

리눅스 블루투스 프로토콜 스택(BlueZ)을 이용한 위치 인식 시스템 설계

이재우, 김진형, 조위덕

아주대학교 유비쿼터스 스마트 스페이스 시스템 연구실

leejeawoo7@paran.com, hishope@ajou.ac.kr, chowd@ajou.ac.kr

Location-Aware System Design using the Bluetooth Protocol Stack (BlueZ) of Linux
in Ubiquitous computing application

JaeWoo Lee, JinHyung Kim, We-Duke Cho

Ajou University Ubiquitous Smart Space System Lab.

요 약

본 논문에서 구현하고자 하는 유비쿼터스 컴퓨팅 응용에 필요한 위치 인식 시스템의 주 요소는 블루투스 프로토콜 스택(BlueZ)에서 제공하는 RSSI(Received Signal Strength Indicator) 값을 측정하는 블루투스 AP, 측정된 RSSI 값을 위치 인식 서버에 전달하기 위한 무선 AP 공유기 그리고, 받은 데이터로 위치 값을 측정하는 위치 인식 서버 및 Context Broker(고 수준의 상황 정보를 추론하는 서버 역할)로 이루어져 있다. 전체적인 동작 시스템은 위치 값을 측정하고자 하는 이동 매체(마스터)를 중심으로 최대 여덟 개 까지 네트워크가 가능한 블루투스 AP(슬레이브)장치로 구성된 피코넷(Piconet) 영역에서 삼각측량 필요에 적절한 세 개의 블루투스 AP를 RSSI값을 이용하여 분류 한 후 이동 매체의 위치를 측정한다. 그 결과로 나온 데이터는 피코넷 영역에서 가장 가까운 무선 AP 공유기를 거쳐서 위치 값을 측정하는 위치 인식 서버에 전달한 후, 그 결과 값으로 Context Broker에서 상황 정보를 추론해서 Community Manager에서 유비쿼터스 컴퓨팅 응용에 맞게 서비스를 구현한다. 또한, 위와 같은 시스템 내부 구조 및 데이터 처리는 리눅스 운영체제 내에서 디바이스 드라이버와 사용자 프로그램으로 구현된다.

1. 서론

유비쿼터스 컴퓨팅은 사용자에게 상황에 따라 서로 다른 서비스와 커뮤니케이션을 위해 다양한 장치를 통해서 구현된다. 최근 유비쿼터스 컴퓨팅 환경을 구현하기 위한 서비스로서 사용자에게 적절한 순간에 주변 환경의 상황을 고려하여 이동 객체의 위치 파악과 함께 커뮤니티를 형성하고 그에 적합한 위치 인식 서비스를 인가하는 시스템이 중요한 요소로 대두되고 있다. 또한, 이러한 위치 인식 시스템에 의해서 측정된 위치 데이터는 유비쿼터스 서비스 구현에서 상황 인식 서비스를 제공하기 위한 필수적인 요소이다.

위와 같은 위치 인식 시스템의 설계에 있어서 이용되는 블루투스는 저전력, 저비용으로 사용 요구에 따라 다양한 거리를 지원하는 장치들과 10미터 반경 내에서 정보 교환 능력을 갖고, DAP(Data Access Point)로 강력한

전송수단(약+20dB)과 좋은 감도의 수신단(약-90dB)을 사용한다면 개방된 공간에서 100m까지 도달 할 수 있으므로 다양한 유비쿼터스 컴퓨팅 응용을 구현하기 위해서 적합하다. [1,9]

