

미들웨어와 효율적인 통신을 위한 컨텍스트 응용의 구현

김동일*, 임원택*, 정국상**, 최덕재*

*전남대학교 전산학과

**한국 전자 통신 연구원

e-mail:hangaram1@nate.com, handeum@daum.net, dchoi@chonnam.ac.kr

Implementation of A Efficient Communication Based on Middleware to Connect Heterogeneous Context Application

Dong-il Kim*, Kug-Sang Jeong**, Deokjai Choi*

*Dept of Computer Science, Chonnam University

**Electronics and Telecommunications Research Institute

요 약

현재 제안된 프레임워크들은 컨텍스트 응용이 미들웨어에 너무 의존적인 형태로 인하여 컨텍스트 응용의 확장성을 저하시켜 적용범위를 제한한다. 또한 컨텍스트 응용의 이동을 어렵게 하여 서비스를 제한적으로 제공하는 문제점이 나타난다. 이러한 문제점을 해결하기 위해 미들웨어와 컨텍스트 응용의 독립을 제안한 프레임워크를 사용한다. 하지만, 독립 프레임워크에서는 미들웨어와 컨텍스트 응용간 통신에 JMS(Java Message Service)를 사용함으로써 메시지 전송의 유실 및 여러 컨텍스트 응용 연결로 인한 처리속도 저하의 문제점이 발생 한다. 이러한 통신상의 문제를 해결하기 위해서 socket통신을 사용한 컨텍스트 응용을 제안한다.

1. 서론

유비쿼터스 환경¹⁾이 구축된 상황에서는 사용자가 주변 환경을 파악할 수 있는 컨텍스트 정보와 서비스를 요청하지 않아도 제공 받을 수 있다. 이는 유비쿼터스 환경의 발달로 인해 사용자의 동적인 환경 변화를 인식하고 이에 적용할 수 있는 서비스를 제공하는 것을 의미한다. 즉 사용자의 상황인식을 통하여 적절한 서비스를 제공하는 것이다. 이러한 적절한 서비스를 제공하기 위해서는 유비쿼터스 컴퓨팅 인프라가 필요하다. 이러한 인프라는 일반적으로 사용자의 주변환경의 변화 정보를 제공해주는 센서 부분과, 센서로부터 받은 정보를 통하여 이를 추론하여 컨텍스트 정보를 만들어내는 미들웨어, 마지막으로 미들웨어로부터 추론되어 넘어온 컨텍스트 정보를 활용하여 가전기 및 응용분야를 제어하는 응용으로 나뉜다[1].

현재 유비쿼터스 인프라를 구성하기 위해 Gaia(Context Infrastructure), SOCAM(Service-Oriented Context-aware Middleware), CoBra(Context Broker Architecture)와 같은 다양한 프레임워크가 제안되어 있다. 그러나 현재 제안된 프레임워크는 컨텍스트 응용개발이 미들웨어에 의존적인 형태를 지니고 있다. 이러한 컨텍스트 응용개발이 미들웨어에 의존적인 형태는 컨텍스트 응

용개발자가 응용을 쉽게 개발할 수 없다는 문제점을 가지고 있다. 이는 컨텍스트 인지 응용이 미들웨어에 종속되어 미들웨어의 정보표현, 정보전달, 컨텍스트 인지 등의 방법을 알아야만 컨텍스트 응용을 개발 할 수 있다는 것을 나타낸다. 이러한 문제는 컨텍스트 응용의 확장성을 저하시켜 적용범위를 제한 할 뿐만 아니라 컨텍스트 응용의 이동을 어렵게 하여 서비스를 제한적으로 제공하는 문제점이 나타난다.

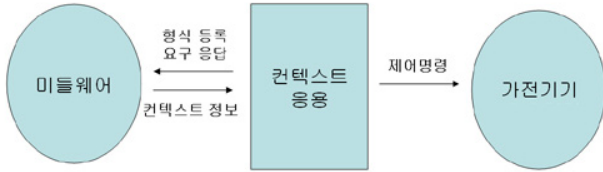
이러한 문제점을 해결하기 위해 UTOPIA[2]에서는 미들웨어와 컨텍스트 응용부분을 독립적으로 설계하여 컨텍스트 응용이 미들웨어에 의존적인 형태에서 벗어나 컨텍스트 응용의 확장성을 높였다. 하지만, UOPIA에서는 미들웨어와 컨텍스트 응용의 독립을 위하여 설계되었지만, 미들웨어와 컨텍스트 응용의 통신에 JMS를 사용하여 메시지 전송의 유실 및 여러 컨텍스트 응용 연결로 인한 처리속도 저하의 문제점이 발생 한다. 이러한 문제점을 해결하기 위해서 이 논문에서는 socket통신을 이용한 컨텍스트 응용을 제안한다.

