않고 상호 정보를 주고받으며(그림 1의 ㉓), 네트워크 계층과 방 계층에서도 상위 계층의 영향 없이 상호 통신이 가능한 관계다(그림 1의 ㉔, ㉕). 수직적인 관계(그림 1의 ㉑, ㉒, ㉖)는 다른 계층에 있는 객체들과의 상호 운용을 의미한다. 예를 들어 위치에 관계 없는 디바이스 사이의 통신(그림 1의 ㉒)이나, 다른 계층간의 통신(그림 1의 ㉑)을 하거나 인터넷을 통해 디바이스를 제어할 수 있는 관계를 의미한다(그림 1의 ㉖).

제안한 구조에서, 상위 계층의 객체는 하위 계층의 객체들을 추상화하며, 같은 계층에 있는 각 객체들은 상호 독립적으로 동작한다. 계층적 홈 네트워크 관리 구조는 네트워크 구성과 동작의 복잡성을 줄이고 그룹 단위의 제어를 가능하게 하며, 다른 서브 네트워크에 있는 객체에 영향을 주지 않고 하나의 객체를 추가하거나 제거하는 것을 용이하게 한다.

4. 룸 기반의 홈 네트워크 관리 미들웨어

4.1 분산 및 중첩 자원 관리 구조



(그림 2) 룸 단위의 홈 네트워크 자원 저장소 모델

다중 에이전트 환경을 제공하기 위해 원격에서

제어하는 에이전트와 제어되는 가전기기 사이의 약한 연결 관계(loosely-coupled connection)는 중요한 이슈가 되고 있다[2,3]. 또한 모든 에이전트가 중앙집중식 자원 저장소에서 가전기기 정보를 가지고 온다면 트래픽은 집중될 것이며 네트워크는 신뢰성을 보장할 수 없게 된다. 결과적으로 위의 상황을 해결하고 더욱 신뢰성 있는 동작을 보장하기 위해, 본 논문에서는 (그림 2)와 같은 룸 기반의 분산 및 중첩 자원 저장소 모델을 제시한다. (그림 2)와 같이 각 룸 단위의 서브 네트워크는 로컬 자원 저장소를 가지고 있어 로컬 가전기기의 상태 정보를 저장하고 관리한다. 룸 단위로 분산된 자원 저장소는 자신이 관리하는 방의 가전기기 정보만 가지는 것이 아니라 가정의 모든 가전기기 정보를 가지고 있다. 가전기기의 상태 정보 변경은 로컬 자원 저장소에 저장되며 원격의 자원 저장소에 전달되어 모든 자원 저장소는 동일한 가전기기 상태 정보를 가지게 된다. 또한 소프트웨어 에이전트들 사이의 제어 및 상태 변경 이벤트는 이벤트 서버에 의해 상호 전달된다. 제안하는 룸 기반의 홈 네트워크 관리 미들웨어는 분산 및 중첩 자원 저장소와 이벤트 서버를 이용하여 홈 네트워크 전체를 하나의 자원 저장소로 보여준다. 즉, 모든 가전기기 정보는 룸 단위의 자원 저장소에서 동일하게 저장되어 사용자와 소프트웨어 에이전트가 어느 위치에 있든지 동일한 정보를 얻을 수 있게 하며 다른 방에 있는 가전기기를 로컬 가전기기를 제어하는 것처럼 사용할 수 있게 한다. 또한 룸 기반으로 가전기기 정보들이 관리되기 때문에 룸 단위의 제어와 같은 그룹 단위의 제어를 더욱 용이하게 한다.

