

웨어러블 컴퓨터를 위한 네트워크 PnP 컴포넌트 연구

장준⁰, 김연수, 이광희, 최훈
 충남대학교 컴퓨터공학과
 {jjang⁰, kimys, khlee, hchoi}@ce.cnu.ac.kr

The Study of a Network PnP Component for the Wearable Computer

June Jang⁰, YounSoo Kim, KwangHee Lee, Hoon Choi
 Department of Computer Engineering, Chungnam National University

요 약

개인 컴퓨터의 소형화에 따라 유비쿼터스 컴퓨팅을 위한 웨어러블 컴퓨터가 개발되고 있다. 하나의 웨어러블 서비스 프로그램이 다양한 웨어러블 네트워킹을 활용하고 웨어러블 디바이스에 독립적으로 동작하기 위해 웨어러블 컴퓨터용 미들웨어가 필요하다. 웨어러블 컴퓨터용 미들웨어는 자원제약적 환경에 적합하도록 설계되어야 하며 BAN(Body Area Network)을 구성하는 다양한 웨어러블 디바이스의 동적 환경설정 기능을 지원해야 한다. 본 논문에서는 초경량, 초절전 웨어러블 미들웨어를 소개하고, 소형 웨어러블 디바이스간의 통신을 위해 동적 디바이스 환경설정 및 BAN 구성을 가능하게 하는 네트워크 PnP(Plug and Play) 컴포넌트를 설계 하였다.

1. 서론

최근 개인 컴퓨터의 환경이 데스크 탑에서 PDA, Smart Phone 의 휴대 가능한 컴퓨터로 바뀌면서 유비쿼터스 컴퓨팅을 위한 입고 다니는 컴퓨터 즉, 웨어러블 컴퓨터에 대한 다양한 연구가 진행되고 있다 [1]. 예를 들어, 필립스와 리바이스는 공동으로 MP3 기능이 지원되는 재킷을 이미 제품화하였다.

유비쿼터스 컴퓨팅을 위한 웨어러블 컴퓨터는 일반적인 이동 단말 컴퓨터와는 다른 제약사항을 가진다. 기능적인 면에서, 사용자의 상황을 미리 인식하여 적절한 명령을 실행해야 하고, 사용자의 관점에서 이용하기 쉬워야 한다. 다양한 네트워크 환경도 지원하여야 한다. 하드웨어 관점에서도 일반적인 모바일 디바이스와 달리 항상 동작되어야 하므로, 전력 관리가 필요하고, 사용자의 상황에 대한 정보를 얻어오는 다양한 센서들과 정보를 교환할 수 있어야 한다. 또한, 웨어러블 디바이스는 사용자의 행동에 방해되지 않는 범위에서 작동되어야 한다.

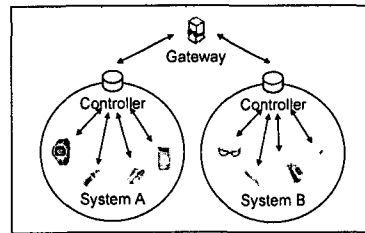
다양한 디바이스 간의 상호작용을 위해서 일반적인 분산 네트워크 환경의 CORBA(Common Object Request Broker Architecture)나 Jini(Java intelligent network infra-structure)와 같이 미들웨어를 사용할 수 있다 [2, 3]. 하지만, 분산환경에서의 미들웨어는 적은 리소스를 활용하여야 하는 웨어러블 컴퓨터에는 적합하지 않다.

본 논문에서는 웨어러블 컴퓨터들을 위한 저전력의 경량 미들웨어를 소개한다. 본 미들웨어는 웨어러블 컴퓨터들에 적합하게 통신, 전원 관리, 보안, 프로파일 관리, 네트워크 PnP, 상황인식 등의 컴포넌트 형태로 이루어져 있다. 미들웨어 컴포넌트들은 사용자가 원하는 기능의 컴포넌트만을 선택하여 설치할 수 있다.

