

논문 2012-49CI-2-10

# 스마트 환경과 협업하는 중재 로봇을 위한 통신 프레임워크

( A Communication Framework for the Robotic Mediator collaborating with Smart Environments )

서영호\*, 이강우\*, 조은선\*\*

( Young-Ho Suh, Kang-Woo Lee, and Eun-Sun Cho )

## 요약

이형의 단말 플랫폼간의 유연하고 효율적인 통신은 로봇과 스마트 환경이 함께 증강된 기능을 구성하여 협업하기 위해 매우 중요하다. 기존의 네트워크 로봇 소프트웨어 플랫폼에서는 CORBA와 같은 범용의 통신 미들웨어를 사용하는 경우가 많았다. 그러나 범용의 통신 미들웨어는 너무 무겁고 사용이 복잡할 뿐만 아니라, 네트워크 로봇에 특화된 기능을 제공하지 못한다. 따라서 본 논문에서는 TCP에 기반 한 원격객체함수 호출방식의 새로운 통신 프레임워크를 제안한다. 전송계층과 원격객체 호출 계층으로 분리하여, 각 계층별로 다양한 특화 기능을 제공함으로써 스마트 환경에서 로봇과 다양한 단말이 효과적으로 통신할 수 있도록 한다.

## Abstract

In order to enable network robots to effectively collaborate with smart environments, it is crucial that various types of heterogeneous device platforms can communicate each other in flexible and efficient manners. Most of existing software platforms for network robots adopted general-purpose middlewares such as CORBA, which are not suitable to the communication between robots and smart environments due to their heavy size and complexity. Moreover, they do not provide network robot-specific features. Therefore, we propose a new TCP-based Remote Method Invocation framework. We divide the middleware into two layers i. e. transport layers and rmi layers and provide key features to each layer so that network robots can effectively communicate with various devices in smart environments.

**Keywords :** Smart Environment, Robotic Mediator, Network Robot, Communication Framework, Protocol

## I. 서론

스마트 환경은 물리 공간 내에 내재된 다양한 센서, 액츄에이터, 디지털 기기, 레거시 시스템 등과 같은 유

비쿼터스 장치들이 서로 연동되어 가상공간을 구성함으로써 거주자의 안락한 생활을 위한 서비스를 제공하는 기술을 말한다<sup>[1~2]</sup>. 스마트 환경 시스템에서 사용자의 상황이나 의도를 알아내고, 공간 내 가용한 자원들을 조합하여 재구성한 협업 서비스를 제공해주기 위해서는 개념적으로 중재자가 필요하다. Mark Weiser<sup>[3]</sup>가 제시한 비전에 따라, 초창기의 시스템에서는 중재자가 환경 내에 분산되고 내재되어 명시적으로 드러나지 않는 경우가 많았다. 그러나 실험결과에 따르면 80% 정도의 피험자들이 보이지 않는 막연한 존재와의 인터랙션이나 명시적인 명령에 의하지 않은 서비스 실행에 대해 거부감을 나타냈다고 한다<sup>[4]</sup>.

\* 정회원, 한국전자통신연구원 로봇/인지시스템 연구부 (Robot/Cognitive System Research Department, Electronics and Telecommunication Research Institute)

\*\* 정회원-교신저자, 충남대학교 컴퓨터공학과 (Department of Computer Science and Engineering, Chungnam National University)

※ 본 연구는 지식경제부의 “미래형로봇컴퓨터 기술개발 과제(과제번호 2010-ZC1140)” 지원을 받아 수행되었음.

접수일자: 2011년2월16일, 수정완료일: 2012년3월5일

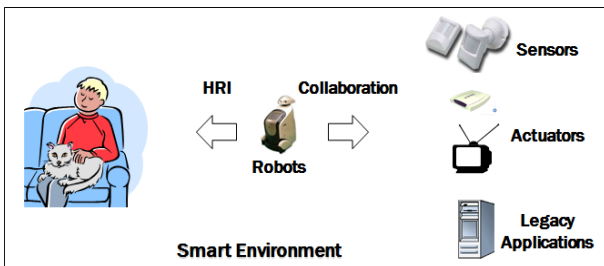


그림 1. 스마트 환경과 협업하는 중재 로봇  
Fig. 1. A robotic mediator collaborating with smart environments.

이에 따라 효과적인 중재자 인터페이스를 찾기 위해, 다양한 단말이나 인터페이스 기술들(스마트 단말, 웨어러블 디바이스, 텐저블 인터페이스, 공간 증강현실 등)에 대해 타당성 연구가 있어 왔지만, 그 자체만으로는 아직 킬러 애플리케이션이 되지는 못하고 있는 실정이다<sup>[5~7]</sup>. 최근 들어 몇몇 연구들<sup>[5, 8~10]</sup>에 의해 중재 로봇의 개념이 제안되면서 중재자 인터페이스 기술의 새로운 대안으로 주목을 받고 있다.