본 논문의 위치 인식 시스템의 설계는 BlueZ에서 제공하는 RSSI 함수를 이용하여 삼각 측량에 필요한 이동 매체에서 가장 가까운 블루투스 AP 세 개를 분류하고, 선택된 블루투스 AP에서 각각의 RSSI값을 무선 AP 공유기에 전달되도록 구현되었다. 무선 AP 공유기에 의해서 전달된 데이터를 가지고 위치 인식 서버에서 삼각측량법과 Friis의 공식을 이용하여 이동 매체의 위치를 측정하게 된다. 또한, 변화의 적용이 필요로 하는 유비쿼터스 환경에서 서비스를 제공하기 위해서 Context Broker에서는 상황 정보를 추론하는 역할을 하고, 좀 더 상위 레벨에서는 리눅스 커널에서 제공하는 라이브러리와 BlueZ에서 제공하는 API와 코드를 이용하여

위치 인식 및 그에 부가되는 서비스에 필요한 세부적인 사항을 제어 및 설정을 한다.

논문의 구성은 2장에서 위치 인식 시스템 필요로 하는 요소인 BlueZ와 Bluez의 HCI 소켓 기능 및 HCI에서 제공하는 함수 hci_read_rssi에 대해서 설명하고, 3장에서는 시스템 구성에 있어서 필요한 상각측량법과 friis 공식, 위치 인식을 위한 디바이스 드라이버와 사용자 프로그램과의 구조 및 데이터 처리 단계에 대해서 설명하겠다. 그리고, 4장에서는 위치 인식 시스템에 이용되는 블루투스에 대한 앞으로의 전망에 대해서 설명하겠다.

2. 블루투스 위치 인식 시스템 요소

위치 인식 시스템을 구현하고자 할 때의 중요한 시스템 요소는 BlueZ와 Bluez에 구성된 모듈 HCI, 그리고 신호 세기를 측정 함수 hci_read_rssi 및 데이터에 대한 전송 및 처리를 구현한 디바이스 드라이버가 있다.

BlueZ는 리눅스 커널을 위한 블루투스 프로토콜 스택이며, 블루투스 무선표준 스펙을 구현한 일종의 라이브러리이다.[12] GPL(General Public License)에 의해서 배포된 리눅스 커널 2.4.6부터 BlueZ는 모듈로 적재되어 있거나 커널의 일부로서 컴파일 되어서 포함되어져 있다. 또 하나의 요소로서 리눅스 운영체제의 디바이스 드라이버는 시스템이 사용되는 하드웨어를 사용자 프로그램에서 제어 할 수 있도록 커널에서 제공하는 일종의 라이브러리이다.

2.1 리눅스 블루투스 프로토콜 스택 (BlueZ)

BlueZ의 특징으로는 유연성과 효율성을 가지고 있고, 모듈적 구조로 다중 블루투스 장치를 지원하며, 하드웨어 추상화하고 있다. 또한, 멀티 스레드로 동작하는 데이터에 대한 처리 및 다중 층을 위한 표준 소켓 인터페이스를 지원한다.

BlueZ의 구성 모듈은 다음과 같이 구성되어 있다.

- HCI 코어
- HCI UART, USB, PCMCIA, Virtual HCI 장치 드라이버
- L2CAP 프로토콜 모듈
- 구성 및 테스트 유틸리티

표 1. BlueZ 소프트웨어 패키지

패키지	용도
Bluez-libs	특정 기능을 위해 모든 블루투스 소프트웨어 개발을 위한 프로그램이 요구하는 응용 라이브러리
Bluez-utils	블루투스 장치의 컨트롤 및 초기화하기 위한 주요 유틸리티

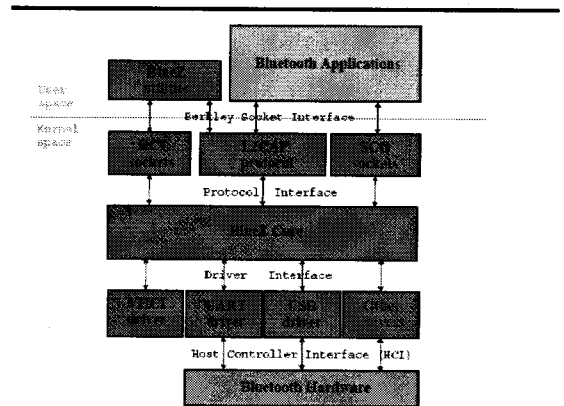
Bluez-sdp	블루투스 서비스의 검색하는 SDP(Service Discovery Protocol) 프로그램
Bluez-pan	블루투스를 사용하여 개인 영역 망의 이용을 가능하게 하는 개발 프로그램
Bluez-hcidump	블루투스 패킷의 모니터링을 위한 디버깅 프로그램
Bluez-bluefw	블루투스 장치에 기초한 Broadcom 칩 셋을 위한 펌웨어

