

# 실내 위치기반 사물인터넷 채팅 서비스 설계 및 구현

이 성 희\*, 정 설 영\*, 강 순 주\*, 이 우 진<sup>o</sup>

## Design and Implementation of IoT Chatting Service Based on Indoor Location

Sunghee Lee\*, Seol Young Jeong\*, Soon Ju Kang\*, Woo Jin Lee<sup>o</sup>

### 요 약

최근 통신, 교통, 의료, 산업시설 등에서 수요가 급증하고 있는 임베디드 시스템은 단순한 임베디드 시스템이 아닌 다양한 임베디드 기기의 동작이 통신을 통해 관찰되고 제어되는 Cyber Physical System(CPS)으로 확장되었다. 또한 구글 글래스, 삼성 갤럭시 기어, 소니 스마트 워치 등의 웨어러블 기기를 활용한 사물인터넷(IoT) 기술 활용에 대한 관심이 증가하고 있다. 이러한 상황에서 삼성전자의 스마트홈(Smart Home)과 LG전자의 홈챗(Home Chat)이 잇따라 출시되고 있다. 하지만 이러한 서비스들은 스마트폰과 가전기기간에만 통신이 가능하며 그 이외의 단말과는 통신이 불가능하며 모든 정보는 상업용 포털이 운영하는 글로벌 메시징 서버를 통하여 통신이 이루어진다는 단점이 있다. 본 논문에서는 앞서 언급한 단점을 보완하기 위해 단위공간 기반 실내위치인지망 구축하는 구조를 제안한다. 이를 통해 기기들 및 개개인에 대한 정보가 외부로 유출되는 것을 방지하면서 사물인터넷 채팅을 통해 망 내에 존재하는 모든 단말과의 선택적 통신이 가능하다. 또한 좀 더 범용적인 기기 제어가 가능해지고 개인정보의 외부유출 방지가 가능하다.

**Key Words** : IoT chatting, Smart home, Cyber physical system, Self-organizing software platform, Indoor localizing network

### ABSTRACT

Recently, embedded system which demand is explosively increasing in the fields of communication, traffic, medical and industry facilities, expands to cyber physical system (CPS) which monitors and controls the networked embedded systems. In addition, internet of things(IoT) technology using wearable devices such as Google Glass, Samsung Galaxy Gear and Sony Smart Watch are gaining attention. In this situation, Samsung Smart Home and LG Home Chat are released one after another. However, since these services can be available only between smart phones and home appliances, there is a disadvantage that information cannot be passed to other terminals without commercial global messaging server. In this paper, to solve above issues, we propose the structure of an indoor location network based on unit space, which prevents the information of the devices or each individual person from leaking to outside and can selectively communicate to all existent terminals in the network using IoT chatting. Also, it is possible to control general devices and prevent external leakage of private information.

\* 본 연구는 미래창조과학부 및 한국산업기술평가관리원의 산업융합원천기술개발사업(SW.컴퓨팅)의 일환으로 수행하였음. [10041145, 자율군집을 지원하는 웰빙형 정보기기 내장 소프트웨어 플랫폼 개발]

• First Author : Kyungpook National University School of Computer Science & Engineering, lee3229910@gmail.com, 학생회원

o Corresponding Author : Kyungpook National University School of Computer Science & Engineering, woojin@knu.ac.kr, 정회원

\* Kyungpook National University School of Electronics engineering, snowflower@ee.knu.ac.kr, 종신회원, sjkang@ee.knu.ac.kr

논문번호 : KICS2014-06-223, Received June 2, 2014; Revised September 1, 2014; Accepted September 24, 2014

## I. 서 론

최근 통신, 교통, 의료, 산업시설 등 임베디드 시스템이 사용되지 않는 분야를 찾아보기 힘들 정도로 그 수요가 많다. 기존의 임베디드 시스템은 우리가 사용하는 컴퓨터와는 다르게 특정한 목적을 위해서 사용되는 시스템이다. 이런 임베디드 시스템의 수요가 증가하면서 임베디드 시스템 간의 상호작용이 필요하게 되었다. 다시 말해, 어떤 임베디드 기기의 상태나 동작에 따라 다른 임베디드 기기의 상태나 동작의 제어가 필요하게 되었으며 이러한 상호작용은 통신망을 통해 이루어지게 된다. 이런 임베디드 시스템이 모여 사이버물리시스템(Cyber Physical System;CPS)을 이루게 되며 CPS는 우리의 생활에 커다란 변화를 가져오고 있다<sup>1)</sup>.

또한 구글의 구글 글래스, 삼성의 갤럭시 기어, 소니의 스마트 워치의 등장은 우리 일상생활의 새로운 패러다임을 제시하고 있다. 이러한 웨어러블 기기의 등장으로 인해 사회 전반적인 각 분야에서 IT융합의 움직임이 더욱 가속화되고 있다<sup>2)</sup>. 이러한 IT융합의 대표적인 분야가 바로 의료기기 분야이며 그 중에서도 U-헬스케어 서비스는 정보통신 기술과 의료정보 시스템을 연결하여 시공간에 구애받지 않고 각종 병들을 예방, 진단, 치료 등의 서비스를 제공한다<sup>3,4)</sup>.

U-헬스케어 기기는 분산되어 있는 센서들로부터 측정된 정보가 네트워크로 송신되면 그 정보를 토대로 사용자의 건강상태를 분석할 수 있는 기기로서 손목시계형 혈압계, 혈당계 등이 있다<sup>4,5)</sup>. 또한 일상생활에서 흔히 사용하는 기기들도 사물인터넷(IoT : Internet of Things) 기술을 활용한다면 얼마든지 U-헬스케어 기기가 될 수 있다. 예를 들어, 손목시계형 혈압계를 통해 측정된 나의 혈압에 대한 정보를 홈 네트워크로 송신한다. 러닝머신은 그 정보를 수신하여 운동이 필요한 경우, 러닝머신은 알람을 울려 나에게 운동을 하라고 알려줄 수도 있다.

이처럼 IT융합의 많은 부분을 차지하고 있는 사물인터넷 기술 활용 또한 그 관심도가 급증하고 있다. 삼성전자의 스마트홈(Smart Home)과 LG전자의 홈챗(Home Chat)이 그 결과물이며 이 서비스들은 스마트폰을 통해 가전기기를 제어하고 있다<sup>2,6)</sup>.

