J2ME 디바이스를 위한 써로게이트 시스템 설계

장준^{O} 최훈
충남대학교 컴퓨터 공학과
{jjang^{O}, hchoi}@ce.cnu.ac.kr

The Design of the Surrogate System for J2ME CLDC-Devices

June Jang^{O}, Hoon Choi
Department of Computer Engineering, Chungnam National University
{jjang^{O}, hchoi}@ce.cnu.ac.kr

요 약

최근 소형 디바이스들이 인터넷 같은 IP 네트워크를 통해 서로 연결되면서, 디바이스 간에 동적 분산 네트워크를 구성하는 기술인 지니가 주목 받고 있다. 하지만, 현재 지니 기술은 J2SE 자바가상머신(JVM)을 탑재한 디바이스에서만 사용할 수 있다. 본 연구에서는 J2ME CLDC를 탑재한 소형 디바이스가 지니 서비스를 사용할 수 있도록, 지니 네트워크에서 제시한 지니 써로게이트 구조를 이용하여 지니 써로게이트 시스템을 설계하였다.

1. 서론

최근 컴퓨터 시스템은 인터넷 같은 IP 네트워크 통해 소형 디바이스들이 통합되면서, 하드웨어나 소프트웨어 컴포넌트들을 네트워크에서 객체 형태로 사용하기 위한 기술들이 개발되었다 [1]. 특히, 썬 마이크로시스템즈(Sun Microsystems)의 지니(Jini)가 디바이스의 소프트웨어나 하드웨어 컴포넌트 객체를 인위적으로 조작할 필요 없이 네트워크 상에서 플러그 앤 플레이(plug and play) 형태로 사용할 수 있게 한다는 장점으로 주목을 받고 있다.

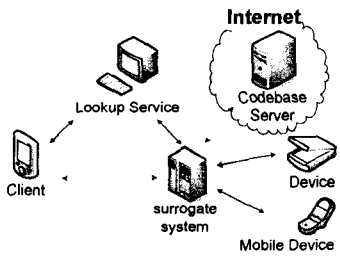
하드웨어나 소프트웨어 컴포넌트가 지니 서비스를 사용하기 위해서는 지니 디스커버리(discovery)와 조인(join) 프로토콜을 수행해야 하며, 자바언어로 구현된 실행 클래스를 다운로드하여 실행해야 한다. 하지만, 이런 기능을 지원하기 위해 디바이스는 데스크탑 정도의 컴퓨팅 능력을 가져야 하기 때문에, 지니 기술은 낮은 계산능력 및 적은 메모리를 가지고 있는 소형 디바이스에는 부적합하다 [2,3].

[4,5,6] 에서는 Non-Java 디바이스가 지니 서비스를 사용할 수 있는 방법을 설계 구현하였다. 하지만 이 방법은 다양한 플랫폼을 탑재한 디바이스를 위해 설계되었기 때문에 특정 플랫폼의 장점을 살릴 수 없고, 구현이 어렵다는 단점을 가진다.

본 연구에서는 J2ME CLDC(Java2 Platform, Micro Edition Connected Limited Device Configuration)를 탑재한 디바이스가 지니 네트워크를 사용할 수 있는 방안에 대해 제시하고, 그 디바이스를 위한 써로게이트 시스템(surrogate system)을 설계하였다.

2. 지니 써로게이트 시스템

지니 커뮤니티에서는 소형 디바이스에서 지니가 갖는 문제점을 해결하기 위해 지니 써로게이트 시스템을 제시하였다 [4]. 지니 써로게이트 시스템은 실행자원과 네트워크 연결이 한정되어 있는 소형 디바이스를 위해 하드웨어나 소프트웨어 컴포넌트의 코드를 다운로드하는 기술을 중심으로 개발되었다.



[그림1] 지니 써로게이트 시스템

지니 써로게이트 시스템은 [그림1]과 같이 소형 디바이스나 Non-Java 디바이스 같이 직접적으로 지니 네트워크에 참여할 수 없는 디바이스들을 대신하여 지니 프로토콜을 수행함으로써 디바이스가 지니 네트워크에 참여할 수 있도록 해주며, 또한, 이 시스템은 지니 클라이언트나 서비스들이 써로게이트 시스템을 통하여 연결된 디바이스인지, 지니 기능이 내장된 장치인지 알 수 없도록 지니 네트워크에서 디바이스들에게 투명성(transparence)을 제공한다.

