

# 개방형 서비스 게이트웨이(OSGi) 환경에서 Jini 서비스의 통합

양 동 준<sup>o</sup>, 김 태 형  
한양대학교 컴퓨터공학과  
(djyang, tkim}@cse.hanyang.ac.kr

## Integration of Jini Services to Open Service Gateway initiative(OSGi) Platform

Dong-Joon Yang<sup>o</sup>, Tae-Hyung Kim  
Dept. of Computer Science & Engineering, Hanyang University

### 요 약

홈 네트워킹에 대한 관심이 높아지면서 이를 실현하기 위해 홈 네트워크에 연결되는 기기들간에 상호 운영성을 보장하는 미들웨어 기술에 관련된 솔루션이 대두되고 있다. 본 논문에서는 이에 따른 미들웨어 기술에 관련된 솔루션 중 OSGi와 Jini간의 연계 가능성을 살펴보고자 한다. 먼저 OSGi의 개념과 OSGi의 Bundle 개념을 살펴보고 Jini의 개념과 OSGi의 Bundle로써 Jini service를 구현하기 위한 방법에 대해 고찰하고자 한다.

### 1 서 론

디지털과 인터넷의 빠른 확산으로 일반 가정에서 인터넷에 대한 요구가 확산되고 있다. 최근 들어 유무선 홈 네트워크와 인터넷을 연동하여 홈 씨어터 서비스, 원격 가전 기기 및 정보 기기를 제어하거나 활용할 수 있는 인터넷 정보 가전에 대한 수요가 빠르게 증가되고 있다. 이에 따라 여러 가지 홈 네트워킹을 위한 미들웨어 기술이 개발되고 있고 다양한 연구가 시도되고 있다. 이러한 홈 네트워킹을 위한 미들웨어 기술에 관련된 솔루션 중 OSGi와 Jini간의 상호 연계 가능성을 살펴보고자 함이 이 논문의 목적이다

### 2 홈 네트워킹 관련 미들웨어

#### 2.1 OSGi

1999년에 홈 네트워킹에 관한 프로토콜에 관심을 보이는 세계 여러 기업들이 모여 조직된 비영리 단체인 OSGi(Open Service Gateway initiative)는 Service Gateway에 대한 새로운 Specification을 발표하였다. 이는 자바의 플랫폼 독립성과 실행코드의 네트워크 이동성을 이용하여 소용량 메모리 디바이스를 위한 동적인 서비스의 제공을 목표로 제정된 표준이다. 이에 Sun, IBM등의 기업들을 중심으로 한 홈 네트워킹 Service Gateway에 대한 표준이다. Service Gateway는 외부 네트워크로부터 가정 내의 내부 네트워크로 서비스를 제공하는 역할을 담당하게 된다. 즉 OSGi는 외부와 내부를 잇는 다양한 네트워크 프로토콜간의 브릿지 역할을 담당한다.[1][4]



그림 1 OSGi Service Gateway

그림 1은 OSGi가 외부 네트워크망과 내부 네트워크에 사용되는 다양한 프로토콜간의 브릿지 역할을 하며 서비스의 Delivery 기능을 담당하는 위치를 보여준다. Service Gateway는 외부 네트워크에서 제공되는 서비스를 관리해주는 역할을 하고 있다. OSGi가 가지는 Service Gateway는 여러 가지 역할을 수행할 수 있다. 외부와 내부에 대한 프로토콜 변환, 라우터, 트래픽 처리, 보안, 에너지 관리 등의 역할을 수행할 수 있다.

OSGi를 구성하는 3가지 엔티티는 서비스(Service), 번들(Bundle), 프레임워크(Framework)이다. 서비스는 특정 기능을 수행하는 자바 인터페이스와 실제 구현객체로 미리 정의된 서비스 인터페이스를 통해 접근이 가능한 컴포넌트이다. 하나의 애플리케이션은 여러 개의 서비스의 협동작업을 통해 구성되고, 런타임시에 필요한 서비스를 요청할 수도 있다. 번들은 서비스를 제공하기 위한 기능적 배포단위이다. 프레임워크는 번들의 라이프 사이클을

관리하는 번들 실행환경이다.