하지만 이 서비스들은 스마트폰과 가전기지만 통신이 가능하며 특히 홈챗의 경우, 네이버가 운영하는 모바일 메신저 라인을 통해서 통신이 이루어지므로 개인정보가 상업용 포털이 운영하는 글로벌 메시징 서버를 통해서만 통신이 가능하다는 문제점이 있으며

이로 인해 개인정보 유출이 일어날 수 있다<sup>7)</sup>. 또한 이러한 글로벌 메시징 서버는 보통 중앙집중형으로 구축되어 있는데 이런 구조는 모든 데이터의 전달과 서비스의 제공이 중앙서버에서 이루어지므로 많은 사용자에게 동시다발적인 서비스를 제공한다거나 복잡도가 높은 서비스를 제공하기에 적합하지 않으며 심각한 경우에는 서버가 다운될 수 있다<sup>8)</sup>. 이러한 중앙집중형 서버의 구축과 유지보수에 드는 비용은 분산컴퓨팅 서버의 경우보다 크다.

본 논문에서는 이러한 문제점을 해결하기 위해 분산컴퓨팅 서버 개념을 가진 슬림허브(SLIM-Hub: Self-organizing Localizing Iot Messaging Hub)를 사용하며 이것을 중심으로 위치기반 실내위치인식망을 구축하여 개인정보의 외부유출을 막고 사물인터넷을 활용한 채팅 에이전트를 사용하여 다양한 기기들의 제어가 가능하며 원하는 서비스를 선택적으로 사용할 수 있는 구조를 제안하고자 한다.

본 논문은 다음과 같이 구성되어 있다. II장에서는 사이버물리시스템과 자율군집형 소프트웨어 플랫폼에 대해서 설명하며 III장에서는 단위공간 기반 실내위치인식망과 기기 간 채팅에 대한 구조를 제안한다. IV장에서는 III장에서 제안한 구조를 구현한 결과를 보이며 V장에서 결론을 맺는다.

## II. 관련연구

### 2.1 사이버물리시스템(Cyber Physical System)

최근 임베디드 시스템이 사용되는 분야를 찾는 것보다 사용되지 않는 분야를 찾기가 더 어려울 정도로 그만큼 다양한 분야에서 사용되고 있다. 단순한 기기부터 복잡한 시스템까지 임베디드의 규모 또한 커지고 있다. 기존의 임베디드 시스템은 소프트웨어를 통해 하드웨어를 동작시키고 상태를 모니터링 하는데 그 초점을 두었지만 지금은 더 나아가 임베디드 기기 간의 상호작용에도 관심을 가지고 있다. 이러한 상호작용의 중요성이 점점 커지면서 임베디드 시스템은 사이버물리시스템으로 확장되고 있다. 사이버물리시스템은 실제 공간의 물리적으로 존재하는 물리시스템과 사이버 공간의 존재하는 소프트웨어인 사이버시스템으로 구성되어 있으며 물리시스템은 통신망을 통해 그 상태나 동작이 관찰되고 제어된다<sup>1)</sup>. 예를 들어, 구글의 구글 글래스, 삼성의 갤럭시 기어, 소니의 스마트 워치 등 웨어러블 기기는 웨어러블 사이버물리시스템의 일종으로 볼 수 있다.

또한 사이버물리시스템은 IT융합 시스템의 한 분야

이기도 하며 그 중에서도 U-헬스케어 서비스는 정보통신 기술과 의료정보 시스템을 연결하여 시공간에 구애받지 않고 각종 병들을 예방, 진단, 치료 등의 서비스를 제공한다<sup>3,4)</sup>. U-헬스케어 기기는 분산되어 있는 센서들로부터 측정된 정보를 토대로 사용자의 건강상태를 분석한 후, 통신망을 통하여 스마트 시계의 알람을 울리거나 복약기의 동작시켜 약을 처방할 수 있다. 이는 물리시스템의 상호작용을 보여주는 좋은 예이다. 본 논문에서는 이러한 물리시스템 간의 상호작용을 모니터링하고 물리시스템을 제어하기 위해 사물인터넷을 활용한 스마트 채팅을 구현하고자 한다.

### 2.2 자율군집형 소프트웨어 플랫폼(SoSP)

기존의 유비쿼터스 환경은 다소 규모가 큰 공간에서 사용자의 위치 정보를 활용하여 서비스를 제공하였다. 하지만 일반 가정환경에서 또는 사무실에서 방단위로 사용자의 위치 정보를 활용할 수 있다면 더욱 발전된 서비스를 제공할 수 있다<sup>9)</sup>. 자율군집형 소프트웨어 플랫폼(SoSP : Self-organizing software platform)은 이러한 실내 위치기반 서비스를 제공하며 단위 공간(방, 사무실 등) 내에서 그곳을 관장하는 라우터, 서비스를 제공하는 리소스 장치와 사용자가 서비스를 요청할 수 있는 모니터링 장치로 구성된다. 자율군집은 단위 공간 내에 라우터와 장치간의 자율적인 서비스 바인딩 등에 필요한 기술을 의미하며 해당 공간 내에 존재하는 기기들 간의 상호작용 통해 자원을 공유하며 메시지를 주고받는다<sup>10,11)</sup>.

SoSp는 위치기반의 비동기 푸시(Push)/풀(Pull) 서비스를 제공한다. 이러한 서비스는 모두 XML 형태의 포스트잇 기반이다. 포스트잇은 우리가 오프라인에서 실제 사용하는 포스트잇의 역할을 온라인상에서 한다. 예를 들어, 팀장이 팀원 전체에게 전달할 내용이 있다고 가정하자. 그리고 그 내용을 모든 팀원들에게 전달하고 싶고 모두가 확인하였는지를 알고 싶다. 이 상황에서 실제 포스트잇을 사용한다면 전달하고자 하는 내용을 각각의 포스트잇으로 작성해 각자의 책상에 붙여놓을 수 있다. 이 방법은 중복된 포스트잇을 팀원들의 숫자만큼 만들어야 하며 그 포스트잇을 확인한 사람이 누구인지는 쉽게 확인하기 어렵다. 하지만 SoSp에서는 쉽게 포스트잇을 전달할 수 있다. 단위 공간을 팀원들이 사용하는 사무실로 설정한 뒤 전달할 내용을 음성메시지 포스트잇 형태로 라우터로 전달하면 각각의 팀원들이 사무실로 들어왔을 때 그 포스트잇을 확인할 수 있다. 그러면 오프라인의 포스트잇을 사용하는 것보다 훨씬 효율적이며 그 포스트잇

을 확인하였는지 유무를 보다 쉽게 확인할 수도 있다. 본 논문에서 이러한 SoSp 개념을 도입하여 서비스를 제공하는 서비스기기, 서비스를 요청하고 서비스의 상태를 제어할 수 있는 스마트기기, 라우터의 기능을 하는 슬림허브로 SoSp를 구성하고자 한다.