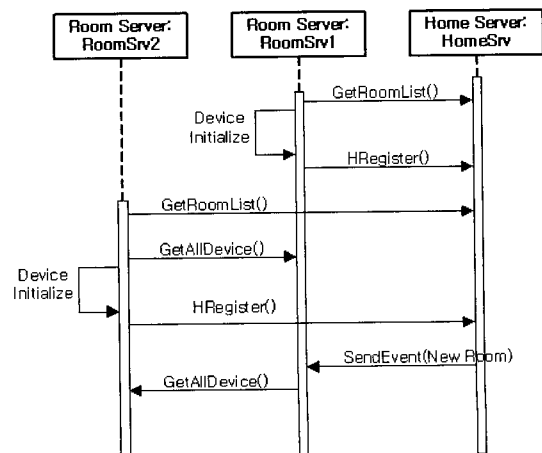
4.2 룸 기반의 관리 미들웨어 사이의 통신

제안하는 홈 네트워크 관리 미들웨어는 크게 홈 서버, 룸 서버와 에이전트로 구성된다. 제안하는 미들웨어는 계층적 구조를 바탕으로 구현되어 홈

서버는 하위의 룸 서버 상태를 감시하며 룸 서버는 가전기기들의 상태를 감시하고 관리한다. 홈 서버 및 룸 서버의 서비스는 크게 자원 저장소와 이벤트 서비스로 구성된다. 가전기기 제어 어플리케이션 및 소프트웨어 에이전트는 자원 저장소 및 이벤트 서비스를 통해 가전기기의 상태 정보를 저장하고 감시하며 가전기기를 제어하게 된다. 본 절에서는 홈 네트워크를 관리하기 위한 룸 기반의 미들웨어 서비스와 에이전트 사이의 상호 연결 및 통신에 대해 살펴 볼 것이다.

4.2.1 홈 네트워크에 새로운 룸 서버의 추가

홈 서버는 룸 서버의 정보를 관리하고 감시하는 역할을 한다. 홈 서버는 내부에 하부 객체인 룸 서버 정보를 관리하기 위해 룸 서버 정보(룸 이름, 위치 정보)를 가지고 있다. 홈 서버는 룸 서버의 작동 유무를 감시하여 룸 서버의 상태 변화를 다른 룸 서버들에게 이벤트 서비스로 알려준다. 하위 객체인 룸 서버들은 홈 네트워크가 초기화될 때 홈 서버를 통해 다른 룸 서버들의 위치 정보를 얻을 수 있으며 룸 서버 위치 정보를 이용하여 룸 단위로 분산된 자원 저장소 및 이벤트 서버가 상호연용 된다.



(그림 3) 룸 서버 및 홈 네트워크 초기화 과정

(그림 3)은 룸 서버가 홈 네트워크에 새로 추가되거나 홈 네트워크가 초기화되는 과정을 보여준다. 새로운 룸 서버가 홈 네트워크에 추가된다면, 홈 서버로부터 다른 룸 서버 상태 정보를 얻어온다. 만약 다른 룸 서버가 없다면 하부 네트워크를 초기화하고 가전기기의 등록을 기다리게 된다. 반면 다른 룸 서버가 존재한다면, 홈 서버로부터 받은 위치 정보를 이용하여 원격의 룸 서버가 관리하는 가전기기 정보를 얻어와 로컬 자원 저장소에 저장하여 룸 단위의 분산 자원 저장소의 가전기기 정보를 동기화 시킨다.

4.2.2 새로운 가전기기의 추가

룸 서버는 홈 네트워크 가전기기가 동작할 수 있는 환경을 제공하고, 에이전트들이 가전기기를 감시, 관리할 수 있는 환경을 제공한다. 이를 위해 룸 서버는 룸 단위로 분산된 자원 저장소의 가전기기 정보를 동기화 시키고, 가전기기 제어 어플리케이션과 에이전트가 생성하는 이벤트를 전달하기 위한 상호 연동 구조를 가지고 있다. 룸 서버는 디바이스 제어를 위한 디바이스 에이전트 인터페이스, 홈 네트워크의 상태 정보를 감시하는 에이전트와의 인터페이스 및 룸 서버들 간의 인터페이스를 정의하고 있다. 아래의 코드는 CORBA IDL로 정의된 인터페이스를 보여준다.

```
interface DeviceManagement {
    // Device Agent Interface
    short DUpdateDuration (in short duration, in DIS dis);
    oneway void DRegister (in DIS dis);
    oneway void DUpdate (in DIS dis);
    oneway void DDelete (in DIS dis);

    // Agent Interface
    DeviceSeq GetAllDeviceInfo (in short RoomNum);
    DIS GetDeviceInfo (in short RoomNum, in string DeviceName);
    short GetRoomDeviceNum (in short RoomNum);
    short GetRoomNum ();

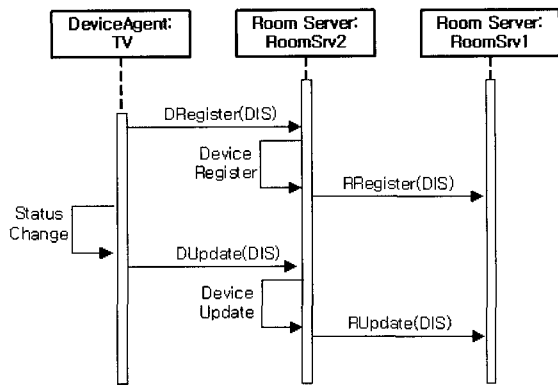
    // Event Interface
    void RegisterListener (in string IOR);
    void ReleaseListener (in string IOR);
    oneway void Push (in char cls, in char cmd, in DIS dis);
};
```

