

Implementation of Workflow Based Smart Building Services

Lee JaeHee[†] · Kim DooYoung^{††} · Song Seheon^{†††} · Lee Sangil^{††††} · Park JaeHyun^{††††}

ABSTRACT

With the development of IoT technology, the scope of object controls and sensor information has expanded and services have been diversified. Energy saving efforts in buildings with high energy consumption is increasing, and services are being extended over life safety, security and convenience. However, Existing smart building services are dependent on IoT devices and are difficult to change or expand. In this paper, we have implement of workflow based smart building services to modify or extend services according to user's needs.

Keywords : Workflow, BPMN, Internet of Things, Smart Building Services

워크플로우 기반 스마트 빌딩 서비스 구현

이 재 희[†] · 김 두 영^{††} · 송 세 현^{†††} · 이 상 일^{††††} · 박 재 현^{††††}

요 약

IoT 기술 발전으로 사물 제어와 센서 정보 영역이 확대되고 서비스 또한 다양해지면서 에너지 소비가 높은 빌딩에서의 에너지 절감 노력이 점차 증가하고 있으며 생활 안전, 보안, 편의에 이르기까지 서비스가 확대되고 있다. 하지만 기존 스마트 빌딩 서비스들은 IoT 디바이스에 종속적이고, 사용 중인 서비스 변경 또는 확장이 어렵다. 본 논문에서는 사용자의 요구에 따라 서비스를 변경하거나 확장할 수 있도록 워크플로우 기반의 스마트 빌딩 서비스를 구현하였다.

키워드 : 워크플로우, BPMN, 사물인터넷, 스마트 빌딩 서비스

1. 서 론

ICT 기술 발전과 사물 인터넷(Internet of Things)의 발전으로 다양한 IoT 디바이스들이 개발되어 이를 활용한 스마트 홈은 개인에게 서비스할 수 있는 수준에 이르렀지만 단순한 디바이스의 제어가 대부분을 차지하고 있다.

최근 가트너의 2017년 10대 전략 기술 트렌드 발표 중[1] Intelligent Things에서처럼 IoT 디바이스의 단순한 제어가 아닌 지능형 사물로의 전환이 필요하게 되었으며 이러한 움직임은 에너지 소비가 가장 많은 제2의 생활공간인 빌딩에 까지 확대되었다.

2017년부터 공공기관의 에너지 저장 장치, 건물에너지관리시스템 설치를 주요 내용으로 하는 공공기관 에너지 이용

합리화 추진에 관한 정책이 시행 중이며 심야 전기 또는 자체 생산된 전력을 비축한 뒤 전력소비가 큰 시간에 여러 가지 데이터를 기반으로 최적의 운영조건을 찾아 전력소비를 최소화하는 것으로 이러한 정책은 에너지 측면의 스마트 빌딩 서비스로 볼 수 있다.

스마트 빌딩 서비스는 에너지 측면뿐 아니라 출입통제를 통한 보안, 화재감시와 대피로 안내 등의 생활 안전, 편의에 이르기까지 영역이 확대되고 있다[2].

하지만 기존 스마트 빌딩 서비스들은 IoT 디바이스에 종속적이며 사용 중인 서비스를 변경하거나 확장하는데 소프트웨어 개발자의 프로그램 변경이 불가피하며 결국 비용 발생으로 이어지게 된다.

스마트 빌딩 서비스 수정 및 확장에 대한 요구사항을 충족시키고 개발시간 단축과 비용절감을 위해서는 디바이스의 재사용과 함께 수정 및 확장에 대한 유연성을 제공할 수 있는 스마트 빌딩 서비스 개발 방법이 필요하다. 이런 요구사항을 만족할 방안으로 워크플로우 기반으로 동작하는 스마트 빌딩 서비스 개발 및 실행 방법을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서 관련 기술 및 기

※ 이 논문은 민간기술협력사업(UMI3018RD1)으로 지원받았음.
† 정 회 원 : 부울(주) 책임연구원
†† 비 회 원 : 부울(주) 수석연구원
††† 정 회 원 : 메타빌드(주) 수석연구원
†††† 비 회 원 : 국방과학연구소 책임연구원
Manuscript Received : February 22, 2017
Accepted : March 17, 2017
* Corresponding Author : Park JaeHyun(forehand@add.re.kr)

존 스마트 빌딩 서비스 연구를 소개하고 3장에서는 워크플로우 기반의 스마트 빌딩 서비스의 시스템 설계 및 구현에 관한 내용을 기술하고 있으며 4장에서는 테스트베드를 구축, 5장에서는 테스트베드를 이용한 워크플로우 기반의 스마트 빌딩 서비스를 검증하고 6장 결론으로 끝을 맺는다.