## III. 단위공간 기반 실내위치인지망 구축 및 채팅 에이전트를 통한 기기 간 통신 실현

### 3.1 단위공간 기반 실내위치인지망을 위한 구성요소 및 전체구조

본 장에서는 단위공간 기반 실내위치인지망을 구성하는 구성요소 및 구조를 그림 1과 같이 제한한다. 그 구성요소는 슬림허브, 스마트 기기(smart device), 서비스 기기(service device)가 있다. 슬림허브는 단위공간 내의 실내위치인지망을 구축하며 그 안에서 슬림허브와 기기 간의 자율군집이 일어난다. 슬림허브는 또한 구축된 자신의 실내위치인지망에 연결된 기기들의 정보를 관리하며 스마트 기기 및 서비스 기기 간 M2M 메시지를 관장한다. 또한 자신의 실내위치인지망을 구축한 슬림허브는 필요시 슬림허브 간의 통신을 통해 서로에게 연결된 기기들의 정보를 공유하거나 메시지를 전달할 수 있다.

스마트 기기는 스마트폰, 스마트 태블릿 등이 존재하며 이들은 슬림허브와 정보를 주고받으며 기본적으로 포스트잇 에이전트(Post-it agent)와 주로 통신이 이루어진다. 서비스 기기는 가스레인지, 러닝머신, 복

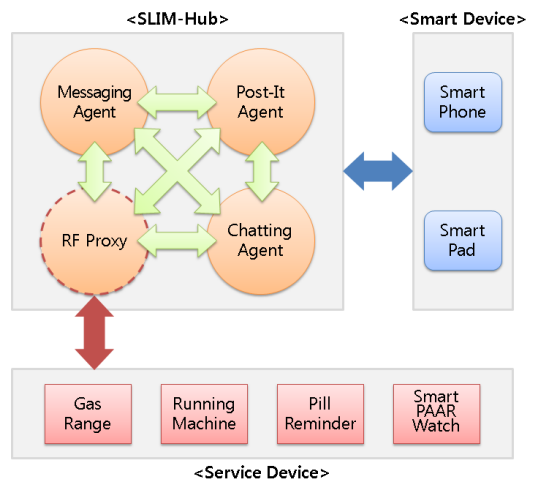


그림 1. 단위공간 기반 실내위치인지망을 위한 구성요소 및 전체구조  
 Fig. 1. Structure and component for network recognizing indoor location based on unit space

약기, Smart PAAR Watch 등이 있으며 이들은 사물인터넷 기술을 탑재하여 슬림허브와 통신이 가능하다. 또한 서비스 기기는 기본적으로 어떤 정보를 처리할 수 있는 능력이 없으며 단순히 슬림허브와 데이터만 주고받을 수 있다고 가정한다. 그리고 그 데이터들을 어떤 의미 있는 정보로 바꿔준다거나 반대의 경우는 모두 RF 프록시(Proxy)에서 그 역할을 수행한다. 따라서 서비스 기기와 유일하게 통신하는 것은 RF 프록시뿐이다. 슬림허브는 자신이 존재하는 실내위치인지 망내에 있는 스마트 기기와 서비스 기기간의 연동을 가능하게 하며 이에 필요한 모든 정보를 관리하고 전달한다. 슬림허브는 포스트잇 에이전트, 메시징 에이전트(Messaging agent), 채팅 에이전트(Chatting agent), RF 프록시로 구성된다.

포스트잇 에이전트는 스마트 기기와 서비스 기기가 포스트잇을 만들 수 있도록 도와주며 메시징 에이전트는 만들어진 포스트잇 메시지 또는 채팅 메시지를 각각의 기기로 전달하는 역할을 한다. 채팅 에이전트는 기기 간의 채팅을 관리한다. 이 때, 채팅이란 우리가 일반적으로 사용하고 있는 채팅의 기능에 추가적으로 일련의 포스트잇이 포함된 것으로 가정한다. RF 프록시는 서비스 기기로부터 오는 데이터를 의미있는 정보로 바꿔주거나 스마트 기기로부터 오는 정보에 따라 서비스 기기를 제어하는 역할을 한다. 또한 RF 프록시는 슬림허브 내의 에이전트와 서비스 기기 간의 일종의 인터페이스로 볼 수도 있고 서비스 기기를 관리하는 에이전트로 생각할 수도 있다. 각각의 에이전트는 상호간의 통신이 이루어지며 이를 통해 포스트잇이 생성되고 전달되며 채팅 생성, 참가 및 채팅 메시지 전달이 이루어진다.

### 3.2 채팅 에이전트

홈 네트워크, 오피스 네트워크 등의 서비스 공간 내에는 수많은 서비스 기기와 스마트 모바일 단말들이 존재한다. 이 기기들 중 사용자가 필요로 하는 기기들 간의 M2M 메시징이 요구된다. 예를 들어, 가스레인지의 상태정보를 스마트폰에서 확인 또는 마케팅부서 간의 채팅 등이 있다. 이를 위해서 인터넷에서 사용 중인 채팅의 개념을 도입한다.