이들 연구에서는 네트워크 기능이 탑재된 로봇들<sup>[11]</sup>이 스마트 환경과 유기적으로 연동하여, 음성, 대화, 제스처 등과 같은 멀티모달 인터페이스를 통해 사용자의 도를 인식, 전달한다.

그림 1과 같이, 중재 기능을 위해 사용되는 네트워크 로봇은 스마트 환경의 이질적인 시스템 및 다양한 외부 장치들과 통신하고 협업해야 한다. 따라서, 로봇이 기반하고 있는 소프트웨어 프레임워크<sup>[11]</sup>에서는 (1) 이형의 통신 프로토콜/운영체제/프로그래밍 언어 간의 상호 운용 방식, (2) 이기종 장치들 간의 연동을 위한 장치 추상화 모델, (3) 용이한 프로그램 개발을 위한 응용 프레임워크 등을 제공할 필요가 있다.

특히 이 중에서 첫 번째인 이형의 통신 프로토콜/운영체제/프로그래밍 언어 간의 상호 운용 방안은 로봇과 스마트 환경이 함께 증강된 기능들을 구성하여 협업하는 데 있어 매우 중요한 기본이 되는 기능이다. 기존에는 서로 다른 프로토콜의 중재를 위해 CORBA 등의 범용의 통신 미들웨어를 사용하는 경우가 많이 있었으나, 너무 무겁고 사용이 복잡할 뿐만 아니라 로봇과 스마트 환경의 연동을 위해 필요한 기능을 효과적으로 제공하지 못한다는 단점이 있다. 일례로, 네트워크 로봇 소프트웨어 프레임워크 중 하나인 URS<sup>[22]</sup>는 시멘틱 웹 기반의 협업을 통해 로봇 서비스의 품질을 높이는 데에 성공하였으나, 체계적인 상호 통신 운용 방안이 지원되

지 않아, 로봇 제어를 위한 응답성, 대용량 blob 전송, 멀티미디어 스트리밍과 같은 통신 성능에 문제를 안게 되었다.

본 논문에서는 이러한 문제를 해결하기 위해서, TCP에 기반한 원격객체 함수 호출방식의 새로운 통신 프레임워크인 PLANET을 제안한다. PLANET은 로봇과 스마트 환경 연동에 특화된 통신 프레임워크로서, 로봇이 가진 저수준의 센서로부터 스마트 환경이 제공하는 고수준의 레거시 시스템까지, 다양한 스펙트럼의 데이터 종류, 제어 방식, 추상화 수준 등을 효과적으로 수용할 수 있는 유연한 통신 및 장치 모델을 제공 한다.

본 논문의 구성은 다음과 같다. II장에서는 관련연구를 간단히 소개하고, III장은 본론으로서 이형의 통신 방식과 이기종간의 제어를 원활히 지원하기 위한 통신, 장치, 서비스 프레임워크에 대해 기술한다. IV장에서 시스템 구현과 구현된 시스템에 대한 실험 결과를 보여준다. 마지막으로 V장에서 결론을 맺는다.

## II. 관련 연구

네트워크 로봇 소프트웨어 플랫폼은 관점에 따라 크게 두 가지로 분류할 수 있다. 먼저, 로봇 소프트웨어 플랫폼인<sup>[12~18]</sup> Orca<sup>[12]</sup>, OROCOS<sup>[13]</sup>, MARIE<sup>[14]</sup>, RT-Middleware<sup>[15]</sup> 등은 로봇의 입장에서 로봇의 기능/성능/서비스 확장을 위해 환경과 연동하려는 접근법에 따른다. 따라서 주로 로봇에 특화된 컴포넌트 기반의 프레임워크이기 때문에, 연동을 위해서는 외부 시스템이 모두 로봇 컴포넌트 모델을 따라야 하는 문제점이 있다. 특히 로봇 컴포넌트 단위의 저수준의 연동으로는 고수준의 다양한 협업 응용을 효과적으로 모델링하여 처리가 복잡해지고 결과적으로 좋은 성능을 얻을 수 없다. 좀 더 발전된 형태로서 인터페이스 레벨의 통신이 가능한 Miro<sup>[17]</sup> Player<sup>[18]</sup> 등도 있으나 각각 범용 미들웨어인 CORBA와 TCP 기반으로 로봇과 스마트환경의 통신에 특화된 응용에 대한 성능을 내기 어렵다. PEIS<sup>[19]</sup>나 Player 미들웨어<sup>[20]</sup>와 같이 개선된 프레임워크들도 로봇을 중심으로 한 저수준의 연동만을 지원하므로 고수준의 다양한 협업 응용을 모델링하기가 어려워진다. 예를 들어, 기존의 저수준의 컴포넌트들은 RFID 센서 값이 로봇의 물체 인식이나 네비게이션에 도움을 주는 정도의 협업에서는 문제가 없으나, 중재 로봇의 멀티모달 인터페이스를 활용한 고수준의 협업을

기술하기에는 매우 복잡한 과정이 필요하다.