2. 관련 연구

2.1 워크플로우 기술

기업의 업무 분야가 다양하고 복잡해짐에 따라 개인이 가지고 있는 지식의 공유와 정형화된 업무가 필요해 지면서 업무의 시각적인 표현 및 문서화가 요구되었다. 이를 지원하기 위해 소프트웨어 회사로 구성된 협회인 BPMI (Business Process Management Initiative)에서 BPMN (Business Process Modeling Notation)을 개발하였고 현재는 OMG (Object Management Group)에 의해 관리되고 있으며 2011년 BPMN2.0[3]이 나오게 되면서 BPMN 표준기법으로 작성된 업무를 BPEL (Business Process Execution Language)로 변환 작업 없이 실행 엔진에 배포하여 실행할 수 있게 되었다.

스마트 빌딩 서비스에서도 사용자에게 서비스 변경 및 확장기능을 쉽게 제공하기 위해 서비스의 가시화가 필요하였으며 제작된 스마트 빌딩 서비스를 실행할 수 있도록 워크플로우 기술을 사용하였다.

2.2 스마트 빌딩 서비스 및 서비스 조합 기술

기존 수행된 스마트 빌딩 서비스 관련 연구[4, 5]들은 서비스 제공자가 사용자의 요구사항을 고려해 디바이스와 함께 서비스를 개발하므로 디바이스가 서비스에 종속적이며, 서비스 개발이 완료된 후에는 서비스의 변경 및 확장이 어렵다.

IoT 서비스 지능화를 위한 디바이스 오브젝트화 및 오케스트레이션 메커니즘 연구[6]에서는 물리적인 디바이스를 오브젝트화하고 웹서비스로 제공하고 각각의 서비스를 오케스트레이션 하는 기술을 연구하였는데 디바이스를 오브젝트화하여 재사용하고 서비스를 조합하는 것은 가능하지만, 전문적인 소프트웨어의 지식이 요구된다.

Table 1은 제안하는 워크플로우 기반의 스마트 빌딩 서비스와 연구된 서비스를 비교 설명하고 있으며 본 논문에서

Table 1. Comparison of Workflow Based Smart Building Services and Related Research Services

Comparison Item	Workflow based smart building services	Researched smart building services
Service combination	possible	possible
Technical knowledge	unnecessary	necessary
Modification and expansion	possible	It is possible but difficult
Device reuse	possible	Depends on the developed service

는 전문적인 소프트웨어 지식이 없더라도 서비스가 개발 완료된 이후에도 사용자의 필요에 따라 스마트 빌딩 서비스를 수정 및 확장할 수 있도록 하였다.

3. 시스템 설계 및 구현

3.1 스마트 빌딩 서비스 구조

워크플로우 기반의 스마트 빌딩 서비스는 Fig. 1과 같은 계층형 구조를 가지며 스마트 빌딩 서비스를 이루고 있는 Task (작업의 단위)는 각각의 디바이스 서비스와 연결되어 있고 디바이스 서비스는 실제 물리적인 디바이스와 연결되어 있다.

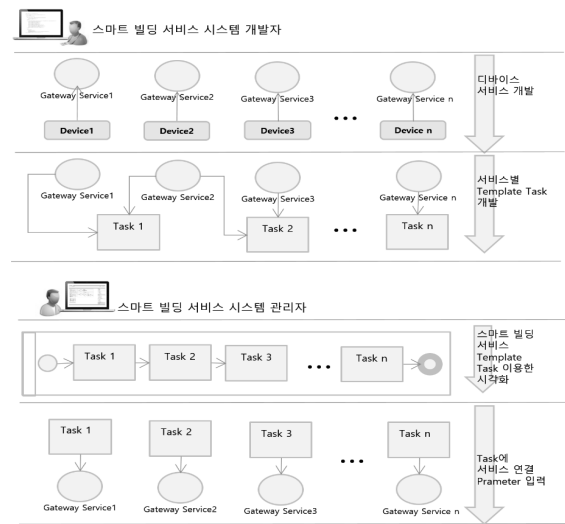


Fig. 1. The Smart Building Service Diagram

3.2 스마트 빌딩 서비스 개발 흐름도

스마트 빌딩 서비스에 필요한 시스템 구성요소는 스마트 빌딩 서비스 모델러, 스마트 빌딩 서비스 실행 엔진, 게이트웨이가 있으며 3.4절에서 상세히 기술한다.