표1에서 보는 것과 같이 "bluez-kernel"로 불리는 BlueZ 소프트웨어 패키지는 각 특정 기능을 담당하는 패키지로 분류되어 있으며, 블루투스 코어의 설정을 위한 툴을 포함한다.

BlueZ에서 사용되는 툴로서 중요한 두 가지의 툴로는 HCI 레벨에서 장치를 구성하는 'hcidconfig'와 장치의 주요 특성을 사용하기 위한 'hccitool'이 있다. 블루투스 인터페이스를 위한 'hcidconfig' 메인 콜인 '#hcidconfig hci0 up'는 리눅스에서 네트워크 커맨드 'ifconfig'와 같이 동작을 하고, 블루투스 장치를 초기화하기 위해 쓰여진다. [11]

2.2 BlueZ의 HCI 기능 및 신호 세기 측정 함수 hci_read_rssi

BlueZ는 HCI, L2CAP, SCO등의 특정 기능을 담당하는 블루투스 스택, 즉, 소프트웨어 프로토콜 및 소켓으로 구분되어 있다. 그림1에서 HCI(Host Controller Interface) 프로토콜은 모듈 안에서 블루투스 스택의 소프트웨어 부분들 사이의 통신을 위해 사용된다. 또한, 링크 매니저의 구성과 모듈의 재 시작 및 초기화를 위해서는 HCI를 사용해야 한다.[6]



■ BlueZ에 의해서 주어진 요소들

그림1. BlueZ 스택

HCI층은 블루투스 모듈과 호스트 프로세서와의 전력 분배를

위해 블루투스 실행에서 호스트 시스템 즉, 리눅스와 블루투스 모듈 사이를 조절하고, 컨트롤 레지스터를 읽고 조절할 뿐만 아니라 블루투스 하드웨어 상태를 읽기 위해서 베이스 밴드 컨트롤러 및 링크 매니저와 같이 통신 기능을 제공한다. 그리고, 위의 기능뿐만 아니라 블루투스 모듈에 대해서 L2CAP 데이터를 주거나 받을 수 있는 기능을 제공한다. 이러한 블루투스 스택의 HCI 층은 받은 신호의 세기를 출력하는 기능을 가지고 있다.[5] BlueZ 코드 'Bluez-libs' 패키지에서 hci.c 파일 안의 함수 hci_read_rssi는 RSSI(Received Signal Strength Indicator)값을 출력한다.

이렇게 출력된 RSSI 값은 위치 측정 시스템을 구현에 있어서 필요한 이동 개체로부터 세 개의 기준점 사이의 거리를 구하기 위한 하나의 요소로 쓰인다. 그림2는 hci.c 파일 안에 정의된 hci_read_rssi 함수에 대한 코드이다.

```
int hci_read_rssi(int dd, uint16_t handle, int8_t *rssi, int to)
{
    read_rssi_rp rp;
    struct hci_request rq;

    memset(&rq, 0, sizeof(rq));
    rq.ogf = OGF_STATUS_PARAM;
    rq.ocf = OCF_READ_RSSI;
    rq.cparam = &handle;
    rq.clen = 2;
    rq.rparam = &rp;
    rq.rlen = READ_RSSI_RP_SIZE;

    if (hci_send_req(dd, &rq, to) < 0)
        return -1;

    if (rp.status) {
        errno = EIO;
        return -1;
    }

    *rssi = rp.rssi;
    return 0;
}
```

그림 2. BlueZ hci.c의 hci_read_rssi 함수

양의 RSSI값은 GRPR(Golden Receive Power Range)의 상위를 기준으로 받아진 신호를, 반대로 음의 RSSI값은 GRPR의 하위를 기준으로 받아진 신호를 데시벨 단위로 측정되고, RSSI값이 0일 경우에는 최적의 GRPR의 범위에 있게 된다.