스마트 기기는 서비스 기기로부터 오는 메시지를 받을 수 있고 반대로 서비스 기기를 제어할 수 있다. 이러한 메시지를 주고받을 때, 보통의 비주기적이거나 이산적인 메시지의 경우 포스트잇으로 주고받는 것이 효과적이지만 그렇지 않은 경우에는 많은 정보가 캡슐화가 된 포스트잇으로 하나하나 메시지를 주고받는

것은 비효율적이다. 채팅 에이전트는 스마트 기기간의 채팅을 담당하는 것뿐만 아니라 이러한 문제점을 보다 효율적으로 해결하기 위해 존재한다. 또한 일련의 포스트잇을 관리하게 위해서도 필요하다. 예를 들어, 사용자가 스마트폰으로 가스레인지의 상태를 받아보고 싶다고 슬림허브에 요청하면 그 시점부터 채팅 에이전트는 가스레인지로부터 오는 정보를 스마트폰으로 전달하게 된다. 이 때, 전달되는 메시지는 단지 일회성의 조각난 메시지에 그치지 않고 가스레인지만의 히스토리로서도 의미를 가지며 꾸준히 모니터링해야 할 의미가 있는 서비스 기기이기도 하다. 따라서 가스레인지에 해당하는 세션을 따로 관리할 필요가 있다. 채팅 세션의 생성, 접속은 모두 포스트잇을 통해 이루어진다. 채팅 세션에 접속이 이루어지면 그 이후에 일어나는 메시지 전달 등의 채팅에 관한 모든 일들은 채팅 에이전트에서 담당한다.

### 3.3 스마트 기기 간의 채팅

채팅이 관여하는 구성기기들은 스마트폰 등의 스마트 기기와 서비스 기기로 나누어지며 이들 간에 자유로운 메시징이 발생한다. 스마트 기기 간의 채팅의 시작은 그림 2와 같다.

먼저 어떤 스마트 기기가 채팅을 생성하고자 하는 포스트잇을 슬림허브에 보내면 포스트잇 에이전트는 채팅 에이전트에게 새로운 채팅 세션을 생성하라는 메시지를 송신한다. 채팅 에이전트가 새로운 세션을 만드는 데 성공하였으면 그 세션 ID 정보를 스마트 기기에 전달하고 생성된 채팅 세션에 대해 다른 스마트 기기들이 접속할 수 있도록 채팅 참가 포스트잇을 포스트잇 에이전트에게 보낸다. 포스트잇 에이전트를 그 채팅에 대한 세션 ID 정보를 메시징 에이전트로 전달한다. 메시징 에이전트는 해당 실내위치인지만에 연결되어 있는 모든 스마트 기기들에게 세션에 참가할 수 있도록 푸시 메시지를 송신한다.

위의 과정을 거쳐 생성된 채팅의 참가 푸시 메시지는 모든 스마트 기기에게 송신된다. 메시지를 받은 기

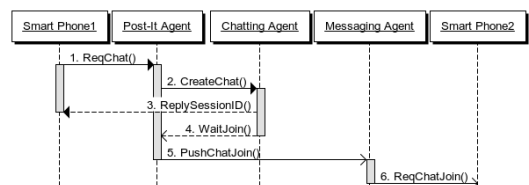


그림 2. 스마트 기기 간의 채팅 생성 및 채팅 참가 푸시 메시지 전달 과정  
Fig. 2. Chatting creation between smart devices and push message transmission process of join the chatting

기들은 채팅에 접속이 가능하며 접속 후, 기기 간의 채팅이 시작된다. 채팅메시지의 전달 메커니즘은 그림 3과 같다.

스마트 기기가 어떤 채팅에 어떤 메시지를 입력하면 그 메시지는 채팅 에이전트로 전송되며 채팅 에이전트는 그 메시지를 메시징 에이전트로 전달한다. 그 메시지는 메시징 에이전트를 통해 해당 채팅에 연결된 모든 기기에게 전송된다.

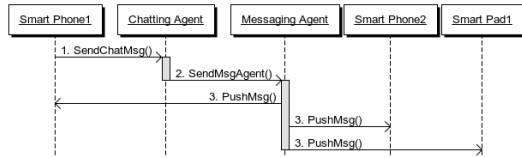


그림 3. 스마트 기기 간의 채팅 생성 및 채팅 참가 푸시 메시지 전달 과정  
Fig. 3. Chatting message transmission process between smart devices

### 3.4 스마트 기기와 서비스 기기 간의 채팅

스마트 기기와 서비스 기기 간의 채팅은 다른 관점에서 보면 스마트 기기가 서비스 기기의 상태 또는 동작을 모니터링할 수 있고 제어할 수 있는 상태가 된다고 할 수 있다. 즉, 서비스 기기는 자신과 연결된 스마트 기기에게 해당 서비스를 제공한다.

스마트 기기와 서비스 기기의 채팅은 스마트 기기로부터 시작되며 그 메커니즘은 그림 4와 같다. 스마트 기기가 어떤 서비스 기기의 서비스를 요청하는 포스트잇을 슬립허브에 보내면 포스트잇 에이전트는 채팅 에이전트에게 서비스 기기에 해당하는 채팅 세션을 생성하라는 메시지를 송신한다. 채팅 에이전트는 RF 프록시로 해당 서비스 기기에 대한 ID 정보를 요청하고 받은 후, 채팅 세션을 생성한다. 이 때, 채팅 세션이 이미 만들어져 있으면 이 과정이 생략된다. 채팅 세션이 생성되어 있으면 채팅 에이전트가 세션 ID 정보를 스마트 기기에 전달하고 스마트 기기는 해당 세션을 통해 서비스 기기에 대한 채팅을 주고 받는다.

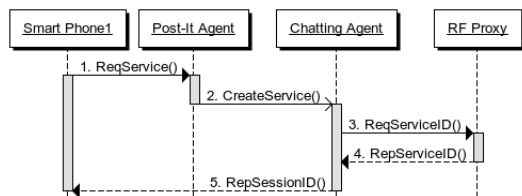


그림 4. 스마트 기기와 서비스 기기 간의 채팅 생성  
Fig. 4. Chatting message transmission process between smart devices

스마트 기기와 서비스 기기 간의 채팅은 스마트 기기에서 서비스 기기로 메시지를 송신하여 서비스 기기를 제어하는 채팅과 서비스 기기에서 스마트 기기로 메시지를 송신하여 서비스를 제공하는 2가지로 나뉜다. 먼저, 스마트 기기에서 서비스 기기로 제어 메시지를 송신하는 과정은 그림 5와 같다. 어떤 제어 메시지를 입력하면 그 메시지는 채팅 에이전트로 전송되며 채팅 에이전트는 그 메시지를 메시징 에이전트로 전달한다. 그 메시지는 메시징 에이전트를 통해 RF 프록시로 전달하며 최종적으로 서비스 기기로 메시지를 전달한다.