스마트 빌딩 서비스는 시스템 개발자와 시스템 사용자로 나뉘며 시스템 개발자는 게이트웨이에 디바이스를 서비스화하여 등록하고 각 디바이스 타입 별 사용할 Task 템플릿을 정의하는 역할을 수행한다.

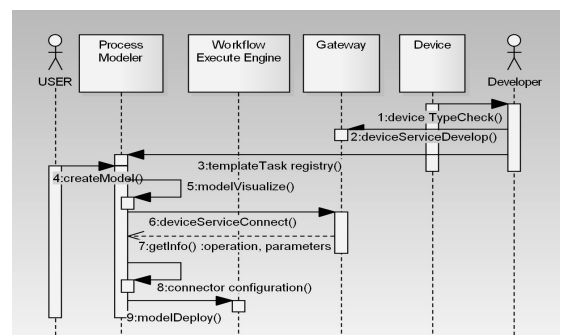


Fig. 2. The Service Development Sequence Flow

스마트 빌딩 서비스 시스템 사용자는 스마트 빌딩 서비스를 제작할 수 있도록 지원하는 모델러를 사용하여 미리 정의된 Task 템플릿으로 스마트 빌딩 서비스를 가시화하고 개발된 디바이스 서비스 연결 및 호출 시 필요한 정보를 입력하여 스마트 빌딩 서비스를 만들고 실행 엔진에 배포하는 역할을 한다. Fig. 2는 역할에 따른 개발 흐름도를 보여주고 있으며 Table 2는 상세 설명이다.

Table 2. Service Development Sequence Flow Description

Role	Flow	Description (Item)
Developer	To develop a device service	Device service development and gateway registration (1-2)
	To develop a task template	Development of task template for each device type and registration to modeler (3)
User	To create a smart building service model	Create a smart building service model(4)
	To visualize a smart building service model	Visualization using template task and BPMN2.0 element(5)
	To connect device service	Device service connection and operation, parameter setting (6-8)
	To deploy a smart building service	Deploying smart building services to the execution engine (9)

3.3 스마트 빌딩 서비스 모델링 방법

제안하는 스마트 빌딩 서비스 모델링 방법은 스마트 빌딩 서비스 가시화를 위해 서술 수준(descriptive level)의 모델링 기법과 실행 엔진에 배포되어 실행할 수 있도록 실행 수준(executable level)의 모델링 기법을 사용하였다.

스마트 빌딩 서비스는 소프트웨어 지식이 없는 사용자도 모델링이 가능하도록 스마트 빌딩 서비스 가시화를 위해 서술 수준의 모델링을 BPMN 2.0 element[7] 중 Table 3의 element 만을 테일러링하여 사용하였으며 Task의 캡슐화를 통해 모델링을 단순화하였고 실행 수준의 모델링에서는 게이트웨이와 연결할 수 있도록 했다.

1) 서술 수준(descriptive level)의 모델링

서술 수준의 모델링 방법은 스마트 빌딩 서비스를 가시화 하는 과정으로 BPMN2.0의 모든 element를 사용하여 모델링을 하면 복잡성 증가와 자유도가 높아 각 개인에 따라 모델링이 달라지며 이해 또한 쉽지 않아 Table 3의 element만을 이용하여 정형화된 템플릿 형태의 Task를 미리 정의하고 사용자의 용도에 맞도록 적절히 사용하는 모델링 방법을 제안하게 되었으며 Fig. 3과 Fig. 4의 예시를 통해 element 중 Sub-Process를 이용하여 모델링을 단순화하는 방법을 설명하고 있다.

Table 3. BPMN2.0 Element Used In the Process Modeler

Attributes	Element	Notation
Event	Start	○
	End	●
	Message Start	✉
	Timer Start	🕒
Activity	User Task	☞
	Service Task	⚙️
	Sub-process	▭
Gateway	Exclusive	⊗
	Parallel	⊕
Connector	Sequence Flow	→
Participant	Pool, Lane	▭

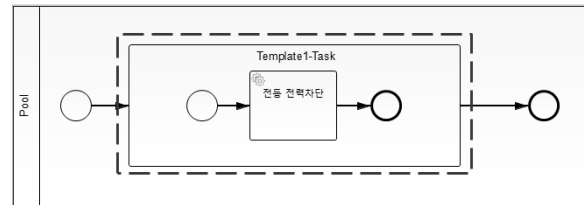


Fig. 3. Example of a Simple Expanded Sub-Process

Fig. 3은 전등 전력을 차단하는 Task로 Sub-Process가 펼쳐진 모습이며 내부는 게이트웨이의 디바이스 서비스나 애플리케이션과 같은 외부 서비스를 실행하는 Service Task로 구성되어 있다.