3. 블루투스 위치 인식 시스템 설계

위치 인식 시스템 설계에 있어서 최소 세 개의 블루투스 AP가 필요하다. 구현은 블루투스를 장착한 이동 매체에 최대 여덟 개까지 연결할 수 있는 블루투스 AP 중 적절한 세 개의 블루투스 AP를 판별하기 위해서는 피코넷에 설치된 AP와 이동 매체간의 각각의 RSSI값을 구하고, 이 값으로 가장 가까운 세 개의 값으로 삼각 측량에 필요한 블루투스 AP를 판별하게 된다. 또한, 기본적으로 블루투스 AP는 리눅스 운영체제를 기반하여 위치인식을 처리하게 되는데, 이 처리는 두 개의 메모리 공간에 설정된 사용자 프로그램과 타이머 인터럽트가 설정된 디바이스 드라이버를 통하여 일정 시간 간격으로 이동하는 개체의 위치를 인식한다.

3.1 삼각측량 및 Friss 공식을 사용한 위치 측정

블루투스 이동 개체의 위치를 측정하기 위해서는 세 개의 기준점이 필요하다. 그림 3과 같이 기준점인 블루투스 AP1, 2, 3 각각의 좌표를 (x_1, y_1) , (x_2, y_2) , (x_3, y_3) 라고 하고, 구하고자 하는 이동 개체의 좌표를 (x, y) 라고 하자.

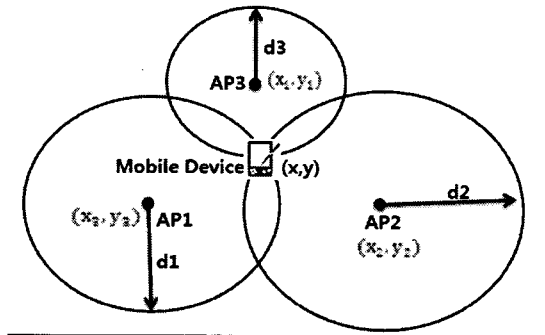


그림 3. 삼각 측량을 이용한 위치 측정

이동 개체로부터 세 개의 블루투스 AP까지 거리를 각각 d_1 , d_2 , d_3 라고 하면, 이동 개체로부터 각 기준점 사이의 거리는 표2와 같이 피타고라스 정리에 의해서 계산된다.

표2. 위치 인식을 위한 피타고라스 정리

식	설명
$d_1^2 = (x - x_1)^2 + (y - y_1)^2$	AP1과 이동 개체 거리
$d_2^2 = (x - x_2)^2 + (y - y_2)^2$	AP2와 이동 개체 거리
$d_3^2 = (x - x_3)^2 + (y - y_3)^2$	AP3과 이동 개체 거리

위에서 이동 매체로부터 3개의 블루투스 AP사이의 거리를 측정하는 값은 다양한 방법으로 구할 수 있다. 이러한 방법으로는 RSSI 값으로 구할 수 있으며, 도착 시간(TOA) 혹은 도착 시간차(TDOA)를 이용하여 구할 수 있다. 본 논문에서는 BlueZ에 의해서 측정된 RSSI의 값을 이용하여 삼각측량법을 이용하여 위치 측정을 하도록 하겠다.

이동 매체가 송신한 신호의 손실은 friss의 공식(1)

$$L = 20 \log_{10} \left(\frac{4\pi d}{\tau} \right) [dB] \quad (1)$$

을 통해 구할 수 있다. 표 3에서 보는 것과 같이 friss의 공식은 자유 공간에서의 경로 손실을 구하는 것이며, τ 는 전파의 파장을 나타내며 거리 d 와 동일한 단위를 사용한다.