본 논문에서는 미들웨어 컴포넌트 중 웨어러블 디바이스 간의 연동을 위해 필요한 네트워크 PnP 컴포넌트 설계를 제시한다. 네트워크 PnP 컴포넌트는 디바이스가 웨어러블 네트워크 환경에서 사용자의 설정 없이 자유로운 참여 및 탈퇴가 가능하도록 동적 네트워크 구성 기능을 지원한다. 또한 웨어러블 디바이스의 이벤트를 감지하여 다른 웨어러블 디바이스를 제어할 수 있다.

2 장에서는 웨어러블 시스템을 위한 경량 미들웨어를 소개하고, 3 장에서는 네트워크 PnP 에 대해 설명 및 웨어러블 환경을 위한 네트워크 PnP 컴포넌트를 제안한다.

2. 웨어러블 컴퓨터를 위한 경량 미들웨어

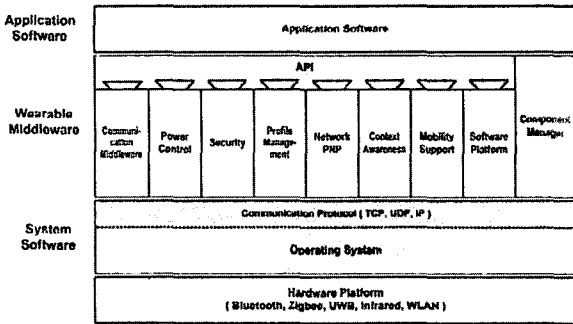


[그림 1] 웨어러블 컴퓨터 구조

[그림 1]은 웨어러블 컴퓨터 구조를 보여준다. 웨어러블 컴퓨터는 위치를 파악하는 센서, 데이터를 저장하는 저장 공간 스피커와 같은 임출력기 등 사용자가 몸에 지니고 다닐 수 있는 작은 웨어러블 디바이스 들로 구성된다. 또한, 기기들의 상호작용을 제어하기 위해서 컨트롤러와 같은 메인 프로세서 컴퓨팅 능력을 가지는 디바이스가 존재해야 한다.

게이트웨이는 외부 네트워크와 연결을 위해 존재하며, 컨트롤러가 게이트웨이 기능을 대신 할 수 있다.

⁰ 본 연구는 정보통신부의 선도기반기술개발사업의 지원으로 수행되었음.



[그림 2] 웨어러블 컴퓨터를 위한 경량 미들웨어 구조

여러 플랫폼이 탑재된 웨어러블 디바이스와 컨트롤러, 게이트웨이의 효과적인 상호 연동을 위해서 미들웨어를 사용할 수 있다. [그림 2]는 현재 연구 중인 웨어러블 컴퓨터를 위한 경량 미들웨어 구조를 보여준다.

RTOS(Real Time Operating System)의 IP 프로토콜 기반에 탑재되는 웨어러블 컴퓨터를 위한 웨어러블 미들웨어는 [그림 2]와 같이 여러 컴포넌트와 컴포넌트 관리자로 구성된다. Communication Middleware 컴포넌트는 프로세스간 통신 지원을 위해, Power Control 컴포넌트는 전원 절약을 위해, Security 컴포넌트는 SSL 등의 보안 프로토콜을 사용하기 위해, Profile Management 는 사용자, 서비스, 디바이스 프로파일의 생성 및 관리를 위해 존재한다. 또한 동적 BAN 구성을 위한 네트워크 PnP 컴포넌트, 사용자의 상황인식을 위한 Context Awareness 컴포넌트, 이동성 지원을 위한 Mobility Support 컴포넌트, WIP(Wireless Internet Platform for Interoperability) 등의 미들웨어 소프트웨어를 탑재할 수 있는 Software Platform 컴포넌트가 존재한다. 컴포넌트들은 웨어러블 컴퓨터 용도나 응용 서비스에 맞게 재구성 될 수 있다.

