

# 유니버설 서비스의 상호운용을 위한 양방향 DPWS 어댑터\*

임형준\*, 이규철\*†

\*충남대학교 컴퓨터공학과

e-mail : {hyungjun25, kclee}@cnu.ac.kr

## The Two-way DPWS Adaptor for Universal Service Interoperation

Hyung-Jun Yim\*, Kyu-Chul Lee\*

\*Dept. of Computer Engineering, Chung-Nam National University

### 요 약

유비쿼터스 환경에 산재하는 서비스 발견 미들웨어는 프로토콜, 표준, 서비스 검색 방법 및 프로그래밍 언어 등의 이질적인 특성을 가진다. 본 연구에서는 이질적인 서비스 발견 미들웨어의 상호운용을 지원하기 위해 유니버설 네트워크를 위한 웹서비스 프레임워크를 제안했다. 따라서 본 논문에서는 대표적인 미들웨어인 DPWS와 웹서비스의 상호운용을 지원하기 위해 양방향 DPWS 어댑터에 대한 설계 및 구현을 서술한다. 또한, 양방향 DPWS 어댑터의 구조, 구성 요소와 동작 과정을 설계하고, 간단한 시나리오를 구현함으로써 양방향 DPWS 어댑터가 제공하는 서비스 지향 아키텍처, 상호운용성 제공, 일반적인 웹서비스 호출, 자가 구성, 동적 서비스 검색 등의 이점을 논한다.

### 1. 서 론

최근 통신, 정보, 인터넷과 디지털 장치 등의 기술의 발전은 좀 더 편안한 생활을 위해 다양한 특징을 가지는 전자기기, 서비스, 미들웨어 등이 계속 생겨나고 있다. 이러한 추세로 인해 오디오, 비디오 장치 등의 홈 네트워크를 기반으로 프로세서가 탑재된 개인 전자기기 및 가전제품에 이르면서 차세대 이동통신, BcN, USN 등과 밀접한 연관을 가지고 있다. 하지만, 서로 다른 플랫폼, 네트워크, 프로토콜 등으로 구축된 시스템들의 이질적인 특성은 아직 문제점으로 남아 있다. 최근 이러한 문제점은 유비쿼터스 환경에 적합하도록 대표적인 Jini [1], UPnP [2], HAVi [3] 등과 전자기기 레벨의 경량화된 웹서비스 프로토콜을 지원하는 *Devices Profile for Web Services (DPWS)* [4]의 서비스 발견 미들웨어 (Service Discovery Middleware)를 통해 해결하고 있다. 불행하게도 미들웨어 역시 상이한 프로토콜, 표준, 프로그래밍 언어 및 보안 등을 기반으로 하기 때문에 각종 전자기기 및 서비스의 위치나 상태 등의 유동성에 의해 발생하는 문제를 기존 미들웨어 중 하나가 모두 반영하기는 어렵다. 예를 들면, 서비스 사용자는 Jini 나 HAVi 에서 제공하는 서비스를 사용하기 위해서는 각각에 해당하는 검색 방법이나 자바 RMI 와 IEEE 1394 의 통신 환경이 제공되어야 한다.

이러한 문제를 해결하기 위해 본 연구에서는 유니버설 네트워크를 위한 웹서비스 프레임워크 (*WSUN*,

*Web Services on Universal Networks*) [5]을 제안했다. 유니버설 네트워크를 위한 웹서비스 프레임워크는 서비스 지향 아키텍처 (SOA, Service-Oriented Architecture) 패러다임 기반의 유비쿼터스 웹서비스 프레임워크이다. 또한, 유니버설 서비스의 발견과 연동 등을 포함하는 프로토콜을 제시하고 있다.

본 논문에서는 *WSUN* 에서 다루고 있는 대표적인 미들웨어 중 *DPWS* 를 논하며 웹서비스와 상호운용을 지원하기 위해 양방향 *DPWS* 어댑터를 제안한다. 또한, 양방향 *DPWS* 어댑터의 구조, 구성 요소, 동작 순서와 단계 및 과정을 설계하고 구현한다.