두 번째로, 유비쿼터스 로봇 공간<sup>[19-22]</sup>으로 분류되는 프레임워크들은 스마트 환경 입장에서 로봇의 이동성, 인지능력, 조작능력 등과 같은 로봇 서비스를 활용하기 위해 로봇을 연동한다. 즉, 스마트 환경 내의 상황인지를 위한 서비스 프레임워크가 주를 이루며, 로봇은 그 안에서 서비스나 장치로 인식된다. 그러나 이러한 프레임워크에서는 로봇의 기능을 서비스 모델에 따라 제공해야 하므로, 로봇의 특화된 기능을 충분히 활용하기 어렵고, 로봇 자율행위를 위한 독립적인 제어 구조를 갖기가 어렵다는 문제가 있다. 이러한 예로 CAMUS<sup>[21]</sup>는 상황 인지 소프트웨어 프레임워크로서, 객체 지향형 표준 서비스 인터페이스를 통해 장치를 추상화하여, 응용 개발자가 이형의 장치들을 동일한 방식으로 연동할 수 있게 하였다. 그러나 일반적인 프레임워크에 가까우며, 협업행위 모델이나 로봇 제어 구조를 지원하지 않는다. 앞서 설명된 URS<sup>[22]</sup>에서는 물리 공간, 의미 공간, 가상 공간을 구성하고 이들 사이의 협업을 통해 로봇 서비스의 품질을 높이고자 하였다. 그러나 시멘틱 웹 기반의 상황 인지 시스템과의 연동에 중점을 두고 있기 때문에, 로봇 제어를 위한 통신 성능에 많은 문제가 있으며, 실제 서비스를 위한 이벤트 기반의 반응형(reactive) 응용 작성 또한 가능하지 않다.

### III. PLANET : 스마트환경과 로봇 통신 연동을 위한 통신 프레임워크

II장에서 살펴본 바와 같이, 기존의 연구에서는 주로 CORBA와 같은 범용의 통신 미들웨어를 사용하였다. 그러나 범용의 통신 미들웨어는 로봇과 스마트 환경의 연동을 위해 특화된 기능을 효과적으로 제공하지 못한다. 따라서 본 논문에서는 자체 개발한 범용의 통신 프

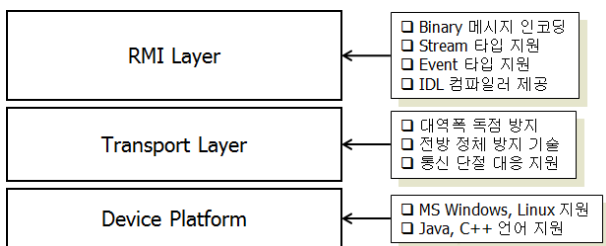


그림 2. 통신 프레임워크 계층 구조  
Fig. 2. The layered architecture of the comm. framework.

레이미워크를<sup>[22]</sup>을 확장하여 스마트 환경과 로봇 통신 연동을 위한 통신 프레임워크를 제안한다. 그림 2는 제안하는 통신 미들웨어의 계층 구조와 각 계층에서 제공하는 특화 기능들을 보여준다.

메시지 전송 계층은 다수의 통신 대상에 대한 네트워크 연결을 안정적으로 맺고, 양방향으로 메시지를 효율적으로 송수신할 수 있도록 해준다. 그리고 원격객체 호출 계층은 사용자에게 투명한 방법으로 원격 객체를 활성화시키고, 활성화된 객체의 함수를 호출하기 위한 하부구조를 제공한다. 각 계층별 특화 기능은 다음과 같다.

#### 가. 메시지 전송 계층

##### (1) 통신 단절 대응

로봇과 스마트 환경 내 장치들은 주로 불안정한 무선 네트워크 환경을 이용하기 때문에 통신 단절이 빈번하게 발생할 수 있다. 그림 3에서 연결 관리자는 연결이 맺어진 대상들과의 주기적인 heartbeat 메시지를 주고 받는다. 이러한 과정에서 통신 단절을 감지하게 되면, 이를 연결 청취자들에게 통보해준다. 또한 통신 단절된 연결 대상들에 대해 주기적으로 재 연결을 시도하여(단절이 복구되어) 다시 연결되면 연결 청취자들에게 통보해준다.