번들은 여러 서비스의 구현을 하나의 패키지이다. 이러한 번들의 라이프 사이클은 프레임워크에서 관리한다. 번들은 JAR 파일의 형태로 존재하며 JAR 파일에는 하나 이상의 서비스의 구현체와 리소스 파일 및 매니페스트(Manifest) 파일을 포함한다. 매니페스트 파일은 해당 번들이 제공하는 서비스와 이 번들의 실행을 위해 필요한 다른 서비스를 명시하고 있다. 번들은 프레임워크 내에서 시작(Start)과 종료(Stop)에 의해 수행되거나 정지하며 DLL(Shared Library)과 같은 기능을 수행한다.

### 2.2 Jini

Jini는 썬마이크로시스템즈사가 창안하여 새로운 제어 모델을 개발하고 이를 홈 네트워크의 대표적인 미들웨어 솔루션으로 활성화하려는 움직임으로 시작되었다. Jini의 기본적인 철학은 단순하며, 신뢰성을 확보할 수 있고, 보다 나은 제어구조로 발전하는데 확장성을 부여하도록 하자는 것이며, 별도의 관리없이 동작하도록 하자는 데에 초점을 두었다. 이와 같은 구조는 자바를 기반으로 하는 분산 네트워크 접속 기술을 이용하여 가능하게 하였다.

Jini의 특징은 네트워크 상에서 플러그 앤 플레이 기능을 제공하며, 소프트웨어나 하드웨어, 연산능력, 스토리지, 사용자 등 모든 요소를 서비스로 처리하는 "서비스 기반"구조를 활용하고 있다. 또한 JVM(Java Virtual Machine)을 사용하므로 하드웨어에 독립적이며, 네트워크의 구성이 단순하고 확장성을 제공해준다. Jini는 RMI 통신 방법을 기반으로 하고있다.[2][3]

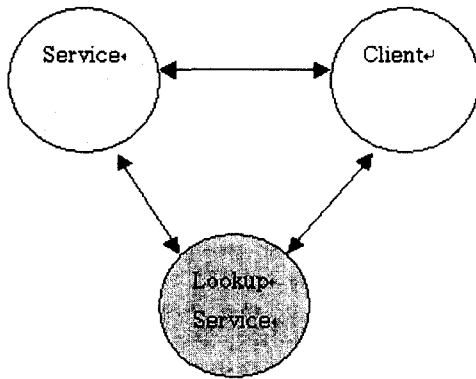


그림 2 Jini 시스템 구성도

그림 2는 Jini 시스템 구성도를 보여주고 있다. Service는 Client에게 Service를 제공하고자 하는 시스템이며, Client는 Service를 이용하고자 하는 시스템이고 Lookup Service는 Service와 Client사이에서 서로 중계 역할을 수행하는 시스템이다. Jini 시스템에서 Client가 Service를 사용하고자 하는 과정은 Service가 Jini 시스템에 자신을 등록하는 일련의 과정(Jini-Discovery)을 행한 후, Service를 Lookup 서비스에 join 하는 과정(Jini-join)이 있는데 이 과정은 Service가 네트워크 상에 새로 접속되었을 때 발생하며 Lookup Server에 자신의 Attribute를 올리는 과정(Jini-Lookup)이다. Service를 사

용하기 위해서는 Lookup Service에 등록된 Service 개체가 Client로 다운로드 되는 과정이 있으며, 이 과정 이후에 Client와 Service 제공자 사이의 서비스가 실행되는데, 이 과정은 Client의 요청에 의해 발생한다.