서비스 기기로부터 스마트 기기로 보내지는 정보는 그림 6과 같다. 서비스 기기에서 스마트 기기로 메시지를 송신하면 메시지는 RF 프록시를 거쳐 채팅 에이전트로 전송된다. 그 메시지는 다시 메시징 에이전트를 통해 최종적으로 스마트 기기로 전달된다.

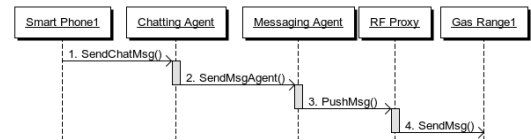


그림 5. 스마트 기기 간의 채팅 생성 및 채팅 참가 푸시 메시지 전달 과정  
Fig. 5. Chatting message transmission process from smart device to service device

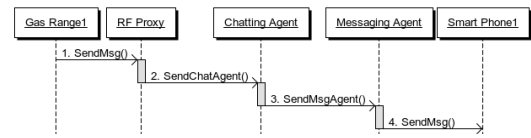


그림 6. 서비스 기기로부터 스마트 기기로 채팅 메시지 전달 과정  
Fig. 6. Chatting message transmission process from service device to smart device

## IV. 구현 및 실험

### 4.1 서비스 시나리오

할아버지, 할머니만 사는 집에서는 여러 가지 일들이 발생할 수 있다. 할머니가 가스레인지 불을 켜놓은 채로 잠이 들 수도 있고 약 먹을 시간이 되었는데 약 먹기를 잊어버릴 수도 있다. 이런 상황은 노약자에게는 치명적인 상황일 수 있다. 이러한 노약자를 제대로 돌보기 위해서는 가정부가 따로 들어가거나 같이 사는 방법이 있을 것이다. 이 모든 것들이 불가능한 경우에는 노약자를 제대로 돌보기에는 매우 힘들 것이다.

하지만 단위공간 기반 실내위치인지망에서는 이러한 문제점을 해결할 수 있다. 그림 7은 단위공간 기반 실내위치인지망에서의 서비스 시나리오들 중 일부를 나타내었다. 원격지에서 보호자는 할아버지, 할머니집의 서비스 기기의 정보를 슬림허브로부터 스마트 기기를 통해 항상 모니터링이 가능하며 원격 제어 또한 가능하다.

그림 7의 한 가지 시나리오를 예를 들어 설명하면, 할머니가 가스레인을 켜면 자동으로 스마트 PAAR Watch는 가스레인의 상태를 인식하게 된다. 그리고 시계는 바로 가스레인의 정보를 슬림허브로 전송하며 슬림허브는 다시 원격지 보호자의 스마트 기기(스마트폰)로 가스레인의 상태를 알려주게 된다. 이를 통해 원격지 보호자는 가스레인을 제어하거나 할머니에게 연락하여 특별한 조치를 취할 수 있을 것이다.

본 논문에서는 이러한 서비스 시나리오에서의 서비스 기기(가스레인지)로부터 원격지 스마트 기기(스마트폰)로 서비스 기기의 상태를 알려주는 메시지 또는 원격지 스마트 기기(스마트폰)로부터 서비스 기기를 제어하는 메시지를 전달해주는 역할을 채팅에이전트를 구현하고 검증하였다.

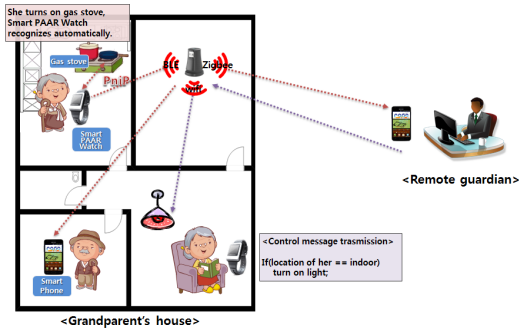


그림 7. 단위공간 기반 실내위치인지망에서의 서비스 시나리오  
 Fig. 7. Service scenario in network of recognizing indoor location based on unit space

#### 4.2 채팅 에이전트의 구현

본 논문에서 제안하는 채팅 에이전트는 간단하게 말하자면 기존의 채팅에서의 서버 역할을 한다고 생각하면 된다. 엄밀하게 말하면 기존의 서버는 채팅의 세션 관리 및 메시지를 전달하는 역할을 하고 있지만 채팅 에이전트는 채팅 세션 관리만 하고 실질적인 메시지 전달은 메시징 에이전트가 담당한다. 또한 슬림허브의 개수만큼 실내위치인지망이 구축되어 있어 채팅에이전트는 슬림허브 개수만큼 존재할 수 있다. 구

현에 사용된 하드웨어는 Hardkernel ODROID-X2로써 프로세서는 Samsung Exynos4412 Cortex-A9 Quad Core 1.7Ghz with 1MB L2 cache, RAM은 2GB LP-DDR2 880Mega data rate, 이더넷 인터페이스는 LAN9730 HSIC USB 2.0 to 10/100 Ethernet controller with HP Auto-MDIX from SMSC/Microchip이다.

#### 4.3 성능평가

슬림허브와 채팅 에이전트에 대한 성능평가를 하기 위해서는 실제 스마트 기기 간 또는 스마트 기기와 서비스 기기 간의 채팅 메시지의 전송시간을 측정하여야 하지만 다소 많은 수의 기기에 대해서는 정확한 검증은 하기 어렵다. 따라서 본 논문에서는 가상의 서버 환경에서의 채팅에이전트 성능평가와 실물의 실내위치인지망에서의 성능평가를 나누어서 실시하였다.

가상의 서버환경에서의 채팅에이전트 성능평가는 PC환경에서 서버에 해당하는 채팅에이전트는 구현하였으나 클라이언트에 해당하는 기기들은 스텝(Stub)으로 대체하였고 실물 실내위치인지망에서의 성능평가는 슬림허브를 구현하고 실제 스마트 기기를 사용하여 성능평가를 하였다.

##### 4.3.1 가상의 서버환경에서의 채팅 에이전트 성능평가

본 항에서는 PC환경에서 채팅에이전트는 실제 구현하였고 클라이언트에 해당하는 기기들은 스텝으로 구현되었다. 성능평가를 위해 사용한 PC의 스펙은 표 1와 같다.