##### (2) 대역폭 독점 방지

로봇과 같이 센서와 액추에이터를 모두 갖는 장치의

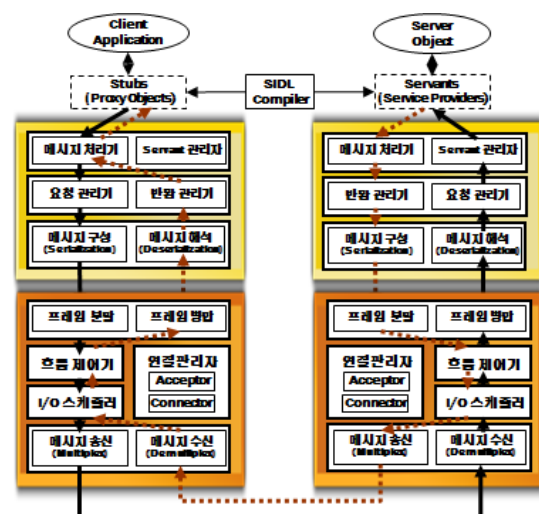


그림 3. 전송 계층과 원격 객체 함수 호출 구조  
Fig. 3. The architecture of transport layer and RMI layer.

경우, 대용량 메시지(영상, 음성 등 멀티미디어 데이터)와 간단한 메시지(모터제어)가 동시에 처리되는 경우가 많다. 이때 대용량 메시지에 의해 대역폭이 독점됨으로써, 간단한 메시지에 대한 처리가 지연되는 현상이 발생할 수 있다. 이렇게 되면 제어 응답성이 현저히 떨어지게 되는 문제가 발생한다.

따라서 그림 3의 프레임 분할/병합 과정에서 메시지를 고정 크기(16K)의 프레임으로 분할한다. 그리고 I/O 스케줄러에서 프레임들을 시분할 다중화/역다중화 함으로써 특정 메시지에 의한 대역폭 독점을 방지한다.

### (3) 전방 정체 방지

대역폭 독점 방지 기능이 메시지 송신측에서의 대역폭 점유 문제를 위한 것이었다면, 전방 정체 방지는 메시지 수신측에서의 대역폭 점유 문제에 해당한다. 로봇과 스마트 환경 내 장치들은 다양한 컴퓨팅 자원을 갖는다. 따라서 대용량의 메시지를 송수신하는 장치간의 처리능력의 차이로 인해 수신측에 정체가 발생하게 되면, 간단한 다른 메시지까지도 수신할 수 없게 된다. 이를 방지하기 위해 그림 3에서 보는 것처럼, 양측의 흐름 제어기가 메시지 송수신 과정에서의 핸드셰이크 프로토콜을 통해 흐름 제어를 한다.

## 나. 원격 객체 함수 호출 계층

### (1) 바이너리 메시지 인코딩

그림 3의 메시지 구성/해석 단계에서는 원격 객체 함수의 호출에 요청에 대한 대상 함수 이름, 인자 값, 반환 값 등을 포함한 함수 호출 정보를 직렬화/역직렬화한다. 이때 메시지의 형태는 바이너리 인코딩 방식을 따르는데, 이는 메시지의 크기를 줄여서 전송 속도를 높이고, 메시지 파싱에 따른 오버헤드를 없애기 위함이다. 또한 로봇과 스마트 환경에서 빈번한 멀티미디어 데이터의 전송에도 효과적이다.

### (2) IDL(Interface Definition Language)

플랫폼 독립적인 원격 객체 인터페이스 정의를 위해 객체 지향형 IDL과 Java와 C++ 로 바인딩되는 컴파일러를 제공한다.

그림 4와 5는 각각, IDL에서 제공하는 기본 데이터 타입과 사용자 정의 데이터 타입들로 구성된 인터페이스의 예를 보여준다.

```
interface PrimitiveInterface {
    void funcVoid();
    byte funcByte(byte v);
    int funcInt(int v);
    short funcShort(short v);
    long funcLong(long v);
    float funcFloat(float v);
    double funcDouble(double v);
    boolean funcBoolean(boolean v);
    string funcString(string v);
    binary funcBinary(binary v);
    stream funcStream(stream v);
}
```

그림 4. 기본형 데이터 타입에 의한 인터페이스 정의  
Fig. 4. An interface definition with primitive data types.

```
enum XyzEnum {
    ONE, TWO, TREE,
};

value XyzValue {
    double field1;
    string field2;
    XyzValue field3;
    XyzEnum field4;
};

event XyzEvent {
    int prop1;
    binary prop2;
    XyzValue prop3
};

exception XyzException;

interface XyzInterface {
    void funcThrows() throws XyzException;
}

interface UserDefinedInterface {
    XyzEnum funcEnum(XyzEnum v);
    XyzValue funcValue(XyzValue v);
    XyzEvent funcEvent(XyzEvent v);
    XyzInterface funcInterface(XyzInterface v);
}
```

그림 5. 사용자 정의 데이터 타입에 의한 인터페이스 정의  
Fig. 5. An interface with use-defined data types.