2. 관련 연구

가. UTOPIA에서 컨텍스트 응용 독립

UTOPIA에서 컨텍스트 응용은 미들웨어로부터 받아진 데이터를 바탕으로 가전기기를 직접 제어하는 부분으로 미들웨어와 가전기기의 중간에 위치하고 있다.

*본 연구는 지역산업진흥사업의 결과로 수행되었습니다.



(그림 1) 미들웨어와 컨텍스트 응용 관계

위의 그림은 컨텍스트 응용이 미들웨어와 가전기기의 중간에 위치하여 미들웨어로부터 컨텍스트 정보를 받고, 가전기기를 제어하는 구성을 나타낸다. 컨텍스트 응용은 처음 미들웨어와 통신을 위해 접속할 때 고유의 등록형식을 미들웨어에 등록 요청한다. 미들웨어는 컨텍스트 응용이 보낸 형식을 등록하고, 컨텍스트 정보를 등록된 형식에 맞추어 컨텍스트 응용에게 전송한다. 이렇게 받은 컨텍스트 정보를 컨텍스트 응용이 분석하여 가전기기에 제어명령문을 전송한다.

나. Socket

소켓은 소프트웨어로 작성된 통신 접속점이라고 할 수 있는데 네트워크 응용 프로그램은 소켓을 통하여 통신망으로 IP 패킷을 송수신하게 된다[3]. 자바에서는 유닉스, 윈도우에서 각각 유닉스 소켓(socket)과 윈도우 소켓(WinSocket)을 사용할 수 있는 소켓 프로그래밍 인터페이스를 지원한다. 자바에서 소켓 프로그래밍을 사용하기 위해서는 java.net 클래스 사용해야 한다. 소켓 통신에는 TCP(Transmission Control Protocol)통신인 연결형과 UDP(User Datagram Protocol)통신인 비연결형으로 나뉜다. TCP통신은 신뢰성 있는 데이터 전송에 주로 사용되며 Socket과 ServerSocket 클래스가 사용된다. UDP통신은 비신뢰적인 데이터 전송에 주로 사용되며 DatagramSocket과 DatagramPacket 클래스가 사용된다.

다. JMS(Java Message Service)[3]

JMS(Java Message Service)는 자바로 개발된 MOM(Message-Oriented Middleware) 벤더와 DB개발 업체 그리고 인터넷 관련 업체가 메시징 기반 제품을 사용하는데 접근하기 위한 공통의 자바 API를 제공하기 위해 만들어 졌다. MOM(Message Oriented Middleware)이란 메시지 기반 미들웨어로서 메시징시스템을 비즈니스 시스템에 적용했을때 엔터프라이즈 메시지 시스템 또는 메시지 기반 미들웨어라고 한다.

메시징 제품은 Point-to-Point(이하 PTP) 또는 Publish/Subscribe(이하 pub/sub)으로 구분하며 PTP는 메시지 큐(Queue)에 관한 개념이며 각각의 메시지가 큐로 전달하면 메시지를 수신하는 클라이언트가 큐로부터 메시지를 수신하는 동작원리를 가지고 있다. Pub/Sub는 다수의 메시지 생성자가 토픽(Topic)으로 전달하여 다수의

메시지 소비자에게 메시지를 전달하는 메시지 모델이다. JMS는 인터페이스를 구현하고 관리와 제어 기능을 제공하는 메시징 시스템을 JMS 제공자(Provider)라고 한다. 메시지 클라이언트는 JMS 클라이언트와 비 JMS 클라이언트로 나눌수 있다. JMS 클라이언트는 자바 언어로 작성하고, JMS API를 사용하여 메시지를 보내고 받는 프로그램이다. 프로그래밍 언어에 상관없이 다른 프로그램은 비-JMS 클라이언트이다. 두 종류의 클라이언트는 서로 통신할 수 있다.