표 3. friss에 대한 설명

기호	설명
c	전파 속도
τ	전파의 파장
f	모듈에 따른 주파수
L	이동 개체가 송신한 신호의 손실
d	이동 개체와 블루투스 무선 AP 사이의 거리

friss의 공식은 자유 공간에서의 경로 손실을 구하는 것이며, τ 는 전파의 파장을 나타내며 거리 d와 동일한 단위를 사용한다. 위의 식을 블루투스 AP와 이동 개체 사이의 거리로 나타내는 d에 관해서 나타내면,

$$d = \frac{\tau}{4\pi} 10^{\frac{L}{20}} = \frac{c}{4\pi f} 10^{\frac{L}{20}} \quad (2)$$

다음과 같은 식(2)을 도출할 수 있다. c는 전파 속도이며 f는 주파수를 나타낸다. 위의 식에 각 블루투스 모듈에 따른 f를 대입하고, 공기 중 무선 신호의 전파 속도인 $3 \times 10^8 \text{m/sec}$ 와 BlueZ hci_read_rssi 함수의 RSSI를 이용해서 L(이동 개체가 송신한 신호의 손실)을 구하고, 대입함으로써 두 지점 사이의 거리인 d를 구할 수 있다. 이렇게 d를 구함으로써 표 2에 대입하면, 이동 개체의 현재 위치 인식을 할 수 있다.

3.2 위치 인식 서비스 구현을 위한 리눅스 디바이스 드라이버와 사용자 프로그래밍의 구조

위치 인식 시스템을 구현하기 위해서는 블루투스 모듈 장치를 추상화하여 서비스에 적합한 사용자 프로그램을 구현하고, 정형화된 인터페이스를 통해 디바이스를 접근할 수 있도록 설계한 프로그램이 필요하다. 리눅스 커널에서는 위와 같은 기능을 담당하고, 블루투스 모듈과 같은 장치에 대한 관리에 필요한 인터페이스 구현에 요구되는 함수와 자료구조의 집합체인 디바이스 드라이버를 제공한다. 또한, 디바이스 드라이버와 연관되어 동작하는 응용 프로그램이 하드웨어를 제어할 수 있도록 인터페이스를 제공할 뿐만 아니라 하드웨어에 독립적인 프로그램을 작성 가능하다.

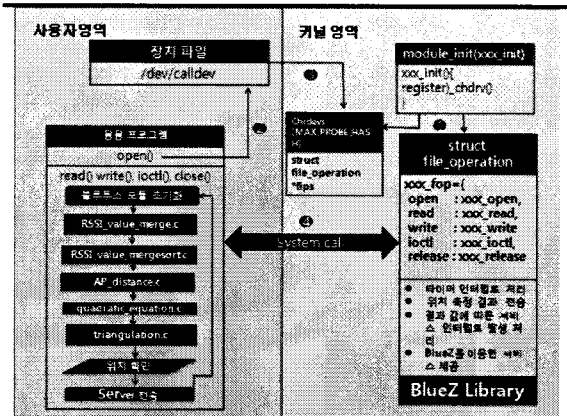


그림 4. 리눅스 디바이스 드라이버 및 사용자 프로그래밍 동작 구조

시스템 설계에서 모듈로써 리눅스 커널에 적재된 디바이스 드라이버와 사용자 프로그래밍간의 동작 구조는 그림 4에서와 같이 구성된다. 두 가지 메모리 공간을 가지고 있는 리눅스 운영체제에서는 '사용자 영역'과 '커널 영역'의 메모리 공간을 분리하여 위치 인식에 필요한 디바이스 드라이버 및 사용자 프로그램을 구현하면 된다. 즉, '커널 영역'에서는 BlueZ에서 제공되는 API나 코드를 이용한 서비스 구현과 실 시간의 위치 검색을 위한 타이머 인터럽트 및 결과 값에 따른 서비스 인터럽트를 제공하는 디바이스 드라이버 구현하고, '사용자 영역'에서는 유비쿼터스 컴퓨팅 응용을 제공하기 위해 필요한 위치 인식에 대한 코드를 작성한다.