또한 룸 서버는 내부에 모든 가전기기의 상태 정보를 저장하기 위해 링크드 리스트 형태의 데이터베이스를 가지고 있다. 여기에 저장되는 상태 정보는 아래 코드와 같다.

```
struct Binding {
    short RoomNum;
    string DeviceName;
    short BindingTime;
    short priority;
};
struct Position {
    short RoomNum;
    string DeviceName;
    string IOR;
};
struct Service {
    string Servicetype;
    string Status;
};
struct Access {
    Binding binding;
    short AccessLevel;
};
struct DIS {
    Position position;
    short ServiceSize;
    Service service [15];
    Access access;
};
typedef sequence<DIS> DeviceSeq;
```

분산된 룸 서버들 사이의 동기화는 두 가지 방법으로 이루어진다. 첫번째로 새로운 룸 서버가 추가될 경우 추가된 룸 서버는 원격의 룸 서버들을 RMI(Remote Method Invocation)를 통해 각 룸 서버가 관리하는 가전기기 정보를 얻어온다. 이때 전달되는 정보는 CORBA IDL에 정의된 Sequence 타입으로 얻어오게 된다. 다른 동기화 방법은 가전기기가 추가되거나 상태정보가 변경되는 경우 이벤트 서비스를 통해 전달된다.

(그림 4)는 새로운 가전기기가 홈 네트워크에 추가될 경우 룸 서버와 디바이스간의 상호 연동 과정을 보여준다. (그림 4)에서처럼 TV 디바이스가 홈 네트워크에 추가되었을 경우, 먼저 로컬 룸 서버에게 자신의 정보를 등록한다. 로컬 룸 서버는 이를 다른 룸 서버에게 전달하여 디바이스 상태 정보를 동기화한다.



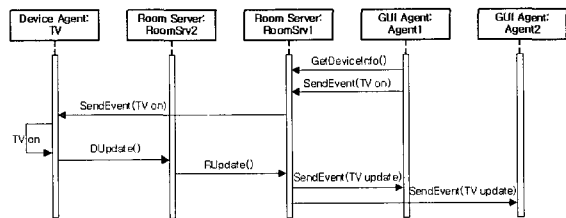
(그림 4) 새로운 가전기기의 추가

4.2.3 에이전트의 디바이스 제어

제어 명령은 로컬에서 디바이스를 실제로 제어하고 있는 디바이스 제어 어플리케이션에게 이벤트로 전달된다. (그림 5)에서처럼 제어 이벤트는 디바이스를 원격에서 제어하는 GUI 에이전트에 의해 생성되어 룸 서버로 전송된다. (그림 5)에서 Agent1은 RoomSrv1에 연결되어 있으며 RoomSrv2에 연결된 TV 디바이스를 제어하기 위해 이벤트를 전송하였다. RoomSrv1은 제어 이벤트를 TV 디바이스 에이전트에 전송하고, 제어 결과는 RoomSrv2의 자원 저장소에 저장된다. RoomSrv2의 정보 변경은 앞서 설명한 동기화 과정을 통해 RoomSrv1에 전달된다. RoomSrv1은 자신에게 연결된 Agent1과 Agent2에 TV 디바이스의 상태가 변경되었다는 것을 알려 디바이스 제어 과정이 일어나게 된다. (그림 5)에서처럼 디바이스 및 GUI 에이전트의 위치에 관계없이 제어 과정이 일어나게 되며, 따라서 GUI 에이전트들은 디바이스를 제어하는데 있어 위치의 투명성을 보장받게 된다.