사용자 정의 데이터 타입 가운데 Enum, Value, Event 타입의 경우, 함수 인자나 반환 값으로 전달될 때 ‘값에 의한 전달’이 이루어진다. 그러나 XyzInterface 와 같이 Remote Reference 타입인 경우는 ‘참조에 의한 전달’ 방식으로 전달된다. 또한 사용자 정의 예외 타입을 선언 할 수 있으며, 로컬 함수에 대한 예외 처리와 동일하게, 위치 투명한 방식으로 동작하게 된다. 마지막으로 위의 그림에는 나타나지 않았지만, 임의의 타입에 대한 시퀀스 타입도 지원한다.

### (3) 스트림 타입 지원

스마트 환경에서 멀티미디어 데이터 스트리밍에 대

한 지원은 필수적이다. 이를 위해서 IDL의 기본형 데이터 타입의 하나로 stream 타입이 제공된다. 함수의 인자나 반환 값으로 stream 타입이 전달되면, 호출자와 원격의 피호출자 사이에 네트워크상의 스트림이 생성된다. 이후에 수신측에서 stream이 제공하는 read 함수를 이용하여, 원하는 만큼의 데이터를 읽어간 후에, close 함수로 스트림을 닫게 된다.

(4) 이벤트 타입 지원

로봇과 환경 내 센서 장치들은 센싱 정보를 이벤트의 형식으로 전달하는 것이 효과적이다. 따라서 다양한 센싱 정보에 대한 이벤트 타입을 쉽게 정의하고 전달할 수 있는 기능을 제공해야 한다. 이벤트는 속성들의 집합으로 구성된다. 그림 5에서 보는 것처럼, 제공하는 IDL에서는, 다양한 타입의 속성들을 갖는 이벤트를 쉽게 정의할 수 있다. 이벤트 타입이 함수 인자나 반환 값으로 전달될 되면, 수신측에서는 각 속성의 이름을 key 값으로 하는 속성 값을 얻을 수 있다.

IV. 구현 및 실험 결과

1. 성능 평가

PLANET은 현재 Java 와 C++ 두 개의 프로그래밍 언어에 대해 구현되었다. 또한 Java는 그 자체로 플랫폼 독립성을 제공하지만, C++ 는 플랫폼 독립성을 위해 ACE 프레임워크<sup>[16]</sup>을 기반으로 구현하여, Windows, Linux, WinCE 등의 다양한 플랫폼에서 동작하도록 구현하였다.

구현된 프레임워크의 성능과 기능의 올바른 동작을 확인하기 위해 주요한 기능에 대해 시험을 수행하였다. 참고로, Java의 RMI의 경우는 성능은 매우 우수하나 Java 언어만을 지원하기 때문에, 임베디드 단말이나 로봇과 같은 저수준 장치의 연동에서는 사용하지 못한다. 그러나 단순한 성능 지표서의 의미로 Java RMI를 시험에 포함시켰고, 이에 따라 PLANET과 CORBA도 모두 동일하게 Java 버전으로 시험하였다.

가. 시험 환경

시험을 위한 환경은 표 1과 같다.

표 1. 성능 평가를 위한 테스트 환경

Table 1. Test environment for performance evaluation.

server	cpu	Intel Core i5-2500 (3.3GHz, 4-core)
	ram	DDR3 8GB
	stroage	120GB SSD 1EA
	ethernet	Intel Gigabit Network Connection
	os	MS Windows7 Enterprise 64-bit
client	cpu	Intel Core i5-2500 (3.3GHz, 4-core)
	ram	DDR3 8GB
	stroage	120GB SSD 1EA
	ethernet	Intel Gigabit Network Connection
	os	MS Windows7 Enterprise 64-bit
network		LAN 100 based Ethernet

나. 메시지 동시 처리

표 2는 로봇과 스마트 환경에서 카메라 영상 이미지 크기인 320\*240 바이트의 바이너리 메시지 전송에 대한 왕복 시간을 비교한 결과를 보여준다. 쓰레드 개수를 늘려가면서 동시처리 능력도 비교해 보았다. 그림 6에서 보는 것처럼, PLANET과 RMI는 비슷하게 우수한 성능을 보이지만, CORBA의 경우 대용량 바이너리 데이터에 대한 전송 속도나 확장성에서 문제를 보였다.

표 2. 바이너리 메시지 인코딩 실험 결과

Table 2. Result of binary message encoding.