그림 4에서 블루투스 모듈을 초기화, 등록 그리고, 하드웨어 검출 처리 및 에러처리를 위해서 xxx_init()함수가 동작하게 한다. (1)번 동작에서는 register_chrdev() 함수의 매개 변수에 등록하게 될 주 번호, 디바이스 드라이버에 선언된 xxx_fop file operation 구조체의 변수 주소, 디바이스 드라이버 영의 선두 주소를 담아 커널 내의 chrdevs 변수에 위치 인식 서비스를 위해서 설계한 디바이스 드라이버를 등록한다. 사용자 프로그램은 (2)번과 같이 디바이스 드라이버의 함수를 사용하기 위해 디바이스 파일을 열고, 디바이스 파일에서 주 번호와 부 번호의 정보를 얻고, 이렇게 얻은 주 번호는 (3)과 같이 chrdevs의 변수에 접근하기 위한 인덱스 값으로 바뀌어 사용한다.

사용자 프로그램의 read(), write(), ioctl(), close() 등의 함수는 open()을 얻은 주 번호를 이용해 chrdevs 배열 변수를 참조 및 이 변수의 필드에 있는 fops를 이용해 xxx_fop구조체의 변수를 얻는다. 각각의 함수에 대응하는 위치 인식을 위한 디바이스 드라이버 함수, 즉 read()함수에 대응하는 xxx_read와 write()함수에 대응하는 xxx_write() 등으로 위치 인식에 필요한 서비스를 블루투스 모듈에 시스템 콜을 한다. 그림 4에서 디바이스 드라이버에 대응되는 사용자 프로그램의 각 파일의 기능으로는 표4과 같이 동작하게 된다. 사용자 프로그램에서 전달된 기능 외에도 블루투스 모듈을 직접적으로 제어하기 위해서 리눅스 커널에서 제공하는 기능들이 동작을 하게 된다. BlueZ에서 제공하는 함수 및 API로 RSSI 값을 얻게 되고, 일정시간 실시간 측정을 위한 소프트웨어 타이머 인터럽트를 처리를 한다. 또한, 결과 값에 따라서 사용자가 원하는 서비스를 받을 수 있도록 구현하고, 측정된 결과 값에 대하여 위치 인식 서버에 전달하는 처리를 하게 된다. [10]

표4. 사용자 프로그램의 구조 및 서비스

파일	용도
RSSI_value_merge.c	✓ 합병 정렬을 이용한 여덟 개의 (블루투스 AP RSSI값-이동 개체M RSSI값)값을 비교
RSSI_value_mergesort.c	✓ RSSI_value_mergesort.c를 사용하기 위한 구성 요소

<p>RSSI_value_mergesort.c</p>	<ul style="list-style-type: none"> ✓ 블루투스에서 마스터에 유일 수 있는 최대의 슬레이브가 여덟 개인 것을 감안하여 이동 개체로부터 가까운 세 개의 블루투스AP들을 구별하는 함수 ✓ (블루투스 RSSI값-이동 매체 RSSI값)이 가장 작은 값들이 이동 개체로부터 가까운 세 개의 블루투스 AP
<p>AP_distance.c</p>	<ul style="list-style-type: none"> ✓ 이동개체로부터 블루투스 AP1,2,3 사이의 거리를 구하는 함수 ✓ 이동 개체 RSSI에서 블루투스AP RSSI를 AP_distance.c 함수의 인자로 대입 ✓ 이동 개체가 송신한 신호의 손실=블루투스 AP의 RSSI-이동 개체의 RSSI ✓ 여기서 BlueZ의 RSSI의 값을 커널 메모리 영역에서 가져옴
<p>quadratic_equation.c</p>	<ul style="list-style-type: none"> ✓ 이동개체의 위치를 인식하기 위한 이차 방정식을 구하는 함수
<p>triangulation.c</p>	<ul style="list-style-type: none"> ✓ 블루투스 AP들의 위치와 앞의 세 개의 점으로부터 이동 개체까지의 거리를 통하여 이동 매체의 좌표 점을 확인 ✓ 피타고라스의 정리를 통해서 세 개의 식이 유도되는데, 두 개의 식으로 x에 관한 방정식을 유도 ✓ x에 관해서 만들어진 식을 나머지 하나의 식으로 대입해서 y에 대한 2차 방정식을 유도