또한, 에이전트들은 “거실의 모든 가전기기 전원을 꺼라”와 같은 방 단위의 제어나 “가정의 모든 전등의 전원을 켜라”와 같은 가전기기 종류 별 제어 명령도 생성할 수 있다. 에이전트들에 의해 생

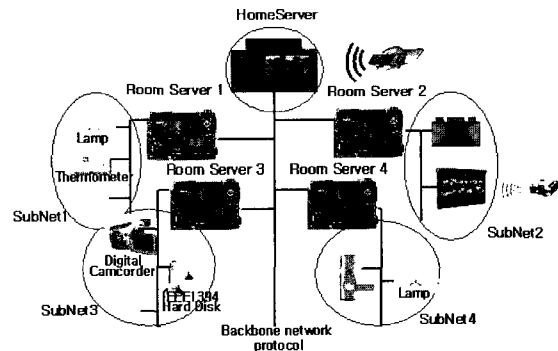
성되는 그룹 단위 명령은 로컬 룸 서버의 이벤트 서버에 의해 해석되어 적합한 디바이스들에게 전달된다. 룸 단위의 계층적 구조를 바탕으로 분산 자원 저장소 및 이벤트 서버 모델이 설계되었기 때문에 그룹 단위의 제어가 가능하다. 그룹 단위 제어는 다음 장에서 자세히 살펴볼 것이다.



(그림 5) 디바이스 제어 흐름도

5. Case Study: 홈 네트워크 관리 에이전트

5.1 하드웨어 환경



(그림 6) 홈 네트워크 하드웨어 구성도

제안한 홈 네트워크 관리 미들웨어와 홈 네트워크 관리 에이전트를 동작시키기 위해 (그림 6)과 같이 홈 네트워크 환경을 구축하였다. 각각의 룸 서버들은 백본 네트워크인 IEEE1394를 이용하여 연결되었으며, 서브 네트워크 프로토콜로 LonTalk을 지원하고 있다. 각 룸 서버들은 X86 CPU 기반의 SBC(Single Board Computers)

를 이용하였으며, Linux 상에서 구현되었다. 제안하는 홈 네트워크 관리 미들웨어는 현재 GNOME 프로젝트로 진행되고 있는 ORBit CORBA ORB(-v:ORBit-0.5.18)버전을 기반으로 하여 IEEE1394 프로토콜을 지원할 수 있도록 수정하여 사용하였다[5]. IEEE1394를 위한 ORB의 구현을 위해 RTLinux[12] 상에서 디바이스 드라이버를 구현하였다. A/V 디바이스로는 IEEE1394 프로토콜을 사용하는 Sony사의 디지털 캠코더와 빔 프로젝터를 사용하였으며, 소형가전기기로 LonWorks 프로토콜을 사용하는 전등, 커튼, 가스 센서 및 리모콘 센서를 이용하였다.