Fig. 4에서는 전등 전력 차단 여부를 사용자에게 승인 요청을 하는 내용이 Sub-Process 안에 포함되어 있다. 이처럼 Sub-Process를 이용하여 템플릿을 미리 정의하고 사용함으로써 사용자는 사용할 템플릿을 선택하고 테일러링된 element만을 사용하여 서술 수준의 모델링을 완성할 수 있다.

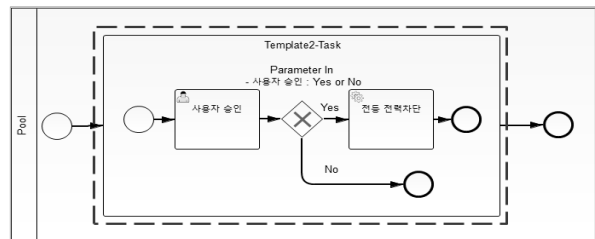


Fig. 4. Example of the Expanded Sub-Process for the User Task

2) 실행 수준(executable level)의 모델링

서술 수준의 모델링에서 템플릿 Task 내부에 포함되어 있는 Service Task에서는 디바이스 서비스 혹은 애플리케이션과 같은 외부 서비스를 연결 또는 호출하는 기능을 수행하는데 이때 필요한 정보들을 입력하여 모델링을 완성할 수 있다.

3.4 스마트 빌딩 서비스 시스템 구성도

본 연구에서 제안하는 워크플로우 기반 스마트 빌딩 시스템은 Fig. 5와 같이 스마트 빌딩 서비스를 실행할 수 있는 실행 엔진, 스마트 빌딩 서비스를 가시화하고 제작할 수 있으며 실행 엔진에 배포할 수 있는 기능의 모델러, 디바이스 연결 및 관리를 위한 게이트웨이로 구성된다.

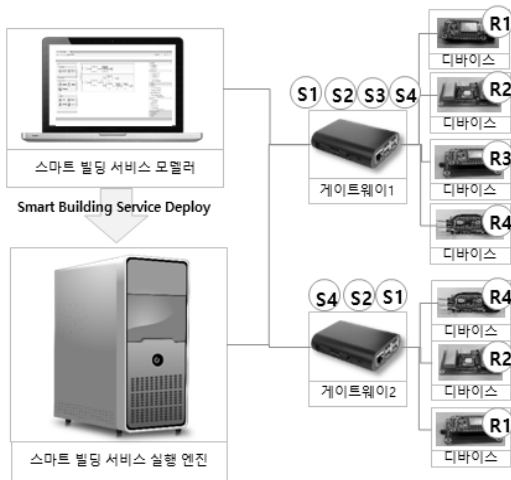


Fig. 5. The Smart Building Service System Diagram

1) 스마트 빌딩 서비스 모델러

스마트 빌딩 서비스 모델러는 사용자가 스마트 빌딩 서비스를 제작 및 수정할 수 있는 웹 기반 모델링 도구이다.

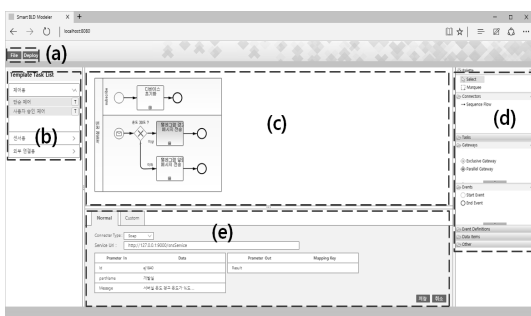


Fig. 6. The UI of the Smart Building Service Modeler

Fig. 6은 스마트 빌딩 서비스 모델러 화면으로 (a)는 파일 로딩 및 저장과 스마트 빌딩 서비스를 실행 엔진에 배포하는 기능, (b)는 제공되는 템플릿 Task의 목록 선택, (c)는 스마트 빌딩 서비스를 드로잉 하는 공간, (d)는 사용할 수 있는 BPMN2.0 elements 선택, (e)는 Service Task,

User Task 실행 시 필요한 정보를 입력할 수 있는 기능을 제공하는 UI 형태로 구성된다.