본 항의 성능평가는 크게 2가지로 나누어서 진행하였다. 첫 번째는 하나의 프로세스에서 전송되는 채팅 메시지의 전송 시간을 100회에 걸쳐 측정하였고 두 번째는 여러 개의 채팅에이전트와 비교할 간단한 형태의 채팅서버를 스텝으로 구현하여 채팅 메시지 전송 시간과 전송 성공률을 측정하였다. 성능평가에 사용된 모든 채팅 메시지는 100 bytes의 문자 데이터이다.

표 1. 가상의 서버환경에서의 성능평가를 위한 PC의 스펙  
 Table 1. Specification for performance evaluation in the environment of virtual server on PC

CPU	Intel(R) Core(TM)2 Quad CPU Q9450 @ 2.66GHz
RAM	4GB
OS	Ubuntu 14.04 LTS
Ethernet interface	RTL8111/8168/8411 PCI Express Gigabit Ethernet Controller

표 2. PC환경에서의 채팅에이전트와 스텝서버를 사용하여 채팅 메시지의 성공률

Table 2. Success rate of chatting message using Chatting agent or stub server in the environment on PC

Division	Stub Server	Chatting Agent
1 Process (200 threads)	100%	100%
2 Process (400 threads)	100%	100%
3 Process (600 threads)	100%	100%
4 Process (800 threads)	100%	100%
5 Process (1,000 threads)	99.5%	100%
6 Process (1,200 threads)	49.7%	100%

첫 번째 성능평가는 총 100번의 시도를 하였으며 채팅에이전트를 통한 채팅 메시지의 전송은 모두 성공하였다. 따라서 전송률은 100%로 측정되었으며 전송 시간 분포는 그림 8과 같다. 평균 전송 시간은 372 usec이며 최소 전송 시간과 최대 전송 시간은 각각 180 usec, 593 usec로 나타났다.

두 번째 성능평가는 총 12회에 걸쳐 진행하였고 결과는 표 2와 같다. 앞의 6회는 우리가 구현한 채팅에이전트를 통하여 채팅 메시지를 전송하여 진행하였고 뒤의 6회는 스텝 서버를 통하여 채팅 메시지를 전송하여 진행하였다. 스텝 서버는 단순히 채팅을 위한 메시지를 주고받는 역할만 하였다. 채팅 메시지를 보내는 클라이언트는 1개의 스레드가 동작하는 1개의 프로세스으로써 1개의 채팅 메시지를 1번만 전송하였다.

클라이언트에 해당하는 스텝은 단일 프로세스이며 프로세스마다 200개의 스레드가 존재한다. 또한 모든 클라이언트는 모두 동일한 채팅 세션에 참여하도록

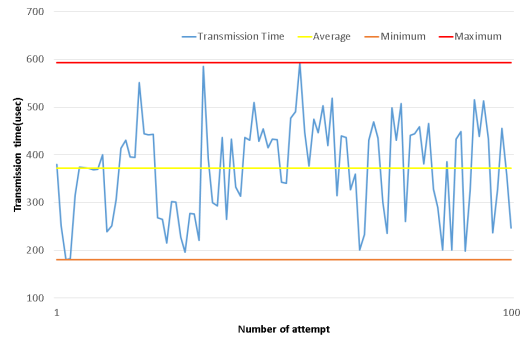


그림 8. PC환경에서의 채팅 에이전트를 통한 하나의 클라이언트의 채팅 메시지 전송시간 분포

Fig. 8. Distribution of transmission time per attempt for a client through Chatting agent in the environment on PC

구현하였다. 클라이언트 수를 제외한 나머지 조건은 각각의 성능평가 시 모두 동일하게 진행하였고 각각의 클라이언트의 수는 200, 400, 600, 800, 1,000, 1,200로 진행하였다. 채팅에이전트는 슬림허브가 3개가 있다고 가정하여 3개의 채팅에이전트를 실행하여 채팅 메시지의 전송시간 및 성공률을 측정하였다.

1,200개의 클라이언트로 채팅 메시지를 전송할 때, 채팅 에이전트에서는 모든 클라이언트가 채팅 메시지를 정상적으로 모두 수신한 것을 알 수 있다. 하지만 스텝 서버에서는 클라이언트가 1,000개 이상일 경우에는 모든 클라이언트가 채팅 메시지를 받지 못하는 문제점이 발생하였다. 표 3은 1,000개의 클라이언트까지 채팅 메시지의 전송률 및 전송시간이다. 6 Process의 경우는 전송률이 너무 낮아서 전송시간을 비교하는데 무리가 있어 1,000개의 클라이언트까지만

표 3. PC환경에서의 채팅에이전트와 스텝서버를 사용하여 채팅 메시지의 전송시간 및 성공률

Table 3. Transmission time and success rate of chatting message using Chatting agent or stub server in the environment on PC

Number of clients	Division	Average transmission time (usec)	Minimum transmission time (usec)	Maximum transmission time (usec)	Success rate of transmission (%)
	Type of Servers				
1 Process (200 threads)	Stub Server	3,540	786	5,433	100
	Chatting Agent	5,501	1,414	9,688	100
2 Process (400 threads)	Stub Server	5,465	918	9,530	100
	Chatting Agent	7,129	904	14,701	100
3 Process (600 threads)	Stub Server	16,355	1,672	34,995	100
	Chatting Agent	10,546	1,095	18,170	100
4 Process (800 threads)	Stub Server	20,995	1,200	51,945	100
	Chatting Agent	13,312	510	30,067	100
5 Process (1,000 threads)	Stub Server	25,305	1,840	55,129	99.5
	Chatting Agent	18,495	935	35,133	100

전송시간을 비교하였다.

채팅 메시지의 최대 전송 시간은 채팅 메시지가 모든 클라이언트에게 전달된 시간을 의미한다. 스텝 서버를 통한 클라이언트 수에 따른 채팅 메시지 전송 시간을 보면 클라이언트의 수가 작을 때는 채팅 메시지의 평균 전송 시간, 최소 전송 시간, 최대 전송 시간 모두 매우 빠르다. 하지만 클라이언트 수가 증가하면서 최대 전송 시간은 급격하게 증가한다. 더욱 치명적인 것은 전송 성공률이 100%가 되지 않는다는 것이다. 이것은 중앙집중형 서버의 문제점이 드러나는 결과이다.