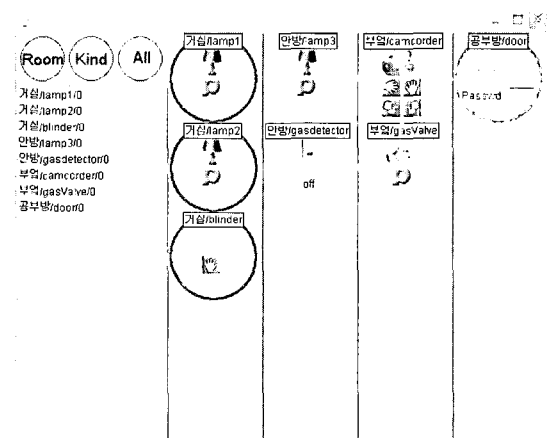
5.2 홈 네트워크 관리 에이전트 설계

홈 네트워크는 일반적인 컴퓨터 네트워크와 다른 복잡한 네트워크이며 홈 네트워크의 사용자는 전문가가 아닌 일반 사용자다. 따라서 가전기기들을 제어하는 내부 기능들은 사용자에게 감춰지고 자동적으로 수행되어 사용자에게 쉽고 편리한 인터페이스를 제공해야 한다. 홈 네트워크 관리 미들웨어는 동적으로 가전기기 정보를 재구성하여 실시간으로 가전기기 상태 변화를 감시 및 제공할 수 있어야 하며, 사용자 인터페이스는 원격지에 자동으로 다운로드 되어 사용될 수 있어야 한다.

본 장에서는 제안된 룸 단위의 관리 미들웨어 상에서 구현된 웹 기반의 관리 에이전트를 설명할 것이다. 관리 에이전트는 Java를 이용하여 구현되었으며 GUI(Graphical User Interface) 아이콘을 사용하여 원격지에서 가전기기들을 제어하고 모니터링할 수 있도록 구현되었다. 에이전트는 제안된 룸 단위 미들웨어에 접근하여 자원 저장소에서 가정의 모든 가전기기 정보를 얻을 수 있으며 이벤트 서비스를 이용하여 가전기기를 제어할 수 있다.

5.3 홈 네트워크 관리 에이전트

룸 단위의 홈 네트워크 미들웨어를 사용하는 관리 에이전트는 사용자의 편의를 위하여 웹 기반의 GUI를 이용하여 구현되었으며 (그림 6)의 하드웨어 환경에서 구동되었다. (그림 6)의 하드웨어 환경에서 룸 기반의 관리 미들웨어를 동작시켰으며 이를 기반으로 구현된 관리 에이전트는 룸 서버에 연결되어 동작한다. 홈 네트워크 관리 에이전트는 (그림 7)과 같이 디바이스 제어 방법을 선택할 수 있는 프레임과 디바이스 상태를 확인할 수 있는 프레임으로 구성된다.

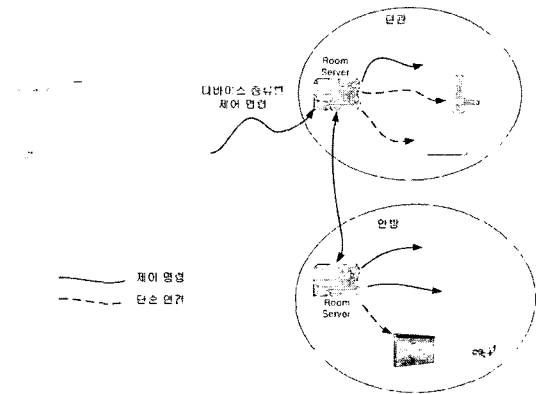


(그림 7) GUI 기반의 분산 제어 및 통합 관리 시스템

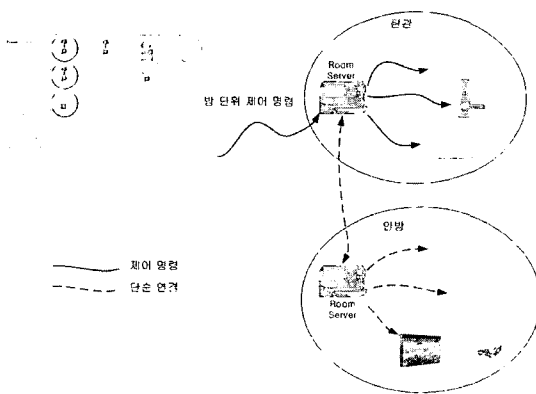
(그림 7)에 나타난 것처럼 가전기기들은 하나의 큰 원으로 표시되며 디바이스의 상태 정보 및 제어 가능한 기능들은 아이콘으로 표시된다. (그림 7)에서 왼쪽 프레임은 디바이스를 룸 단위, 종류별, 전체로 나타내는 기능을 가지고 있으며 이를 이용하여 룸 단위나 종류별 가전기기 제어가 가능하다.