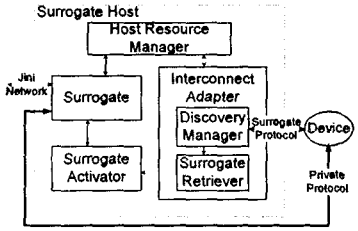
지니 써로게이트 시스템의 핵심 요소는 써로게이트 프로토콜과 써로게이트 호스트이다.

써로게이트 호스트는 컴포넌트의 실행을 위한 자바 응용 환경을 제공하는 장비로서, 지니 네트워크에 직접적으로 연결 할 수 없는 소형 디바이스를 대신할 서비스 객체인 써로게이트 객체를 다운받아 이를 동적으로 실행한다.

써로게이트 호스트는 [그림2]처럼 Interconnect Adapter 모듈, Host Resource Manager 모듈, Surrogate Activator 모듈로 구성된다 [4]. Interconnect Adapter 모듈은 디바이스를 찾고,

* 본 논문은 한국과학재단이 지정한 지역협력연구센터(RRC)인 충남대학교 소프트웨어연구센터의 지원으로 수행된 과제의 결과입니다.

디바이스와의 연결여부를 감시한다. Host Resource Manager 모듈은 써로게이트 호스트의 리소스를 제어한다. Surrogate Activator는 소형 디바이스에서 써로게이트 호스트로 다운로드된 써로게이트 객체를 활성화 시킨다.

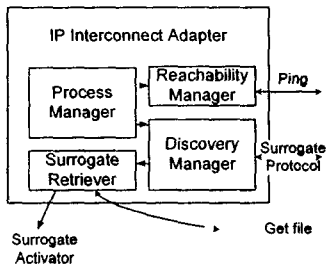


[그림2] 지니 써로게이트 호스트 구조

써로게이트 프로토콜은 써로게이트 호스트와 디바이스가 서로를 찾고, 둘간의 접속 상태를 제어하는 프로토콜이다. 디바이스와 써로게이트 호스트 간에는 다양한 네트워크 연결 방식이 있는데, 지니 커뮤니티에서는 IP, USB, 블루투스, IEEE1394, 스마트카드 네트워크 환경을 위한 다양한 써로게이트 프로토콜 방법을 연구하고 있다.

3. IP 기반의 지니 써로게이트 호스트

지니 커뮤니티에서는 지니 써로게이트 호스트의 모듈 중 써로게이트 객체를 다운로드하고, 써로게이트 호스트와 디바이스와의 연결 상태를 감시하는 IP Interconnect Adapter 모듈을 설계 구현하였다 [5].



[그림3] IP Interconnect Adapter

IP Interconnect Adapter는 [그림3]과 같이 Process Manager, Reachability Manager, Discovery Manager, Surrogate Retriever로 구성된다.

Process Manager는 써로게이트 프로토콜 기능을 담당하는 프로세스를 생성하고, 생성된 프로세스를 유지한다. 또한, Reachability Manager와 Discovery Manager도 관리한다. Reachability Manager는 써로게이트 호스트와 디바이스의 연결 상태를 관리한다. Discovery Manager는 써로게이트 호스트에서 써로게이트 프로토콜을 이용하여 디바이스의 존재를 찾거나, 써로게이트 호스트 자신의 존재를 디바이스들에게 알리는 역할을 담당한다. Surrogate Retriever는 써로게이트 객체가 저장된 코드베이스의 URL을 이용하여 써로게이트 객체를 다운로드하는 역할을 담당한다.

4. J2ME CLDC용 디바이스를 위한 써로게이트 시스템 설계

디바이스와 써로게이트 객체 사이의 통신 방법인 사설 프로토콜(private protocol)은 각 디바이스의 플랫폼 종류에 의존한다.

일단 써로게이트 객체가 써로게이트 호스트로 다운로드되면, 사설 프로토콜을 이용하여 디바이스와 정보를 교환하게 된다. 사설 프로토콜은 디바이스의 프로토콜이나 써로게이트 프로토콜 방식에 따라 구현에 큰 변화가 일어날 수 있기 때문에, 지니 커뮤니티에서는 따로 정의하고 설계하지는 않고 있다. 그래서, 개발자는 디바이스와 써로게이트 객체를 위한 정보 교환 방법을 결정해야 하고, 써로게이트 객체와 지니 네트워크 간의 정보교환 방법을 정의 구현하여야 한다.