쓰레드 수(개)	Planet 처리속도(ms)	RMI 처리속도(ms)	CORBA 처리속도(ms)
1	0.085	0.119	0.197
2	0.100	0.060	0.213
3	0.096	0.068	0.158
4	0.100	0.073	0.214
5	0.117	0.085	0.277
6	0.129	0.105	0.316
7	0.134	0.115	0.382
8	0.145	0.134	0.434
9	0.159	0.146	0.488
10	0.166	0.159	0.549
11	0.177	0.160	0.613
12	0.187	0.177	0.654
13	0.196	0.196	0.709
14	0.207	0.196	0.774
15	0.217	0.217	0.905
16	0.224	0.254	0.922
17	0.234	0.260	0.967
18	0.246	0.269	1.046
19	0.257	0.288	1.097
20	0.268	0.297	1.200

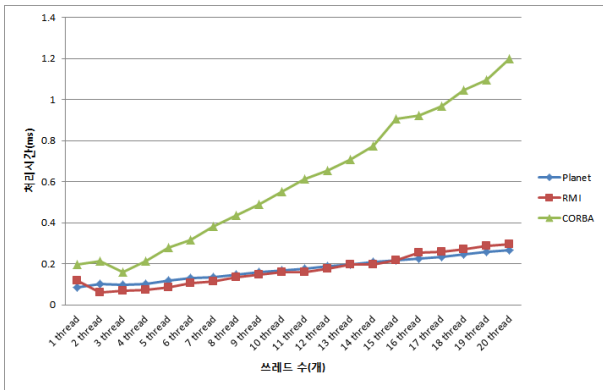


그림 6. 바이너리 메시지 인코딩  
Fig. 6. Encoding of binary message.

다. 대역폭 독점 방지

표 3은 4바이트 정수 메시지와 대용량의 바이너리 메시지를 동시에 보낼 때, 바이너리 메시지의 크기를 변화 시켜감에 따라, 4바이트 정수 메시지의 왕복 시간을 비교한 결과를 보여준다. 그림 7에서 보는 것처럼, 세

표 3. 대역폭 독점 방지 실험 결과  
Table 3. The bandwidth occupancy of BLOB.

Binary 크기 (kbyte)	Planet 처리속도(ms)	RMI 처리속도(ms)	CORBA 처리속도(ms)
1	0.39	0.13	0.46
2	0.39	0.13	0.46
4	0.39	0.13	0.47
8	0.38	0.12	0.46
16	0.37	0.14	0.47
32	0.38	0.13	0.46
64	0.36	0.12	0.47
128	0.38	0.13	0.46
256	0.37	0.11	0.46
512	0.36	0.11	0.46
1024	0.35	0.11	0.47
2048	0.33	0.1	0.47
4096	0.33	0.11	0.46

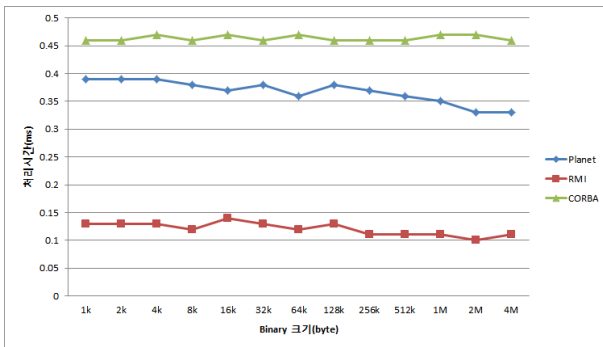


그림 7. 대역폭 독점 방지  
Fig. 7. The bandwidth occupancy of BLOB

가지 경우 모두 대역폭 독점 현상은 나타나지 않았다. 하지만, CORBA의 경우 전송 속도에서 문제를 보였다.

라. 전방 정체 방지

표 4는 4바이트 정수 메시지와 스트림 타입의 메시지를 동시에 보낼 때, 수신측에서의 스트림 메시지 처리 속도를 변화 시켜감에 따라, 4바이트 정수 메시지의 왕복 시간을 측정된 결과를 보여준다. RMI나 CORBA 모두 스트림 타입을 프레임워크 수준에서는 지원하지 않고, 응용에서 지원해야 하기 때문에 비교 대상이 되지 못한다. 따라서 이번 실험은 PLANET만을 대상으로 하였다. 그림 8에서 보는 것과 같이, 송수신하는 장치간의 처리능력의 차이에 상관없이 일정한 성능을 보여주고 있다.

표 4. 스트림 데이터 지원 실험 결과  
Table 4. Result of stream data support.

처리량(byte/ms)	Planet 소요시간(ms)
131	0.8
227	0.79
248	0.79
291	0.8
339	0.79
406	0.63
506	0.79
672	0.63
997	0.64
1,934	0.63
69,905	0.64

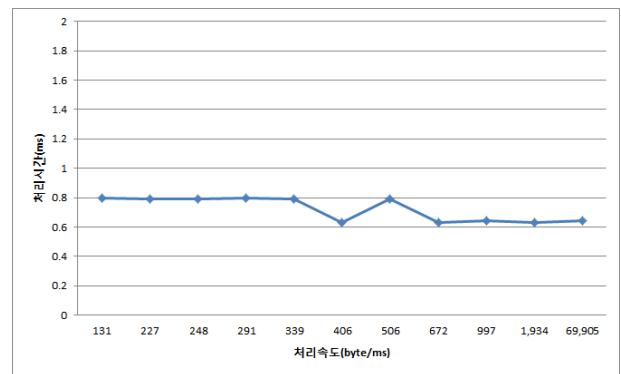


그림 8. 스트림 데이터 지원  
Fig. 8. Support of stream data.