JMS가 사용하고 있는 관리객체와 인터페이스 개발순서를 보면 관리객체는 시스템 관리자가 생성하는 객체로서 JMS Configuration Information을 포함하는 객체를 말하고 이 객체는 클라이언트가 사용하는데 관리자는 이 객체를 네이밍 서비스의 네임스페이스(JND)내에 둔다.

3. 컨텍스트 응용 구현

컨텍스트 응용은 세 개의 모듈로 구성되어 있다. 미들웨어와 통신을 담당하는 미들웨어 통신모듈, 통신을 통하여 들어온 메시지를 파싱하여 명령문을 구성하는 파싱모듈, 그리고 마지막으로 파싱되어 구성된 명령문을 가전기기에 전달하는 가전기기 통신모듈로 구성되어 있다.



(그림 2) 컨텍스트 응용 구성

가. 미들웨어 통신모듈

컨텍스트 응용에서 미들웨어 통신모듈은 미들웨어와 통신을 하기위한 부분으로 메시지전송에 대한 신뢰와 빠른 메시지 처리를 위해 TCP통신을 통하여 컨텍스트 응용을 구현하였다. 이는 UTOPIA에서 JMS통신 방법을 사용하여 메시지를 전송할 때 발생하는 메시지 유실 현상과 여러 컨텍스트 응용 연결로인한 메시지 처리속도 저하의 문제점을 해결하고자 TCP통신을 사용한다. TCP통신은 신뢰성 있는 데이터 전송과 안정적인 메시지 처리를 할 수 있어 적합한 통신 방법이다.

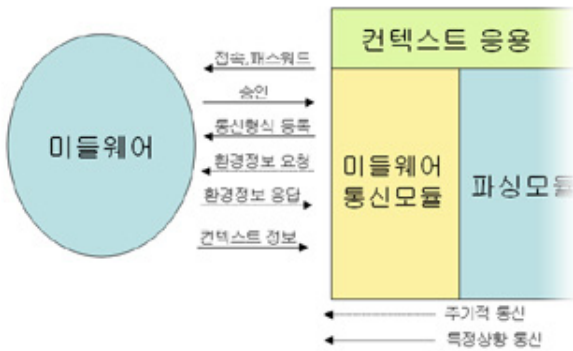
TCP통신을 통하여 미들웨어와 접속할 때 패스워드를 통한 간단한 인증 절차를 거친다. 이는 미들웨어에 허가되지 않은 컨텍스트 응용의 접근을 차단하기 위해서이다. 인증에 통과하면 미들웨어로부터 통신형식등록을 위한 'OK' 메시지를 받는다. 통신형식등록 상태가 되면, 컨텍스트 응용은 미리 정의되어 있는 통신형식을 등록한다.

<표 1> 통신형식 등록의 예

```

Enterence(?x, w:_IN)
Name(?x, w:_cheon)
Is(?x, ?y)
    
```

위의 통신형식은 사용자의 출입여부, 이름, 위치, 외출 등 여러 상황을 등록할 수 있다. 등록된 통신형식은 각각의 컨텍스트 응용에서 관심 있는 상황으로, 미들웨어에서 등록된 상황이 발생하면, 각각의 컨텍스트 응용에게 컨텍스트 정보를 전송한다. 또한 컨텍스트 응용은 미들웨어에 센서로부터 수집된 현재 환경 정보를 주기적으로 요청하여 가지고 있다. 이는 가전기기제어를 위해 현재 환경정보와 상황설정정보를 비교하여 가전기기의 제어를 판단하기 위해서다. 이렇게 수집된 환경정보는 DB파일로 저장되어 파싱모듈에서 가전기기 제어 명령문을 구성할 때 사용된다.



(그림 3) 통신모듈과 미들웨어 통신 구성

나. 파싱모듈

미들웨어 통신모듈을 통하여 들어온 컨텍스트 정보는 파싱모듈을 통하여 분석 처리되어 가전기기 통신모듈로 보내진다.

미들웨어로부터 들어오는 컨텍스트 정보 형식은 아래 표2와 같다.