2) 스마트 빌딩 서비스 실행 엔진

스마트 빌딩 서비스 실행 엔진은 스마트 빌딩 서비스 모델러로 개발된 스마트 빌딩 서비스 실행할 수 있는 엔진으로 라이프 사이클 관리, 실행상태 모니터링 기능을 제공하며 Fig. 7과 같이 Listeners, Process Manager, Service Connectors로 구성된다.

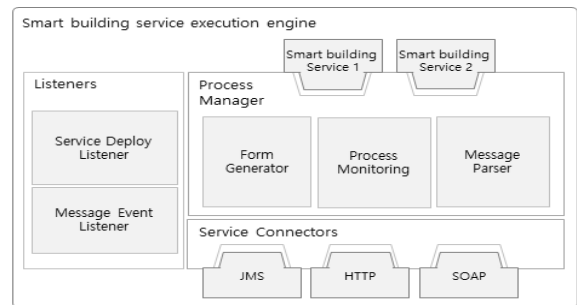


Fig. 7. The Smart Building Service Execution Engine

a) Listeners

Listeners는 외부의 요청을 수행하는 기능을 하며 모델러에서 배포 요청 시 실행 엔진의 중지 없이 배포하는 기능을 하는 Service Deploy Listener와 배포된 스마트 빌딩 서비스에서 Event를 수신할 수 있는 Message Event Listener가 있다.

b) Service Connectors

Service Connectors는 스마트 빌딩 서비스 실행 엔진에서 외부와의 통신 기능을 제공하며 게이트웨이 또는 외부 애플리케이션과 통신을 위해 사용된다. 스마트 빌딩 서비스 실행 엔진에 배포된 스마트 빌딩 서비스에서 Service Task 실행 시 모델링 할 때 선택된 Service Connector 및 설정된 정보를 이용하여 실행 엔진의 Service Connector를 사용하게 된다.

c) Process Manager

Process Manager는 스마트 빌딩 서비스의 시작, 중지, 회수 등의 라이프 사이클 관리를 위해 필요하며 서비스 실행 시 필요한 Form Generator, Message Parser와 상태 모니터링을 위한 Process Monitoring 있다.

d) Form Generator

Form Generator는 배포된 스마트 빌딩 서비스 실행 시 User Task를 사용하여 입력 값 요청이 있으면 배포된 스마트 빌딩 서비스에 설정된 스마트 빌딩 모델 파일에서 모델링 시 설정한 Fig. 8의 User Task 정보를 실행 엔진에서 사용할 수 있도록 Html Form으로 변환하여 Fig. 9 처럼 사용자에게 보여지게 된다.

```

<bpmn2:userTask id="UserTask_1" name="전력공급 승인">
  <bpmn2:extensionElements>
    <user:formData>
      <user:formField id="k" label="전등 켜시켰습니까?" type="enum">
        <user:value id="Y" name="예"/>
        <user:value id="N" name="아니요"/>
      </user:formField>
    </user:formData>
  </bpmn2:extensionElements>
</bpmn2:userTask>
    
```

Fig. 8. Example of Smart Building Service Model File



Fig. 9. Example of the User Task Form

e) Message Parser

Message Parser는 디바이스 서비스 또는 외부 애플리케이션 서비스를 호출할 때 모델링 시 설정된 정보를 가지고 요청 메시지를 만들거나 응답 결과를 매핑할 때 사용된다.

f) Process Monitoring

Process Monitoring은 현재 배포되어 실행 중인 스마트 빌딩 서비스의 상태 정보를 보여주는 기능으로 Fig. 10에서 화면의 상단의 리스트는 배포된 스마트 빌딩 서비스 목록이며 아이디, 이름, 디플로이 시간, 실행 중인 인스턴스의 수, 실패, 성공, 총 실행 횟수에 대한 정보를 보여주고 있다.

스마트 빌딩 서비스 리스트 중 하나를 선택하면 해당 스마트 빌딩 서비스의 가시화된 모델이 나오는데 현재 실행 중인 인스턴스의 Task 위치가 표시된다.



Fig. 10. The Process Monitoring

3) 게이트웨이

게이트웨이는 디바이스 연결 및 관리를 위하여 필요하며 OSGi (Open Service Gateway initiative) 기반으로 사물을 인터넷에 연결할 수 있도록 다양한 통신 방식(WiFi, Bluetooth, ZigBee 등)을 지원한다.