지니 커뮤니티에서 제안한 IP기반의 써로게이트 호스트는 J2SE(Java2 Platform, Standard Edition)환경에서 설계 되었다. 본 연구에서는 소형 디바이스가 J2ME CLDC를 탑재하였을 경우, 써로게이트 호스트와 디바이스 간의 정보 교환 방법 및 써로게이트 객체를 설계하고 모듈화하였다. 써로게이트 객체 구현에 필요한 공통된 부분을 모듈화하여 정의하였기 때문에 개발자는 써로게이트 객체를 쉽게 구현할 수 있다.

4.1 J2ME CLDC용 디바이스를 위한 써로게이트 객체 설계

써로게이트 객체는 다음과 같은 요구사항이 필요하다.

(가) 디바이스와 써로게이트 객체 사이의 정보 교환은 사설 프로토콜을 통해 제어할 수 있어야 한다.

디바이스와 써로게이트 객체는 정보를 교환을 할 때 서로의 통신 방식을 알고 있어야 한다. 또한, 디바이스와 써로게이트 객체 사이의 통신이 실패하였을 경우, 이를 Surrogate Activator 모듈과 IP Interconnect Adapter 모듈로 알려주어야 한다.

(나) 써로게이트 객체의 지니 네트워크 연결이 가능해야 한다.

써로게이트 객체는 지니 네트워크에 참여 할 수 있어야 한다. 그러므로 써로게이트 객체는 지니 라이브러리를 이용하여 이를 활성화 시킬 수 있어야 한다.

J2ME CLDC용 디바이스와 써로게이트 객체 사이의 통신방식은 신뢰성 있는 통신을 위해 소켓을 이용할 수 있다.

```
interface Serverinit{
  ServerSocket ServerStart(int port){}
    throw IOException
  void SendStream(String output){}
    throw IOException
  String ReceiveStream(){}
    throw IOException
  int ServerClose() throw IOException
}
```

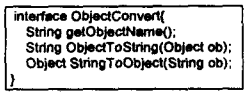
```
interface Clientinit{
  StreamConnection ClientConnector
    (int url, int flag, boolen b){}
    throw IOException
  void SendStream(String output){}
    throw IOException
  String ReceiveStream(){}
    throw IOException
  int ClientClose() throw IOException
}
```

[그림4] Serverinit과 Clientinit interface

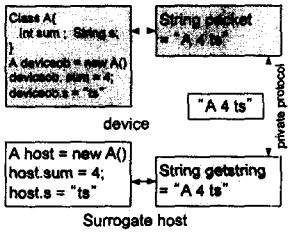
써로게이트 객체와 디바이스의 통신을 담당하는 모듈은 [그림4]와 같은 인터페이스를 구현한다. Serverinit는 써로게이트 객체에서, Clientinit는 J2ME CLCD용 디바이스에서 각각 구현된다.

J2SE에서 다른 JVM으로 객체를 이동하기 위해서 객체의 변환이 이루어진다. 변환된 객체는 다른 JVM으로 전송되고, 일련의 바이트로부터 객체의 상태가 재구성된다. 특별한 클래스가 필요하다면 객체의 코드베이스 URL에 접근하여 객체의 클래스 파일을 다운로드하고 객체를 사용하려는

목적지의 JVM에 로드된다. 하지만 J2ME CLDC는 리플렉션, 객체직렬화, 객체 변환 기능과 사용자 클래스 로더가 없다. 그래서 디바이스와 써로게이트 객체 사이에는 소켓을 통한 string, integer, float, boolean 같은 단순한 자료형 만을 전달할 수 있다.



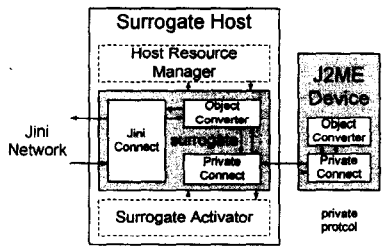
[그림5] ObjectConvert interface



[그림6] 두 디바이스 간의 객체의 정보 전달

본 연구에서 설계한 지니 써로게이트 객체 구조에서는 객체의 정보를 다른 디바이스에 보내기 위해서 전달되는 객체는 [그림5]의 ObjectConvert 인터페이스를 구현한다. [그림6]은 integer와 string 정보를 가진 디바이스의 객체 deviceob를 써로게이트 호스트로 전달하는 과정이다. 디바이스는 deviceob에 구현된 ObjectConvert를 이용해 클래스 이름 " A" 와 interger 값 " 4" , string 값 " ts" 를 " A 4 ts" 로 변환하고, 소켓을 통해 써로게이트 호스트로 이동시킨다. 써로게이트 호스트는 객체 전달 요청이 받으면, 패킷의 클래스 이름 " A" 를 이용해, 나머지 정보와 함께 객체 host를 활성화 시킨다. 결론적으로 디바이스의 deviceob 객체와 같은 정보를 가진 객체 host가 써로게이트에 생성된다.