본 논문에서 제안된 네트워크 PnP 컴포넌트는 물리적 데이터 전송 미디어(Bluetooth, Zigbee 등)에 독립적이고 웨어러블 사용자 네트워크인 BAN 관점에서 웨어러블 디바이스의 상호 연동 및 동적 환경 설정 지원을 위해 정의된 컴포넌트이다.

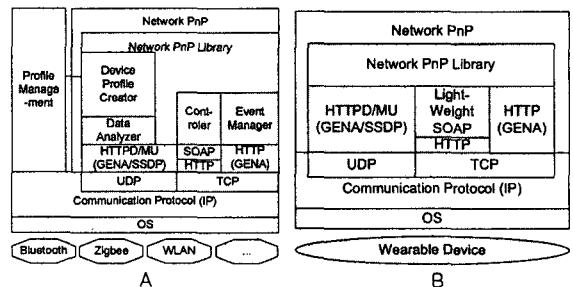
3. 웨어러블 컴퓨터를 위한 네트워크 PnP 컴포넌트 설계

네트워크 PnP 는 분산 시스템 상에서 자원의 자유로운 참여(추가) 및 탈퇴(제거)를 지원하여 사용자에게 Zero Configuration 을 가능하게 해주는 미들웨어로 SUN Microsystems 의 Jini 와 Salutation consortium 의 Salutation, Microsoft 의 UPnP(Universal Plug and Play) 등이 대표적이다 [4, 5]. Jini 는 사용자가 '등록서비스' 를 통해 서비스를 찾아 사용하는 자바 기반의 미들웨어이고, Salutation 은 다양한 전송 계층에 독립적으로 동작하는 미들웨어이다. UPnP 는 컨트롤 포인트를 이용하여 디바이스들을 제어하고, IP 기반의 TCP/UDP 를 사용하여 디바이스들간의 상호작용을 한다는 점에서 웨어러블 컴퓨터 환경에 적합하다. 하지만, UPnP 는 디바이스의 정보를 XML(Extensible Markup Language) 형식의 데이터만을 사용하기 때문에 WAP(Wireless Application Protocol)의 WML(Wireless Markup Language), Jini 의 자바 객체와 같은 다양한 디바이스 데이터 형식을 사용하는 웨어러블 환경에는 적합하지 않다.

웨어러블 컴퓨터를 위한 네트워크 PnP 컴포넌트의 요구사항을 다음과 같이 정의하였다.

- 가. 다양한 웨어러블 네트워크 프로토콜을 통해 전송된 데이터 형식들을 분석하여 디바이스 프로파일을 생성해야 한다. 단, 디바이스가 웨어러블 컴퓨터의 디바이스 프로파일을 제공한다면, 이를 직접적으로 사용해야 한다.
- 나. 생성된 디바이스 프로파일을 프로파일 관리자 컴포넌트로 전달할 수 있어야 한다.
- 다. BAN 에 새로 참여하는 디바이스를 감지할 수 있어야 한다.
- 라. BAN 에 참여한 디바이스를 제어할 수 있어야 한다.
- 마. 참여한 디바이스에서 보내오는 이벤트를 감지할 수 있어야 한다.
- 바. TCP, UDP 전송 프로토콜을 기반으로 한다.

3.1 웨어러블 미들웨어를 위한 네트워크 PnP 컴포넌트 구조



[그림 3] 네트워크 PnP 컴포넌트 구조

[그림 3]는 2 가지 종류의 네트워크 PnP 컴포넌트를 보여준다. A 는 손목시계와 같이 게이트웨이 역할을 할 수 있는 디바이스를 위한 컴포넌트로서 디바이스 프로파일 생성과 외부 네트워크와의 연결이 가능하도록 설계하였다. B 는 소형 웨어러블 디바이스를 위한 컴포넌트로서 SOAP(Simple Object Access Protocol) 대신 제어 명령의 수신 및 반응만을 위한 Light-Weight SOAP 을 이용한다. 다음은 각 모듈에 대한 설명이다.