Fig. 11은 게이트웨이 구성도이며 통신 방식 선택을 위한 Field Protocol Adaptor, 디바이스 서비스 관리를 위한 Service Manager, 외부에서의 접속할 수 있도록 하는 Service Connector, 디바이스의 서비스화를 위한 Things Service Applications로 구성된다.

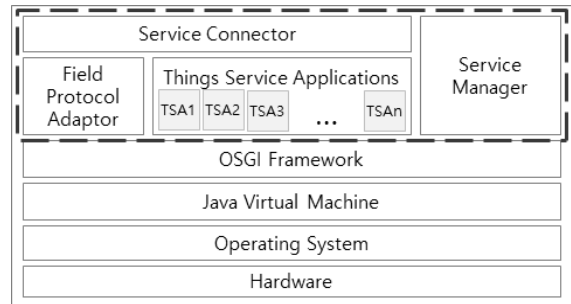


Fig. 11. The Gateway Architecture

4. 테스트베드 구축

4.1 디바이스 제작

테스트베드 구축을 위해 일반 빌딩에 설치할 IoT 디바이스를 제작하였다. 제작된 디바이스는 센서/액추에이터를 제어하는 게이트웨이, 전등 전력 제어기, 출입문 제어기, 인체 감지기 등이며 Table 4와 같다.

Table 4. Devices Developed for Smart Building Services

Name	Figure	Purpose of use
Gateway		Sensor, Actuator management
Lamp circuit controller		Lamp circuit control(on, off)
Door controller		Door control (open, close)
Human detection sensor		Human detection

4.2 디바이스 서비스 개발

스마트 빌딩 서비스 개발에 앞서 디바이스 재사용을 위해 디바이스 서비스화가 필요하였으며 개발된 디바이스 서비스는 Table 5와 같다.

Fig. 12는 게이트웨이에 등록된 출입문 제어 서비스 명세 파일이며 (a)는 Field Protocol Adaptor, (b)는 Things Service Applications (c)는 Service Connector 내용이다.

Table 5. Device service list

Service name	Data(in, out)	Device
Door control service	in : operation(open, close, status) out : result (Y, N)	Door controller
Lamp circuit control service	in : operation(on, off, status) out : result (Y, N)	Lamp circuit controller
Human detection service	in : operation(detect) out : result(Y, N)	Human detection sensor

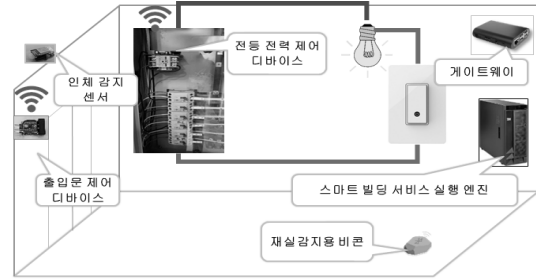


Fig. 14. The Smart Building Test Bed Diagram

```
<?xml versj "1.0" encoding="UTF-8"?>
<blueprint xmlns="http://www.osgi.org/xmlns/blueprint/v1.0.0"
.....
default-activation="eager">
<bean id="tcpClient" class="com.mesim.arana.tcp.TCPClient">
(a) <argument value="192.168.1.18" />
<argument value="7300" />
</bean>
<bean id="doorMf" class="com.mesim.arana.sb.Door"
init-method="start" destroy-method="stop">
<argument ref="blueprintBundleContext"/>
<property name="thingId" value="1" />
<property name="deviceId" value="1" />
<property name="soap" value="true" />
<property name="io" ref="tcpClient" />
(b) <property name="role" value="257" />
<property name="METHOD_OPEN" value="Door:open" />
<property name="METHOD_CLOSE" value="Door:close" />
<property name="METHOD_STATUS" value="Door:status" />
<property name="autoClose" value="true" />
<property name="autoCloseInterval" value="5000" />
</bean>
<jaxws:endpoint
(c) implementor="#doorMf" address="/things/door/1"
publishedEndpointUrl="http://0.0.0.0:8181/cxf/things/doorMf"
implementorClass="com.mesim.arana.sb.Door" />
</blueprint>
```