2. 시나리오 기반 기능 검증

그림 9는 PLANET 기능 검증을 위한 시연 환경을 보여준다. 스마트 환경 내에 설치된 카메라로부터 사용자의 위치를 추적하여, 이형의 로봇들이 사용자의 위치



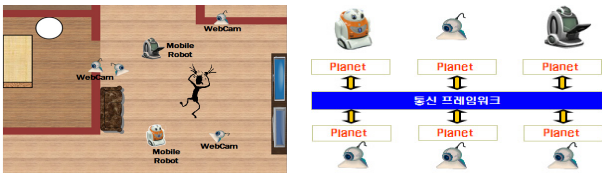


그림 9. 시연 환경  
Fig. 9. The test environment.



그림 10. 거실에서 사용자 추종  
Fig. 10. User following in the living room

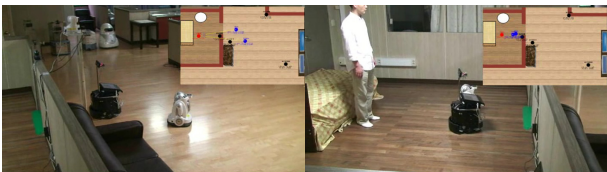


그림 11. 침실에서 사용자 추종  
Fig. 11. User following in the bedroom.

정보를 알아내어 사용자를 계속 추종하는 시연이다. 시나리오 시연을 위한 응용에서는 PLANET을 통해 4개의 카메라로부터 영상 이미지를 15fps로 얻어온다. 얻어온 영상을 분석해 사용자의 위치를 실시간으로 추적한다. 이 정보를 바탕으로 2대의 이형의 로봇들에게 각각 PLANET을 통해 사용자의 현재 위치로 이동시키는 명령을 내린다.

그림 10은 거실에 있는 3개의 카메라와 2대의 로봇이 PLANET을 통해 연동하는 시연 결과를 보여준다. 그림의 미니맵에서 검은 점들이 카메라, 파란 점들이 로봇, 그리고 빨간 점이 사용자를 각각 나타낸다.

그림 11은 거실에 있던 사용자가 침실로 이동했을 때, 두 로봇들의 사용자 추종 과정을 보여준다.

## V. 결 론

본 논문에서는 스마트 환경에서 이형의 통신 프로토콜/운영체제/프로그래밍 환경을 가진 다양한 플랫폼 간의 효과적인 상호 운용 방식을 제공하기 위한 통신 프레임워크인 PLANET을 소개하였다. PLANET은 객체지향형 IDL 기반의 원격 객체 함수 호출 방식의 통신 모델을 제공하기 때문에 범용의 통신 미들웨어로 사용할 수 있다. 아울러, 로봇과 스마트 환경 연동에 특화된

주요한 기능들을 지원함으로써, 로봇이 가진 저수준의 센서로부터 스마트 환경이 제공하는 고수준의 레거시 시스템까지 다양한 통신 대상들에 대해 유연하고 효율적인 통신 및 장치 모델을 제공한다.

II장에서 언급한 바와 같이, 저수준의 장치 서비스의 연동만으로는 고수준의 다양한 협업 응용을 모델링하기가 어렵다. PLANET을 통한 연동 또한 IDL 인터페이스 기반이기 때문에 저수준의 연동에 속한다. 따라서 현재 PLANET을 기반으로 행위 추상화 기반 협업 모델을 연구 중이며, 향후에 PLANET의 상위 계층으로 통합하여, 스마트 환경과 중재 로봇이 통합된 단일 로봇 제어 구조를 설계할 계획이다.

## 참 고 문 헌

- [1] Cook DJ, Das SK, editors. Smart environments: Technologies, protocols and applications. Hoboken: John Wiley and Sons; 2004
- [2] Setfan Poslad, Ubiquitous Computing: Smart Devices, Environments and Interactions. Wiley, ISBN: 978-0-0470-03560-3, 2009.
- [3] M. Weiser, "The Computer of the 21st Century", Scientific American, vol. 265, no. 3, 1991, pp. 66-75.
- [4] De Carolis B., Cozzolongo G., Interpretation of User's Feedback in Human-Robot Interaction. Journal of Physical Agents, vol. 3, no. 2, 2009
- [5] D. Lee, T. Yamazaki and S. Helal, "Robotic companions for smart space interaction", Pervasive Computing, pp. 78-84, 2009.
- [6] Ishii, H. Tangible bits: Beyond pixels. In Proc. of TEF'08, pages xv - xxv, Bonn, Germany, 2008.
- [7] 허기수, 이동우, 정현태, 박준석, "프로젝션 기반 증강현실 기술동향", 전자통신동향분석, 제26권, 제 5호, 92-101쪽, 2011년.
- [8] G. Coaaolongo, B. D. Carolis, S. Pizzutilo, "Social Robots as Mediators between Users and Smart Environments" IUI '07, pp. 353-356, 2007.
- [9] S. Coradeschi and A. Saffiotti, "Symbiotic robotic systems: Humans, robots, and smart environments," IEEE Intelligent Systems, vol.21, no.3, pp. 82-84, 2006.
- [10] Bien, Z.; Lee, H.; Do, J.; Kim, Y.; Park, K. & Yank, S. (2007) Intelligent Interaction for Human-friendly Service Robot in Smart House Environment. International Journal of Computational Intelligence Systems, vol. 1, no. 1, pp 77-93, (Jan 2008)