채팅 에이전트를 사용한 경우에는 클라이언트의 수가 작을 때는 채팅 메시지의 평균 전송 시간, 최소 전송 시간, 최대 전송 시간 모두 스텝 서버를 통한 전송 시간보다 느리다. 하지만 클라이언트 수가 증가하면서 최대 전송 시간은 스텝 서버를 통한 때보다 경우보다 완만하게 증가한다. 왜냐하면 채팅 에이전트 간의 통신도 일어나기 때문에 클라이언트 수가 적을 때는 전송 시간을 지연시키지만 오히려 클라이언트의 수가 많아질수록 전송 시간을 줄여주게 된다. 이것은 분산 컴퓨팅 서버의 장점이 된다. 채팅 에이전트 및 스텝 채팅 서버를 통한 클라이언트 수에 따른 채팅 메시지 최대 전송 시간은 그림 9와 같다. 분산형 서버인 채팅 에이전트를 사용하였을 때와 중앙집중형의 스텝 서버를 사용하였을 때의 채팅 메시지 전송 시간 차이를 확인할 수 있다.

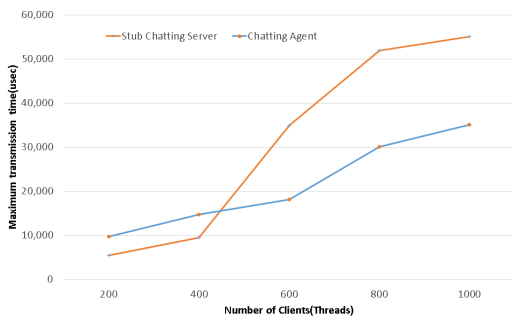


그림 9. 채팅 에이전트 및 스텝 채팅 서버를 통한 클라이언트(스레드) 수에 따른 채팅 메시지 최대 전송 시간  
Fig. 9. Maximum transmission time of a chatting message through chatting agent and stub chatting server according to the number of clients(threads)

#### 4.3.2 실물의 실내위치인자망에서의 성능평가

채팅 에이전트에 대한 성능평가를 하기 위해서는 실제 스마트 기기 간 또는 스마트 기기와 서비스 기기

간의 채팅 메시지의 전송시간을 측정하여야 하지만 다소 많은 수의 기기에 대해서는 정확한 성능평가를 하기 어렵다. 따라서 본 논문에서는 채팅 에이전트 성능평가에 사용한 실제 기기는 가스레인지와 슬림허브는 각각 1대를 사용하고 스마트폰은 1대, 3대를 사용하여 2세트에 걸쳐 성능평가를 수행하였다. 그 모든 성능평가에 사용된 메시지는 가스레인지가 켜질 때 생성되는 메시지인 “A gas stove is turned on.”이며 이 메시지는 RF 프록시에서 생성되어 스마트폰으로 전송된다.

첫 번째 성능평가는 총 50번의 시도를 하였으며 실내 위치인자망에서의 채팅 메시지의 전송은 모두 성공하였다. 따라서 전송률은 100%로 측정되었으며 전송 시간 분포는 그림 10과 같다. 평균 전송 시간은 약 1.3초이며 최소 전송 시간과 최대 전송 시간은 각각 약 0.006초, 2.7초로 나타났다. 단, 이 때 전송 시간은 RF 프록시에서 스마트폰까지 채팅 메시지가 전송되는 시간이 아니라 RF 프록시에서 생성된 메시지가 스마트폰에 전송된 후, 응답 메시지 수신되는 시간인 채팅 메시지의 왕복시간(RTT)을 의미한다. 왜냐하면 RF 프록시와 스마트폰 간의 메시지 전송은 서로 다른 기기 간의 메시지 전송이기 때문에 기기 간의 시간 동기화가 어렵기 때문이다.

두 번째 성능평가는 첫 번째 성능평가에서 슬림허브의 개수와 스마트 기기의 개수를 다르게 하고 나머지 조건은 모두 동일하게 하여 진행하였다. 슬림허브가 1대이고 스마트 기기가 3대인 경우는 모든 스마트 기기가 하나의 슬림허브에 연결되어 있는 경우이고 슬림허브가 3대이고 스마트 기기가 3대인 경우는 각각의 스마트 기기가 각각 서로 다른 슬림허브에 하나

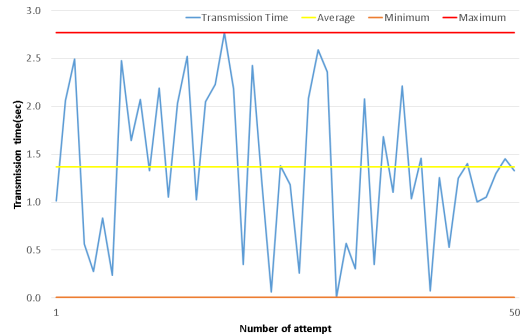


그림 10. 실내 위치인자망에서의 RF 프록시에서 하나의 스마트 폰으로 전송되는 채팅 메시지의 전송 시간 분포  
Fig. 10. Distribution of transmission time per attempt from RF Proxy to a smart phone in indoor localizing network



표 4. 슬림허브를 사용한 채팅 메시지의 성공률  
Table 4. Success rate of chatting message using SLIM-Hub

Division	Number of SLIM-Hub	1	1	3
	Number of Smart device	1	3	3
Number of total transmitted messages		50	150	150
Success rate of transmission(%)		100	100	97.33

씩 연결된 경우를 의미한다. 스마트 기기가 100개인 경우는 실제 기기를 연결하여 테스트하기 어렵기 때문에 가상의 프로세스를 스텝으로 대체하여 성능평가를 진행하였다. 표 4는 슬림허브를 사용한 채팅 메시지의 성공률을 나타낸다.