<표 2> 컨텍스트 정보 형식

```

Situation:entrance(x,y)
State:temperature(x,y)
    
```

파싱모듈은 위의 컨텍스트 정보로부터 현재 상황과 상태를 추출한다. 파싱되어 추출된 상황과 상태는 사용자가 미리 정의해둔 DB정보를 바탕으로 상황과 상태에 맞추어 가전기기를 제어하는 명령문을 구성한다. 사용자가 미리 정의해둔 DB는 사용자가 어떤 상황에 맞추어 그에 해당하는 가전기기를 설정해둔 데이터를 말한다. 예를 들어 사용자가 현관에 입실하였을 경우 TV와 현관 전등을 켜는 내용을 DB에 정의해 둔다. 이렇게 정의해둔 상황과 컨텍스트 정보를 통하여 알아낸 상황이 일치할 경우 정의해둔 상황의 해당가전기기 제어 값과 미들웨어 통신모듈을 통

해 저장된 환경정보 DB 값을 비교한다. 비교하여 서로 값이 틀릴 경우 정의해둔 상황DB의 가전기기 값을 바탕으로 가전기기 제어 명령문을 구성한다. 이렇게 구성된 가전기기 제어 명령문은 가전기기 통신모듈을 통하여 가전기기로 전송된다.

미들웨어로부터 들어온 환경정보는 파싱모듈로 전해져 현재 실행중인 상황을 체크하는 데이터로 사용된다. 예를 들어 현재 온도를 15℃에서 20℃로 올리는 상황에 따라 온풍기를 실행시켰다. 외부 환경정보를 모를 경우 현재 온도가 얼마인지 전혀 예측 할 수 없어 온풍기를 중지할 수 없고, 온도는 계속해서 증가하게 된다. 그래서 이와 같은 경우를 해결하기 위해 미들웨어 통신모듈은 미들웨어로부터 들어온 환경정보를 파싱모듈에게 전달한다. 이렇게 전달된 환경정보는 실행중인 가전기기의 정보와 비교하여 가전기기의 실행 중지를 판단한다.

다. 가전기기 통신모듈

가전기기 통신모듈은 파싱모듈에서 만들어진 가전기기 제어 명령문을 전달하는 역할을 담당한다.

가전기기 통신모듈의 구성은 가전기기의 통신방법에 따라 구성이 달라진다. 가전기기가 RS232통신을 통하여 통신한다면, 컨텍스트 응용의 가전기기 통신모듈을 RS232 통신으로 구성하며, RS485통신을 통하여 통신하는 가전기기면, 컨텍스트 응용의 가전기기 통신모듈은 RS485로 통신으로 구성한다. 이처럼 다양한 통신을 가지고 있는 가전기기들을 하나의 컨텍스트 응용에서 제어해야만 하는 환경에서는 다양한 통신환경을 하나의 가전기기 통신모듈에 넣어 개발 할 수 있고, 각각의 통신방법에 따라 여러 컨텍스트 응용으로 개발 할 수 있다.

4. 결론

JMS통신을 사용한 컨텍스트 응용은 메시지 유실로 인한 컨텍스트 서비스를 제공 할 수 없는 상황 발생과 다수의 컨텍스트 응용의 연결로 인한 처리속도 저하 문제를 발생한다. 이와 같은 문제를 해결하기 위해서 TCP통신 컨텍스트 응용을 제안하였다. TCP통신은 1대1 통신을 통한 신뢰성 있는 데이터 전송과 다수 메시지의 빠른 처리속도로 유비쿼터스 환경에서 정확한 컨텍스트 정보와 서비스를 제공함으로써 신뢰성과 빠른 메시지 처리가 요구되는 미들웨어와 컨텍스트 응용간의 통신에 적합하다고 판단된다.

앞으로의 연구과제는 컨텍스트 정보를 통하여 다양한 서비스를 제공하기 위해 기존 가전기기 및 여러 응용기기와의 효율적인 연동을 제공하는 것이다.

참고문헌

[1] Tao Gu, Hung Keng Pung, and Da Qing Zhang "A service-oriented middleware for building context-aware services" Journal of Network and Computer

Application(JNCA), vol. 28, Issue 1, pp. 1-18, January 2005.

[2] 정국상 "미들웨어 독립적인 응용개발을 위한 컨텍스트 인지 프레임워크" 전남대학교, pp. 45 2007

[3] 강동현, 김화중 "Socket, RMI 및 JMS를 이용한 통신 프로그램의 성능 비교" Journal of Telecommunications and Information, Vol. 8, pp. 103-107,