그림 3은 Jini를 이용한 가정의 홈 네트워크 구성의 한

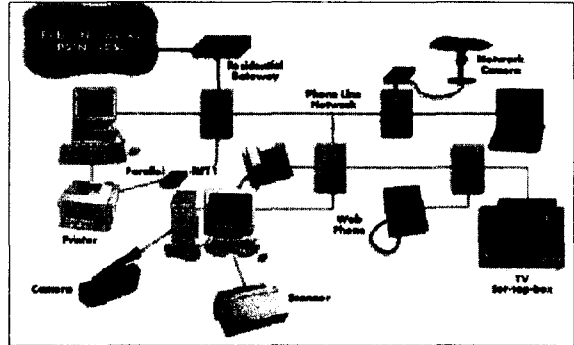


그림 3 Jini를 이용한 가정의 홈 네트워크 구성의 예

예이다. Jini는 전자제품이나 네트워크상의 서비스를 모두 통합할 수 있는 기반을 제공해주며, 네트워크 집단에 포함되거나 삭제될 때 매우 쉽게 처리가 되며, 분산 컴퓨팅에 유리한 구조이다. 일반적으로 알려진 Jini 기술은 네트워크에 신규로 소속되거나 분리되는 경우 자동으로 인식하여 처리될 수 있는 기술이다.

### 3 OSGi의 Jini Bundle

#### 3.1 OSGi와 Jini Bundle 역할

OSGi는 Service Gateway를 중심으로 다양한 네트워크를 구성할 수 있다. 그림 1처럼 외부에서 내부 네트워크에 접속하거나 내부에서 외부로 접속할 때 OSGi를 중심으로 하는 브릿지 역할을 OSGi가 담당하게 된다. 내부 네트워크를 Jini로 구성할 때 OSGi는 Jini Network에 대한 접근을 허용하기 위해 OSGi와 Jini Network간에 매개체 역할을 하는 것이 필요하다. 이러한 매개체 역할을 하는 것이 Jini Bundle이다.

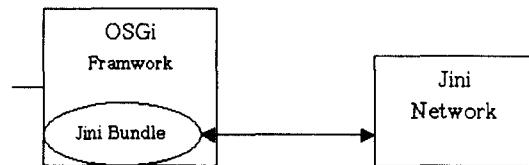


그림 4 OSGi와 Jini Bundle과의 관계

그림 4는 OSGi가 Gateway 역할을 담당할 때 내부적인 네트워크는 Jini가 담당하게 된다. 이러한 OSGi와 Jini간에 서로 연결을 담당하게 되는 것이 Jini Bundle이다. Jini는 RMI 통신 방법을 기반으로 하므로 OSGi에는 내부적으로 RMI에 대한 패키지가 포함되거나 상주되어야 한다. 이 구성은 홈 네트워크 상에서 OSGi가 브릿지,

라우터 역할을 담당하고 가정 내에서는 Jini Network가 구성되어있는 환경이다.

### 3.2 Jini Bundle의 구현

OSGi의 Jini Bundle 구현은 OSGi가 제공하는 OSGi Specification에서 나타내고 있는 번들의 내용으로 구성한다.[4]

기본적인 번들의 구성은 start(시작), stop(정지)에 대한 메소드로 구현한다. 번들의 구성에 대한 내용은 번들 Activator가 담당하게 된다. 번들 Activator는 번들의 시작시 OSGi 프레임워크에 번들을 등록하며, 번들의 정지시 OSGi 프레임워크시 번들의 등록을 삭제하는 역할을 담당한다.

```

.....
package jini.impl;

import org.osgi.framework.*;
import java.util.*;
import jini.service.*;

public class Activator implements BundleActivator {

    private ServiceRegistration reg = null;

    public void start(BundleContext ctx) throws BundleException {
        System.out.println("starting...");

        JiniService svc = new JiniServiceImpl();

        Properties props = new Properties();
        props.put("description", "Jini Service");
        reg = ctx.registerService(
            "jini.service.jiniService", svc, props);
    }

    public void stop(BundleContext ctx) throws BundleException {
        System.out.println("stopping...");
        if (reg != null) reg.unregister();
    }
}
.....