홈 네트워크 환경에서 디바이스 종류에 따라 제어하기 위해 (그림 7)의 "Kind" 버튼을 이용한다. (그림 8)의 GUI 관리 에이전트에서 그룹 제어 이벤트가 연결된 룸 서버에 전달되면 미들웨어는 자

원 저장소에서 같은 종류의 제어 디바이스 목록을 찾아 각 가전기기에 전달한다. 에이전트에 의해 생성된 이벤트는 다른 룸 서버에도 전달되어 가전기기를 제어하게 된다. 또한 (그림 7)의 "Room" 버튼을 이용한다면 방 단위의 제어가 가능하다. (그림 9)에서처럼 GUI 관리 에이전트에서 제어 이벤트가 룸 서버에 전달되면 모든 디바이스에 제어 명령을 전달하여 방 단위의 제어를 하게 된다.



(그림 8) 홈 네트워크 관리 시스템에서의 디바이스 종류별 제어



(그림 9) 홈 네트워크 관리 시스템에서의 방 단위 제어

6. 관련 연구

홈 네트워크를 위한 많은 연구들이 이미 진행되고 있다. HAVi(9)는 IEEE1394를 기반으로 하여 A/V 서비스를 제공하기 위해 제안된 미들웨어

다. 하지만 디바이스의 위치에 따른 그룹 제어나 다른 프로토콜들 사이의 상호 운용을 지원하는 것은 고려하지 않고 있다. UPnP(10)의 경우 마이크로 소프트웨어에 의해 제안된 홈과 오피스 네트워크를 위한 미들웨어다. 하지만 IP 기반의 네트워크만을 지원하고 있으며 앞서 설명한 바와 같이 디바이스 자체의 높은 컴퓨팅 파워를 요구한다. Jini(11) 또한 홈과 오피스 네트워크를 위해 제안된 미들웨어로서 디바이스를 네트워크 상에서 찾는 룩업(look-up), 디스커버리(discovery) 서비스를 가지고 있다. 하지만 앞서 설명한바와 같이 중앙 집중식 자원 관리라는 문제점을 가지고 있다. 다른 관련 연구 활동으로 Stanford 대학에서 개발된 iROOM(4)은 가정이나 빌딩에서 사용자와 디바이스 사이의 상호 작용에 중점을 두고 있다. iROOM은 TSpace(3)를 기반으로 디바이스와 사용자 사이의 상호 작용을 위해 Event Heap을 사용하고 있다. iROOM은 Space를 이용한 자원 저장소 개념을 사용하고 있지만 홈 네트워크의 제어와 감시를 위해 제안된 본 논문과는 다른 관점을 가지고 연구되었다. Essex 대학의 IIE (Intelligent Inhabited Environment) group(1)은 방을 빌딩 제어를 위한 기본 단위로 제안하고 있지만 인공지능 에이전트 사이의 통신 프로토콜과 사용자에 따른 경향 분석에 중점을 두고 있다.

7. 결론 및 향후 과제

본 논문에서는 계층적 관리 구조를 바탕으로 설계된 룸 단위의 홈 네트워크 관리 미들웨어를 제안하고 사례 연구로 미들웨어를 기반으로 동작하는 GUI 기반의 관리 에이전트를 구현하였다. 홈 네트워크 관리 미들웨어는 룸 단위로 분산된 자원 저장소와 이벤트 서버를 통해 홈 네트워크의 모든 가전기기들의 상태 정보를 관리할 수 있다. 분산

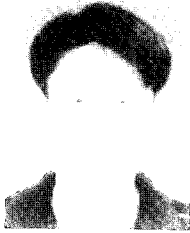
된 자원 저장소는 모든 가전기기 상태 정보를 중첩 저장하여 가전기기과 에이전트 사이에 위치의 투명성을 제공하고 홈 네트워크가 좀더 신뢰성 있고 안정적으로 동작하도록 하였다. 또한 이벤트 서비스를 통해 상호 통신이 가능하도록 하였다. 제안한 미들웨어는 테스트 환경을 구축하여 구동하였으며 관리 에이전트를 통해 다양한 가전기기의 제어를 수행하였다. GUI 기반의 관리 에이전트는 가전기기 각각의 제어뿐만 아니라 방 단위의 제어 및 가전기기 종류별 제어가 가능하도록 구현되었다.