슬림허브의 개수가 3개일 때, 전송률이 조금 감소하는 것으로 나타났다. 이는 현재 성능평가가 수행된 네트워크 환경이 무선 Wi-fi망이기 때문에 감소한 것으로 추정된다. 슬림허브를 여러대를 사용할 때는 슬림허브를 1개 사용했을 때보다 메시지가 전송되는 경로가 다소 길어져 메시지의 전송 손실이 일어날 확률이 커진다. 그 이유는 하나의 슬림허브로 메시지가 전송되면 다른 이웃된 슬림허브로 메시지가 전송되게 되고 이는 분산형 서버의 단점이기도 하다. 하지만 이 단점은 클라이언트가 작은 경우에만 해당하며 많은 수의 클라이언트가 서비스를 사용하게 될 때에는 중앙집중형의 서버보다 안정적인 서비스의 제공이 가능하다.

### V. 결 론

본 논문에서는 자율군집 소프트웨어 플랫폼 환경에서의 단위공간 기반 실내위치인지망 구축 및 사물인터넷을 활용한 채팅 에이전트에서의 기기 제어에 대하여 연구하였으며 그 구조를 제안하였다. 슬림허브 중심의 실내위치인지망을 구축하여 개인정보의 외부 유출을 방지하고 실내위치인지망안에서 채팅 에이전트를 통해 스마트 기기 간 채팅뿐만 아니라 스마트 기기와 서비스 기기간의 선택적 채팅도 가능하다. 또한 하나의 서버집중형이 아닌 분산된 슬림허브 중심의 서버를 통해 보다 안정적인 서비스를 제공함으로써 채팅 메시지의 전송률에 있어서 더 높은 성능을 보임을 검증하였다.

향후 연구로는 어떤 서비스 기기의 서비스 상태에

따라 또 다른 서비스 기기를 제어할 수 있는 서비스 기기 간의 채팅에 대한 연구가 필요하며 궁극적으로 실내위치인지망내에 있는 기기들 간의 자율군집적인 상호작용이 이루어지는 플랫폼에 대한 연구가 필요하다.

### References

- [1] G. Park, M. Gang, and S. Son, "Overview of cyber physical systems and analysis of issue relating to network," *Inf. Commun. Mag.*, vol. 29 no. 5, pp. 98-105, Apr. 2012.
- [2] J. Jeon, N. Kim, J. Ko, T. Park, H. Kang, and C. Pyo, "Device product and technology trend of IoT," *Inf. Commun. Mag.*, vol. 31, no. 4, pp. 44-52, Mar. 2014.
- [3] D. Kim, H. Shin, and H. Ryu, "Study on preference of healthcare smart home system," *J. Architectural Inst. Korea, Planning and design*, vol. 28, no. 5, pp. 151-158, May. 2012.
- [4] D.-M. Shin, D.-I. Shin, and D.-K. Shin, "Development of u-health care system for dementia patients," *J. KICS*, vol. 38, no. 12, pp. 1106-1113, Dec. 2013.
- [5] J. Im, "Implementation of personal u-healthcare device and cuffless sphygmomanometer which is shape of watch," *J. IEIE*, vol. 40, no. 1, pp.39-49, Jan. 2013.
- [6] J. Hwang, *Smart age, digital secretary LG 'Home Chat' for you*, Retrieved Apr, 2014, from [http://social.lge.co.kr/lg\\_story/the\\_blog/product/homechat/](http://social.lge.co.kr/lg_story/the_blog/product/homechat/)
- [7] B. Kim and D. Jo, "Trend and outlook of network security technology," *Inf. Commun. Mag.*, vol. 31, no. 4, pp. 99-106, Mar. 2014.
- [8] S. H. Park and J. S. Cho, "Design of hierarchical u-LifeCare system using surrogate," in *Proc. KIISE Korea Computer Congress 2007*, pp. 401-405, Muju, Korea, Jun. 2008.
- [9] D. Ahn and R. Ha, "Indoor localization methodology based on smart phone in home environment," *J. KICS*, vol. 39, no. 4, pp. 315-325, Apr. 2014.
- [10] H.-Y. Kang, S.-Y. Jeong, C.-S. Ahn, Y.-J.

Park, and S.-J. Kang, "Self-organizing middleware platform based on overlay network for real-time transmission of mobile patients vital signal stream," *J. KICS*, vol. 38, no. 7, pp. 630-642, Jul. 2013.

- [11] S. Y. Jeong, H. G. Jo, and S. J. Kang, "Fully distributed monitoring architecture supporting multiple trackers and trackers in indoor mobile asset management application," *Sensors 2014*, vol. 14, pp. 5702-5724, Mar. 2014.

**이 성 희 (Sunghee Lee)**



2012년 : 경북대학교 컴퓨터학부 (학사)  
 2014년~현재 : 경북대학교 컴퓨터학부 석사과정  
 <관심분야> 임베디드 소프트웨어 테스트, 분산 컴퓨팅 환경 테스트 등

**정 설 영 (Seol Young Jung)**



2001년 2월 : 동아대학교 컴퓨터공학과(학사)  
 2009년 2월 : 계명대학교 전산교육학과(석사)  
 2014년 2월 : 경북대학교 전자공학부(박사)  
 2001년~2004년 : (주)유진로봇 연구원

2005년~2008년 : (주)맥산 선임연구원  
 <관심분야> 분산 미들웨어 시스템, 임베디드 시스템, 실시간 운영체제, 무선 센서 네트워크

**강 순 주 (Soon Ju Kang)**



1983년 2월 : 경북대학교 전자공학과 공학사  
 1985년 2월 : 한국과학기술원 전자계산학과 공학석사  
 1995년 2월 : 한국과학기술원 전자계산학과 공학박사  
 1985년~1996년 : 한국원자력연

구소 연구원, 핵인공지능연구실 선임연구원, 전산정보실 실장  
 1996년~현재 : 경북대학교 전자전기컴퓨터학부 정보통신공학전공 정교수  
 2011년~현재 : 경북대학교 자율군집소프트웨어 연구센터장  
 <관심분야> 자율군집 소프트웨어 시스템, 실시간 시스템, 소프트웨어 공학, 지식기반 시스템

**이 우 진 (Woo Jin Lee)**



1992년 : 경북대학교 전자계산학과(학사)  
 1994년 : KAIST 전산학과(석사)  
 1999년 : KAIST 전산학과(박사)  
 1999년~2002년 : ETRI 선임연구원

2002년~현재 : 경북대학교 IT대학 컴퓨터학부 교수  
 <관심분야> 임베디드 소프트웨어 테스트, 임베디드 소프트웨어 개발환경 분석 등