```

리스트 1 JiniBundleActivator Interface의 예

Jini는 Service 제공을 처리하기 위해 기본적인 Lookup 서비스가 가동된다. JiniBundle은 Lookup서비스를 찾는다.

```

LookupDiscovery discovery =
    new LookupDiscovery(LookupDiscovery.ALL_GROUPS);
discovery.addDiscoveryListener(myListener);
public void discovered(DiscoveryEvent e) {
    ServiceRegistrar[] registrars = evt.getRegistrars();
}

```

리스트 2 JiniBundle에서 지니 룩업서비스를 찾는 예(Discovery)

이제 JiniBundle이 Lookup서비스를 찾아 Jini Service를 등록하는 과정이 필요하다. JiniBundle에서 Jini 서비스는 Jini 룩업서비스에 자신의 프로록시를 등록함으로써 지니 클라이언트가 지니 룩업서비스에 자신이 찾고자하는 지니 서비스를 요청할 경우 이용될 수 있다. 이를 등록하는 과정은 다음과 같다.

```

ServiceRegistrar registrar = .....;
ServiceID id = null; // lookup service에서 할당된다.
Entry entry = null; // service의 부가정보
MyServiceInterface proxy = createMyServiceInterfaceInstance();
ServiceItem item = new ServiceItem(id, proxy, entry);
long leaseTime = 1*60*1000 // milliseconds
ServiceRegistration registration = registrar.register(item, leaseTime);

```

리스트 3 JiniService를 Lookup서비스에 등록하는 JiniBundle Service interface의 예

이로써 JiniBundle을 이용하여 OSGi와 Jini network에 서로간 서비스 운영이 가능하다. 그림 5는 OSGi와 Jini 기술을 이용한 네트워크 구성도이다. JiniBundle을 이용한 Jini network에 대한 서비스를 사용할 수 있다.

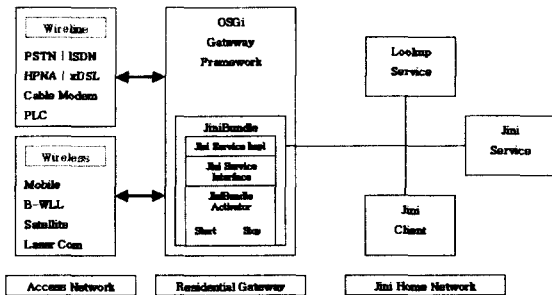


그림 5 OSGi Gateway를 이용한 Jini Home Network 구성도

### 4 결론

본 논문에서는 OSGi에서의 OSGi Gateway를 이용하여 Jini Network를 구성하는데 있어 OSGi의 Jini Bundle을 사용하여 미들웨어 솔루션 Jini를 통합하는 방법에 대해서 살펴보았다. OSGi는 다양한 미들웨어를 수용하는데 있어 각각의 미들웨어 Bundle을 이용하여 서로간에 네트워크 통신을 할 수 있는 방법을 제시하였다. 이러한 미들웨어 Bundle을 이용하여 분산형 연산환경과 서비스를 지원받을 수 있으며, 데이터 네트워크와 제어 네트워크를 쉽게 통합할 수 있고, 여러 가지 미들웨어 솔루션으로 기기 간의 다양한 데이터를 통합하기 쉽다.

향후 연구 방향으로 UPnP, HAVi와 같은 다른 미들웨어 솔루션을 OSGi 환경에서 제공하는 연구가 필요하다. 이렇게 함으로써 다양한 형태로 존재하는 미들웨어 표준을 OSGi 환경에서 통합하여 제공하는 것이 가능하다.

### 5 참고문헌

[1] OSGi 홈페이지. <http://www.osgi.org>.  
 [2] Jini 홈페이지. <http://www.javasoft.com/products/jini>  
 [3] Henry Wong, Scott Oaks, *Jini in a Nutshell*, O'Reilly, 1999.  
 [4] Open Service Gateway Initiative. "OSGi Service Gateway Specification 1.0". 5. 2000.