3.3 블루투스 AP 및 Context Broker와 위치 인식 서버 간의 전달 데이터 처리 구조

위치 인식으로 처리된 위치 값과 그에 따른 처리에 대한 결과를 유비쿼터스 컴퓨팅 응용에 필요한 서비스로 설계하기 위해서는 위치 인식 서버에 데이터를 보내고, 그에 따른 값을 Context Broker에서 가공해야 한다. 이러한 시스템의 처리를 위해서 사용한 무선 AP는 물리적 매체인 케이블을 통해서 네트워크로 접속이 가능한 유선AP보다는 이동성이 보장된다. 노트북이나 PDA등의 end-user 시스템들은 액세스가 가능한 AP를 통해서 필요 시에 네트워크로 접속과 사용이 가능하며, 실내에서 공간적인 변화가 있을 시에도 다시 재구축할 필요가 없고, 다양한 무선 네트워크 환경의 구성이 가능하다.[7,8]

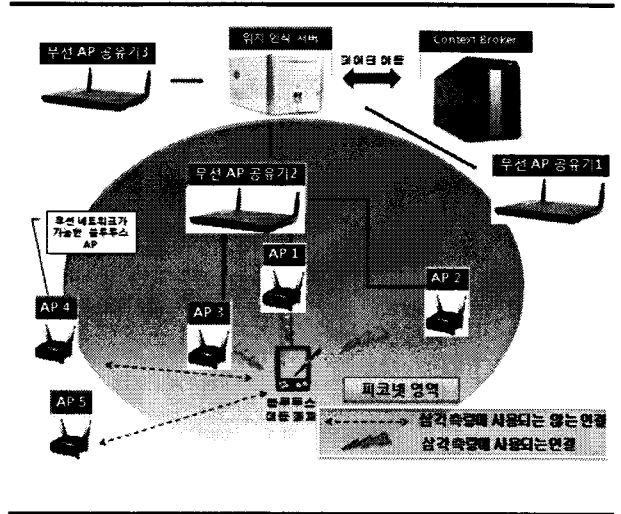


그림 5. 위치 인식 서버로 데이터 전달을 위한 시스템 구조도

그림 5에는 블루투스 AP와 이동 매체의 데이터 전송 구현을 하기 위한 시스템 구조도이다. 앞에서 설명한 것과 같이 블루투스 이동 매체로 인해서 만들어진 피코넷 영역에서 존재하는 세 개의 AP를 이용해서 삼각측량법으로 위치 좌표를 구하기 위한 값들을 도출해낸다. 이렇게 처리된 데이터는 각각의 블루투스 AP에서 피코넷 영역 내부의 가장 가까운 무선 AP 공유기에 수집되고, 다시 이 데이터는 위치 인식 서버로 이동하여 이동 매체의 정확한 위치 정보를 구하게 된다. 다음으로 Context Broker로 이동하여 컴퓨팅 응용 환경에 적합한 상황 정보를 추론하고 세부적이고 다양한 서비스를 제공하게 된다. 또한, 이동 매체가 이동 시에 새로운 피코넷 영역을 구축하여 새로운 데이터가 발생하기 때문에 변화된 데이터 값이 위치 인식 서버에 전달 할 수 있도록 일정 시간 간격으로 데이터를 위치 인식 서버에 넘겨주고, 그에 따른 새로운 컴퓨팅 응용 서비스를 제공한다.