제안된 미들웨어는 계층적 구조를 바탕으로 설계되어 신뢰성 있는 홈 네트워크 관리를 제공하며 룸 단위의 자원 저장소를 통해 다양한 제어 방법을 제공한다. 향후 과제로서 IEEE1394의 등시성 전송을 이용한 미들웨어 구조 및 구현된 미들웨어 상에서 동작하는 다양한 인공지능 에이전트를 구현할 것이다.

참고문헌

- [1] V. Callaghan, G. Clarke, A. Pounds-Cornish and S. Sharples, "Buildings as Intelligent Autonomous Systems: A Model for Integrating Personal and Building Agents," the 6th International Conference on Intelligent Autonomous Systems, Venice, July, 2000
- [2] N. Carriero and D. Gelernter, "Linda in Context," Comm. ACM, vol. 32, no. 4, Apr. 1989.
- [3] P. Wyckoff, S. McLaughry, T. Lehman, D. Ford, "TSpaces," IBM Systems Journal, Volume 37, Number 3 August, 1998, pp 454-474
- [4] Brad Johanson and Armando Fox, "The Event Heap: A Coordination Infrastructure for Interactive Workspaces," 4th IEEE Workshop on Mobile Computing Systems & Applications, June 200
- [5] J. Y. Oh, J. H. Park, G. H. Jung and S. J. Kang, "CORBA based Core Middleware Architecture Supporting Seamless Interoperability between Standard Home Network Middlewares," ICCE, June 2003
- [6] IEEE Std. 1394-1995, IEEE Std. for a High Performance Serial Bus, IEEE, 1995.
- [7] LonTalk Protocol Specification Version 3.0, Echelon, 1994.
- [8] Gerard O'Dricoll, Essential Guide to Home Networking Technologies, Prentice Hall PTR, 2000.
- [9] Specification of the Home Audio/Video Interoperability (HAVi) Architecture Version 1.0, 18 Jan. 2000
- [10] Universal Plug and Play: Background. <http://www.upnp.com/resources/UPnPbkgnnd.htm>, 1999
- [11] Jini Architecture Specification Revision 1.0, Sun microsystems, Jan. 1999
- [12] <http://www.fsmlabs.com>

저자약력



박준호

1998년 경북대학교 전자공학과 (공학사)
2000년 경북대학교 전자공학과 정보통신전공 (공학석사)
2002년 경북대학교 전자공학과 박사과정 수료
관심분야 : 실시간 시스템, 홈 네트워크, 미들웨어
이 메 일 : zec@palgong.knu.ac.kr



박성호

1999년 경북대학교 전자공학과 (공학사)
2001년 경북대학교 정보통신학과 정보통신전공 (공학석사)
2003년 경북대학교 전자공학과 (박사수료)
관심분야 : 실시간 시스템, 소프트웨어 공학, 홈 네트워크,
미들웨어, LonTalk, IEEE1394
이 메 일 : slblue@origio.net



오주용

2000년 경북대학교 전자공학과 (공학사)
2002년 경북대학교 전자공학과 정보통신전공 (공학석사)
2003년 경북대학교 전자공학과 정보통신전공 (박사과정)
관심분야 : 실시간 시스템, 홈 네트워크, 미들웨어,
IEEE1394
이 메 일 : anyong@palgong.knu.ac.kr



강순주

1983년 경북대학교 전자공학과 (공학사)
1985년 한국과학기술원 전자계산학과 (공학석사)
1995년 한국과학기술원 전자계산학과 (공학박사)
1985년~1996년 한국원자력연구소, 핵인공지능연구실 선임연구
원(과책), 전산정보실 선임연구원(실장)
1996년~1998년 경북대학교 전자전기공학부 정보통신전공
전임강사
2000년~2001년 University of Pennsylvania, Dept. of
CIS, 방문연구교수
1998년 -현재 경북대학교 전자전기공학부 정보통신전공 부교수
관심분야 : 실시간 시스템, 임베디드 시스템, 지식기반시스템
이 메 일 : sjkang@ee.knu.ac.kr