- 가. SSDP(Simple Service Discovery Protocol): IP/UDP 기반에서 디바이스의 존재 여부를 찾는 프로토콜이다. SSDP 는 HTTPU(HTTP Unicast) 및 HTTPMU(HTTP Multicast) 기반 위에 구축되며, 제어 포인트가 네트워크 상에서 원하는 리소스를 검색하는 방법 및 장치들이 네트워크 상에서 자신들의 가용성을 알리는 방법을 정의한다.
- 나. SOAP: 디바이스를 찾은 후 SOAP 를 이용하여 디바이스를 제어한다. SOAP 는 XML 및 HTTP 의 용법을 정의하여 원격 프로세서 호출을 실행한다. SOAP 은 보안용으로 SSL(Secure Sockets Layer)을 사용할 수 있다.
- 다. Light-Weight SOAP: 일반적인 SOAP 에서 제어 명령에 대한 수신 및 반응만을 위한 모듈만 탑재한다.
- 라. GENA(Generic Event Notification Architecture): 디바이스가 자신의 이벤트를 알려주기 위해 GENA 형식을 사용하여 생성한 데이터는 SSDP 를 이용하여 전송한다.
- 마. Data Analyzer: 웨어러블 디바이스의 다양한 정보 형식을 웨어러블 컴퓨팅 환경이 이용할 수 있도록 분석한다.
- 바. Device Profile Creator: Data analyzer 가 분석한 정보를 바탕으로 디바이스 프로파일을 생성한다. 디바이스 프로파일은 생성 후 Profile Manager 컴포넌트로 전달한다 또한 SOAP 의 제어 명령을 위해 필요한 XML 파일을

생성한다.

사. Controller and Event Manager: Controller 는 SOAP 프로토콜을 이용하여 응용 프로그램이 더욱 쉽게 제어 가능하도록 API 를 정의한다. Event Manager 는 GENA 를 더욱 쉽게 이용하여 SSDP 를 사용하는 API 를 정의한다.
 아. 네트워크 PnP Library: 응용에서 UPnP 사용시 제공하는 API 들이다.

```

Device profile {
    double long DeviceID;
    DeviceInfo DvInfo;
    DeviceCapability DvCapability;
}
    
```

[그림 4] 디바이스 프로파일 구조

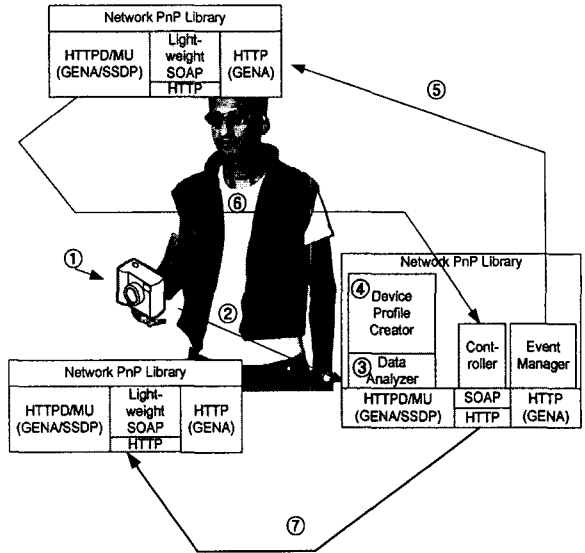
네트워크 PnP 컴포넌트에서 생성된 디바이스 프로파일은 [그림 4] 과 같이 DeviceID, DvInfo, DvCapability 로 구성된다. DeviceID 는 디바이스 프로파일의 고유 아이디를 표현한다. DvInfo 는 디바이스의 모델명, 모델번호, 제조일, 메모리 용량, 저장공간 등 디바이스에 대한 정보를 표현하는 필드이다. 마지막으로 DvCapability 는 디바이스의 타입, 즉 오디오, 디스플레이, 프린트 등의 정보와, 오디오의 Stop, Play, Pause 같은 디바이스 고유의 능력에 대한 실행 가능 명령어를 포함한다.