Fig. 12. Example of the Door Control Service File

```
192.168.1.15:8080/96312cff32e4494998922a2694df0b78:8181/cxf/things/
This XML file does not appear to have any style information associated with it. The doc
<?xml version="1.0" encoding="UTF-8"?>
<definitions xmlns:sd="http://www.w3.org/2001/XMLSchema" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
targetNamespace="http://sb.arana.mesim.com/">
<import location="http://0.0.0.0:8080/96312cff32e4494998922a2694df0b78:8181/cxf/thir
<binding name="ElectricSwitchServiceSoapBinding" type="ns1:ElectricSwitchSOAP">
<soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
<wsdl:operation name="off">
<soap:operation soapAction="" style="document"/>
<wsdl:input name="off">
<soap:body use="literal"/>
</wsdl:input>
<wsdl:output name="offResponse">
<soap:body use="literal"/>
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="on">
<soap:operation soapAction="" style="document"/>
<wsdl:input name="on">
<soap:body use="literal"/>
</wsdl:input>
<wsdl:output name="onResponse">
```

Fig. 13. WSDL of the Lamp Circuit Control Service

Fig. 13은 웹 브라우저를 통해 운용 중인 전등 전력 제어 서비스에 접근하여 해당 서비스의 WSDL 파일을 확인한 결과이다.

4.3 구축범위

테스트베드 구축 범위는 Fig. 14와 같이 1개의 층에 게이트웨이 및 전등 전력제어 디바이스와 출입문 제어 디바이스, 인체감지 센서를 각 배전반과 출입문에 설치하고 스마트 빌딩 서비스 실행 엔진 탑재를 위한 서버를 설치하였다.

5. 검증

시나리오를 기반으로 테스트베드를 이용하여 스마트 빌딩 서비스를 확장하는 것으로 검증한다.

<시나리오>

스마트 빌딩 서비스 중 재실감지 서비스, 특정 시간 전등 전력제어(Light On)서비스가 스마트 빌딩 서비스 실행 엔진에 배포되어 실행 중이며 재실 인원 감지 서비스에 특정 시간 전등 전력제어(Light On)서비스를 결합하여 새로운 스마트 빌딩 서비스 모델을 제작하는 것으로 검증한다.

Fig. 15는 재실감지 서비스 모델이며 Message Start Event를 통해 모바일 디바이스 앱으로부터 비콘 ID, 사용자 정보가 수신되면 애플리케이션으로 전송하는 서비스로 서버 애플리케이션에서는 수신 정보를 저장하여 사용자에게 제공한다.

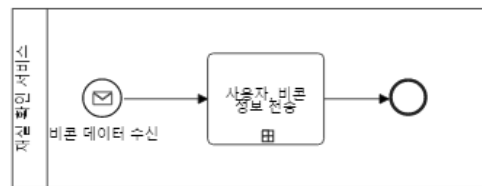


Fig. 15. The Process Model of the People Detection Service Using Beacon

Fig. 16은 특정 시간 전등 전력제어(Light On)서비스 모델이며 매일 6시에 Timer Start Event가 발생 되어 전력제어(Light On) 서비스를 호출한다.

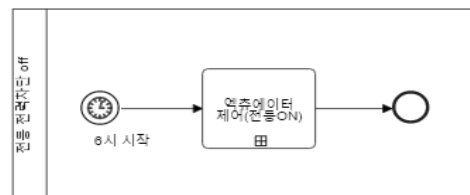


Fig. 16. The Process Model of the Light Control Service Using a Circuit Breaker Controller

Fig. 17은 재실감지 서비스 Fig. 15를 변경하여 확장한 서비스로 검증을 위한 스마트 빌딩 서비스이다.

확장된 서비스는 6시에 맞춰 전등 전력제어(Light On)를 하거나 6시 이전이라도 재실 인원이 감지되면 전등 전력제어(Light On)를 하는 기능을 한다.

또한, 인체감지와 출입문 제어 디바이스 서비스를 활용하여 사용자 비콘 정보가 수신된 직후 출입구에 설치된 인체감지 센서에 인체가 감지되면 출입문까지 열리는 기능을 추가하였다.

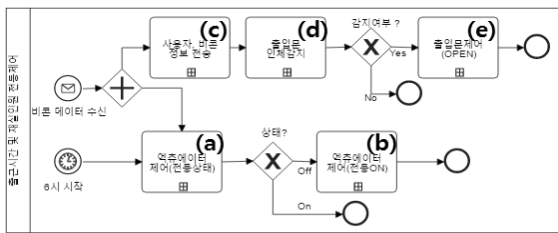


Fig. 17. The Process Model of the Expanded Smart Building Service



Fig. 18. The Smart Building Service Execution Engine Log

Fig. 18은 서비스가 배포되어 실행되고 있는 스마트 빌딩 실행 엔진의 로그이며 Fig. 17의 서비스가 성공적으로 실행되었음을 확인할 수 있다.