4.2 J2ME CLDC용 써로게이트 객체 구조 및 시스템 동작과정



[그림7] J2ME CLDC용 써로게이트 객체의 구조

[그림7]은 J2ME CLDC용 디바이스를 위한 써로게이트 객체의 구조이다. 써로게이트 객체는 Private Connect 모듈, Object Converter 모듈, Jini Connect 모듈로 구성된다.

Private Connect 모듈은 활성화된 써로게이트 객체와 J2ME CLDC 디바이스의 연결 중에 정보를 제어한다. 이 모듈은 정보 교환을 위해 소켓을 이용하여 통신한다.

Object Converter 모듈은 다른 디바이스로 전달할 객체를 문자열로 변환한다. 이때 전달될 객체는 개발자가 string형으로 변환하는 정보를 제공해야 한다.

Jini Connect 모듈은 지니 프로토콜을 대행해주고, 써로게이트 객체의 정보(지니 프록시)를 일반적인 지니의 룩업서비스로 전달한다. 이 모듈은 J2ME CLDC 디바이스가 지니 네트워크에서 디바이스에 대한 투명성을 제공해준다.

J2ME 디바이스가 제시한 써로게이트를 통해 지니 서비스를 사용하는 동작과정은 다음과 같다.

가. 처음 활성화된 J2ME CLDC 디바이스는 써로게이트 프로토콜을 이용하여 써로게이트 호스트를 찾아 써로게이트 호스트로 써로게이트 객체를 전달한다.

나. 써로게이트 호스트는 써로게이트 객체를 활성화 시킨다.

다. 만약 디바이스가 지니 서비스인 경우, 써로게이트 객체는 지니 네트워크에서 룩업서비스를 찾아 자신의 정보가 포함된 객체(지니 프록시 객체)를 업로드하고, 디바이스를 대신하여 지니 네트워크와 통신한다.

라. 디바이스가 지니 클라이언트인 경우, 써로게이트 객체는 룩업서비스에서 원하는 지니 서비스를 찾아, 통신한 후 결과를 디바이스에 전달한다.

마. 디바이스가 동작하지 않거나 더 이상 지니 서비스를 사용할 필요가 없는 경우, 써로게이트 호스트는 다운로드된 써로게이트 객체를 비활성화 시키고 이를 삭제한다.

5. 결론

써로게이트 시스템은 지니 네트워크로 응용 프로그램과 디바이스들을 통합하기 위한 개념이다.

J2ME CLDC용 디바이스를 위한 써로게이트 시스템은 J2ME CLDC를 탑재한 디바이스 생산자와 개발자가 생산하는 제품을 단순화하고 강력하게 해주는 새로운 틀을 제공해주고, 디바이스가 지니 서비스를 사용할 수 있게 한다.

본 연구에서는 J2ME CLDC 디바이스를 위한 써로게이트 시스템의 써로게이트 객체의 구조와 private 프로토콜에 대해 설계하였다. 향후 J2ME CLDC 에 최적화된 써로게이트 프로토콜을 설계하고 써로게이트 시스템을 구현할 예정이다

6. 참고 문헌

[1] Mark Weiser, " Some Computer Science Issue in Ubiquitous Computing" Communication of the ACM, July 1993.

[2] Sun Microsystems Inc., Jini Connection Technology, http://www.sun.com/jini/

[3] Sun Microsystems Inc,. " Jini Architecture Specification" , 1999.

[4] Jini Community, " Jini Technology Surrogate Architecture Specification" , 2000.

[5] Jini Community, " Jini Technology IP Interconnect Specification" , 2000

[6] 최현석, " Non-Java 장치를 지원하기 위한 Jini 써로게이트 시스템의 설계 및 구현" , 광운대, 한국정보과학회 논문지 v.029 n.006 pp.685-695 1229-7720, 2002.12