3.2 웨어러블 미들웨어를 위한 네트워크 PnP 컴포넌트의 동작

네트워크 PnP 컴포넌트의 동작 과정은 다음과 같다.

- ① 디스커버리 단계: BAN 에 카메라 디바이스가 참여하면, SSDP 프로토콜을 이용하여 자신을 제어할 수 있는 손목시계 디바이스를 찾는다. 손목시계 형 디바이스가 웨어러블 컴퓨터의 main processing 디바이스라고 가정하였다.
- ② 정보제공 단계: 카메라 디바이스가 손목시계 디바이스를 찾으면 자신의 디바이스 정보를 손목시계에 알려준다.
- ③ 정보 분석 단계: 손목시계 디바이스의 Data Analyzer 는 다양한 네트워크를 통해 얻어온 정보를 웨어러블 미들웨어가 알 수 있게 분석한다.
- ④ 디바이스 프로파일 생성 단계: Device Profile 은 분석한 정보를 바탕으로 디바이스 프로파일을 생성한다.
- ⑤ 이벤트 발생 단계: 새로운 디바이스가 추가 되었음을 BAN 을 구성하고 있는 웨어러블 디바이스들에게 전달한다.
- ⑥, ⑦ 이벤트 반응 및 제어 단계: 손목시계 디바이스는 웨어러블 디바이스가 주기적으로 발생하는 이벤트를 감지하여 적절한 제어 명령을 내린다.

웨어러블 미들웨어를 위한 네트워크 PnP 컴포넌트는 [그림 5] 과 같이 게이트웨이 역할을 하는 손목시계와 일반적인 웨어러블 디바이스에 각각 다른 컴포넌트가 탑재된다. 게이트웨이의 경우 외부 네트워크의 디바이스와의 상호 연동을 위해 SOAP 이 필요하지만, 웨어러블 디바이스의 경우 게이트웨이를 통해서만 제어되기 때문에 Light-Weight SOAP 만으로도 통신 가능하다.



[그림 5]: 네트워크 PnP 컴포넌트 동작 과정

4. 결론

본 논문에서는 웨어러블 컴퓨터에 탑재될 경량 웨어러블 미들웨어 구조를 소개하고, 웨어러블 컴퓨터에 적합한 네트워크 PnP 컴포넌트를 설계하였다. 웨어러블 컴퓨터는 제한적인 자원을 이용해야 하고, 웨어러블 디바이스로부터 제공되는 다양한 형식의 데이터를 분석해야 하기 때문에, 일반적인 네트워크 PnP 미들웨어를 사용할 수 없다. 본 논문에서는 공인된 프로토콜을 사용하는 UPnP 를 사용하여 웨어러블 환경에서의 프로토콜 신뢰도를 높였다. 또한 웨어러블 컴퓨터의 요구사항을 분석하여 웨어러블 디바이스가 제공하는 다양한 정보의 데이터 형식을 지원하도록 하였다.

본 연구는 향후에 네트워크 PnP 미들웨어를 구현하고 다른 웨어러블 미들웨어 컴포넌트와 통합하여 웨어러블 컴퓨터에 탑재할 것이다. 웨어러블 컴퓨터는 개인의 유비쿼터스 컴퓨팅 환경 구축에 중심적인 역할을 할 것이다.

5. 참고문헌

- [1] M. Weiser, " The Computer for the Twenty-First Century," Scientific American, pp. 94-10, September 1991
- [2] OMG, " The Common Object Request Broker: Architecture and Specification, Revision 2.3.1" , Object Management Group, 1999
- [3] Sun Microsystems Inc., " Jini Architecture Specification" 1999
- [4] Salutation Consortium. " Salutation Architecture Specification (Part-1), 1999
- [5] Microsoft Inc., " UPnP Device Architecture 1.0" , 2003