이처럼 Fig. 17을 통해 본 논문에서 제안하는 방법인 워크플로우 기반의 스마트 빌딩 서비스를 사용자의 요구에 따라 개발이 완료된 이후에도 변경 및 확장할 수 있는 것을 검증하였다.

6. 결론

본 논문에서 제시한 워크플로우 기반의 스마트 빌딩 서비스는 스마트 빌딩 서비스의 변경 및 확장에 대한 한계를 극복하고 스마트 빌딩 서비스 모델러를 이용하여 요구사항을 반영한 사용자 중심의 스마트 빌딩 서비스를 제작하거나 서비스 변경 및 확장 시 스마트 빌딩 서비스 모델을 정의, 수정하는 것만으로 가능하다는 것을 확인하였다.

본 연구를 활용한다면 사용자 중심의 스마트 빌딩 서비스를 제작하여 에너지 절감, 생활안전, 보안, 편의 등 다양한

분야에 적용할 수 있으며 사용 중인 스마트 빌딩 서비스 또한 요구 조건에 따라 변경 및 확장이 자유로워 유연한 대처가 가능할 것으로 보인다.

References

- [1] Gartner's Top 10 Strategic Technology Trends for 2017 [Internet], <http://www.gartner.com/smarterwithgartner/gartner-top-10-technology-trends-2017/>.
- [2] Y. C. Baek, and H. S. Kim, "Construction IT Convergence Technology: Smart Building Technology Development Case," *Korean Institute of Communications and Information Sciences*, Vol.30, No.10, pp.25-31, 2013.
- [3] Business Process Model And Notation Version 2.0 Release Date: January 2011 [Internet], <http://www.omg.org/spec/BPMN/2.0/>.
- [4] S. I. Hwang, T. J. Park, Y. K. Sohn, and G. P. Jeon, "Smart Grid Use Case and Service Requirement Based on M2M: Energy Management System for Public Buildings," *The Journal of The Korean Institute of Communication Sciences*, Vol.38C, No.7, pp.612-620, 2013.
- [5] H. Y. Bahn, "Efficient Scheduling of Sensor-based Elevator Systems in Smart Buildings," *Journal of the Korea Academia-Industrial cooperation Society*, Vol.17, No.10, pp. 367-372, 2016.
- [6] Y. J. Kim, Y. K. Jeon, and I. Y. Chong, "Device objectification and orchestration mechanism for IoT intelligent service," *The Journal of Korean Institute of Communications and Information Sciences*, Vol.38, No.1, pp.19-32, 2013.
- [7] BPMN 2.0 - Business Process Model and Notation [Internet], http://www.bpmb.de/images/BPMN2_0_Poster_EN.pdf.



이재희

e-mail : ej1840@metabuild.co.kr
 2010년 순천대학교 컴퓨터과학과(학사)
 2015년~현 재 부울(주) 책임연구원
 관심분야: 사물인터넷, 알고리즘, AI



김두영

e-mail : netdoo@buall.com
 2003년 동양대학교 컴퓨터공학부(학사)
 2016년 숭실대학교 IT정책경영학과(석사)
 2010년~현 재 부울(주) 수석연구원
 관심분야: 소프트웨어 공학, 웹 공학



송 세 현

e-mail : secheon.song@metabuild.co.kr
2001년 아주대학교 정보컴퓨터공학부(학사)
2003년 아주대학교 정보통신공학과(석사)
2013년 아주대학교 정보통신공학과(박사)
2014년~현 재 메타빌드(주) 수석연구원
관심분야: 서비스컴퓨팅, 사물인터넷,
시멘틱웹



박 재 현

e-mail : forehand@add.re.kr
1990년 경북대학교 컴퓨터공학과(학사)
1992년 경북대학교 컴퓨터공학과(석사)
1999년~현 재 국방과학연구소 책임연구원
관심분야: M&S, SOA, 사물인터넷



이 상 일

e-mail : happyjoy@add.re.kr
1994년 성균관대학교 정보공학과(학사)
1996년 성균관대학교 정보공학과(석사)
2010년 성균관대학교 컴퓨터공학과(박사)
1999년~현 재 국방과학연구소 책임연구원
관심분야: 상호운용성, 사물인터넷, SOA