4 결론

지금까지 리눅스에서 지원하는 BlueZ의 특성을 가지고 무선 네트워크 기능을 가지고 있는 블루투스 AP를 이용하여 위치 인식 서버와 Context Broker를 통해 상황 정보를 추론하고, 상위 레벨에서 위치 인식 및 그에 따른 유비쿼터스 컴퓨팅 서비스를 제공하기 위한 설계를 하였다. 유비쿼터스 위치 인식 응용 분야는 위치 인식 및 물리적 값에 대한 센싱을 기반으로 이용자가 환경에 적합한 서비스를 이용할 수 있도록 설계 됨으로써 단순하게 통신과 정보의 전달 뿐만 아니라 새로운 패러다임으로서의 서비스를 지원하고 있다. 이러한 서비스를 위한 기반 기술로 블루투스 이외에 Active Badge, Active Bat, RFID, RADAR, UWB 등이 다양하게 연구되고 있다. 각 시스템마다 서로 다른 기술적 메커니즘과 해결능력을

가지고 있으며 이 같은 이유로 위치 인식 시스템 구현에 있어서 변형된 컴퓨팅 서비스에 접근하고 있다.

그 중에서 본 논문에서 연구한 블루투스는 현재 모바일 폰, PDA, 노트북 등의 여러 분야에 보급되어 있고, 그로 인해서 장소와 시간을 영향을 받지 않고 관련 서비스를 적용 받을 수 있을 것이라고 기대된다. 또한, RFID, UWB, GPS등과 같은 기술들의 장점과 결합해서 좀 더 세밀하고 향상된 기술적 진보를 보임과 동시에 다양한 유비쿼터스 컴퓨팅 기술에 접목될 것이다.

5 참고문헌

- [1] Stephen S. Yan, Fariaz Karim, "Context-sensitive object request broker for ubiquitous computing environments," 2001. FTDCS 2001. Proceedings. The Eighth IEEE Workshop on Future Trends of Distributed Computing Systems, pages 34-40, November 2001
- [2] Udana Bandara, Mikio Hasegawa, Masugi Inoue, Hiroyuki Morikawa, Tomonori Aoyama. "Design and Implementation of a Bluetooth Signal Strength Based Location Sensing System," Radio and Wireless Conference, 2004 IEEE, pages 319-322, 19-22 sept. 2004
- [3] Francesca Cuome, Andrea Pugini. "A linux based Bluetooth scatternet formation kit: from design to performance results," ICPS REALMAN Workshop 2005, Santorini (Greece), July 2005
- [4] Jan Beutel, Maksim Krasnyanskiy "Linux BlueZ Howto Bluetooth protocol stack for Linux"
<http://bluez.sourceforge.net/howto/index.html> 24 August 2001
- [5] Jason Yipin Ye, "Atlantis: Location Based Services with Bluetooth", Master's thesis, 2005. Dept of Computer Science, Brown University.
- [6] Olof Rensfelt. "Lunar over Bluetooth"
- [7] P. Johannsson, R Kappor, M Gerla, M Kazantzidis, "Bluetooth an Enabler of Personal Area Networks," IEEE Network, Special of Personal Area Networks, pages 28-37, September/October 2001
- [8] 이찬행, 최창원, 이형우 "무선 랜에서의 침입 차단 시스템(Wireless-IPS) 구축에 관한 연구"
한국인터넷정보학회 학술발표대회 논문집, pages 23-26, 2005.6
- [9] <http://www.bluetooth.com/>
- [10] 유명창, "IT EXPERT 리눅스 디바이스 드라이버"
한빛미디어, 2004
- [11] Roger Weeks "LINUX UNWIRED" O'REILLY, 2005
- [12] BlueZ Bluetooth Implementation for linux,
<http://bluez.sourceforge.net>