- [11] Nader Mohamed, Jameela Al-Jaroodi, and Jawhar, "A Review of Middleware for Networked Robots", IJCSNS, Vol. 9, 2009.
- [12] Brooks, A., Kaupp, T., Makarenko, A., Williams, S., and Orebaeck, A. (2007). Orca: A component model and repository. Software Engineering for Experimental Robotics, 231.
- [13] H. Bruyninckx, "Open Robot Control Software: The OROCOS Project," Proc. IEEE Int. Conf. Robot. Autom., 2001, pp. 21-26.
- [14] C. Côté et al., "Robotic Software Integration Using MARIE," Int. J. Advanced Robot. Syst., vol. 3, no. 1, 2006, pp. 55-60.
- [15] N. Ando et al., "RTMiddleware: Distributed Component Middleware for RT (Robot Technology)," IEEE/RSJ Int. Conf. Robots and Intelligent Systems, 2005, pp. 3555-3560.
- [16] D. C. Schmidh, "ACE: An Object-Oriented Framework for Developing Distributed Applications", Proceedings of the 6<sup>th</sup> USENIX C++ Technical Conference, April 1994.
- [17] H. Utz et al., "Miro-Middleware for Mobile Robot Application," IEEE Trans. Robot. Autom., vol. 18, no. 4, 2002, pp. 493-497.
- [18] B.P. Gerkey, R.T. Vaughan, and A. Howard, "The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems," Proc. Int. Conf. Advanced Robotics, 2003, pp. 317-323.
- [19] A. Saffiotti and M. Broxvall, "PEIS Ecologies: Ambient Intelligence Meets Autonomous Robotics," Int. Conf. Smart Objects and Ambient Intelligence, 2005, pp. 275-280.
- [20] M. Kranz, R. Rusu, A. Maldonado, M. Beetz, A. Schmidh, "A Player/Stage System for Context-Aware Intelligent Environments," in Proc. of the System Support for Ubiquitous Computing Workshop(UbiSys), Sep. 2006.
- [21] Kim, H., Lee, K.-W., Suh, Y.-H., Cho, J.-M., Cho, Y.-J.: Client/Server Framework for Providing Context-Aware Services to Network Based Robots. In: 16th IEEE International Symposium on Robot Human Interactive Communication, Jeju Island, Korea, pp. 475 - 480 (2007)
- [22] W. Yu et al., "Design and Implementation of a Ubiquitous Robotic Space," IEEE Trans. Autom. Sci. and Eng., Vol.6, No.4, Oct. 2009.

---

 저 자 소 개
 

---



서 영 호(정회원)  
 1998년 전남대학교 컴퓨터공학과  
 학사 졸업.  
 2000년 광주과학기술원 정보통신  
 공학과 석사 졸업.  
 2000년~현재 한국전자통신  
 연구원 선임연구원.

<주관심분야 : 유비쿼터스 컴퓨팅, 상황 인지 시스템, 지능형 서비스 로봇>



조 은 선(정회원)-교신저자  
 1991년 서울대학교 계산통계학과  
 학사 졸업  
 1993년 서울대학교 계산통계학과  
 석사 졸업  
 1998년 서울대학교 계산통계학과  
 박사 졸업

1999년~2000년 한국과학기술원 연구원  
 2000년~2002년 아주대학교 조교수대우  
 2002년~2006년 충북대학교 전기전자컴퓨터공학  
 부 조교수

2006년~현재 충남대학교 컴퓨터공학과 부교수  
 <주관심분야 : 프로그래밍 언어, 프로그래밍 모델, 상황인지 시스템, 이벤트 시스템, 컴파일러>



이 강 우(정회원)  
 1991년 서울대학교 계산통계학과  
 학사 졸업.  
 1993년 서울대학교 계산통계학과  
 석사 졸업  
 2000년 서울대학교 계산통계과  
 박사 졸업.

2000년~현재 한국전자통신연구원 책임연구원.  
 <주관심분야 : 유비쿼터스 컴퓨팅, 상황 인지 시스템, 지능형 서비스 로봇>