본 논문의 구성은 다음과 같다. 2 장에서는 이질적인 미들웨어의 상호운용을 지원하기 위한 관련 연구를 소개하고, 3 장에서는 제안한 양방향 *DPWS* 어댑터의 개요를 설명한다. 양방향 *DPWS* 어댑터의 설계는 4 장에서 기술하며, 5 장에서는 설계에 따른 시나리오의 구현을 통해 양방향 *DPWS* 어댑터의 동작을 살펴본다. 마지막으로 6 장에서는 결론과 향후 연구방향에 대하여 논한다.

### 2. 관련 연구

이번 장에서는 본 논문에서 제안한 양방향 *DPWS* 어댑터와 관련된 연구를 다룬다. 최근에 *DPWS* 에 대한 연구가 진행되고 생겼기 때문에 이에 대한 직접적인 관련 연구는 없으나 이질적인 미들웨어의 상호운용성을 위한 기존의 연구에 대해 살펴본다.

\* 본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT 연구센터 육성·지원사업 (IITA-2008-C1090-0801-0031) 의 연구결과로 수행되었음.

† 교신 저자

## 2.1 OSGi 를 확장한 프레임워크 [6]

[7]은 이질적인 미들웨어의 통합을 Open Service Gateway Initiative (OSGi) [7]를 이용하여 서비스 프레임워크를 제공한다. 이는 OSGi 가 가지는 서비스 게이트웨이와 분산 미들웨어의 장점을 혼합한 형태이다. 컨트롤 서비스 (Control Services)를 통한 디바이스 연동을 위한 OSGi 확장한 구조로 OSGi 레지스트리와 디바이스 위치기, 디바이스 선택기로 구성된 OSGi 디바이스 매니저 등을 통해 미들웨어와 연결을 한다. [6]에서는 OSGi 의 게이트웨이가 가지는 특징을 이용하기 위해 미들웨어마다 컨트롤 서비스를 생성하여 OSGi 의 게이트웨이와 연결하는 인터페이스를 제공한다. 또한, 이벤트에 대한 처리를 담당하여 OSGi 의 프레임워크에서 처리하지 못하는 메시지를 변환하여 전달한다.

하지만, [6]은 OSGi 를 연동하기 위해 각 미들웨어마다 디바이스 드라이버 번들 (Device Driver Bundle)이 존재한다. 이 디바이스 드라이버 번들은 [6]에서의 컨트롤 서비스와 1:N 으로 연결이 되어 동작하는데 OSGi 의 특성을 반영하고 있기 때문에 새로운 디바이스가 생길 때마다 수동적으로 생성해야 한다. 따라서 서비스의 상태 정보를 포함한 동적인 발견은 지원하지 못한다.

## 2.2 Domotics Network [8]

DomoNet 는 이질적인 미들웨어의 서비스를 웹서비스로 변환하여 상호운용성을 보장한다. 서비스 지향 아키텍처를 기반으로 연구로 하나의 서비스를 웹서비스로 변환하는 기능을 담당하는 TM (TechManager)가 존재한다. TM 은 각 미들웨어에 존재하여 모든 서비스에 대해 변환하는 과정을 미들웨어에 종속적으로 동작하기 때문에 효율적이다. DomoNet 네트워크와 미들웨어의 연결을 담당하는 TM 은 미들웨어의 서비스를 DomoNet 환경에서 사용 가능하도록 하기 위해 미들웨어의 서비스 상태에 따라 웹서비스로 변환하는

동작을 한다. 하지만, 이러한 동작은 서비스의 수정이 일어나면 매번 웹서비스로 변환해야 하고 다양한 서비스 검색 방법을 제공하지 못한다.

본 논문에서는 미들웨어의 상호운용성을 지원하기 위한 관련 연구의 문제점을 토대로 DPWS 와 웹서비스의 이질적인 특성을 반영한 양방향 DPWS 어댑터를 제안한다.

## 3. 양방향 DPWS 어댑터 개요

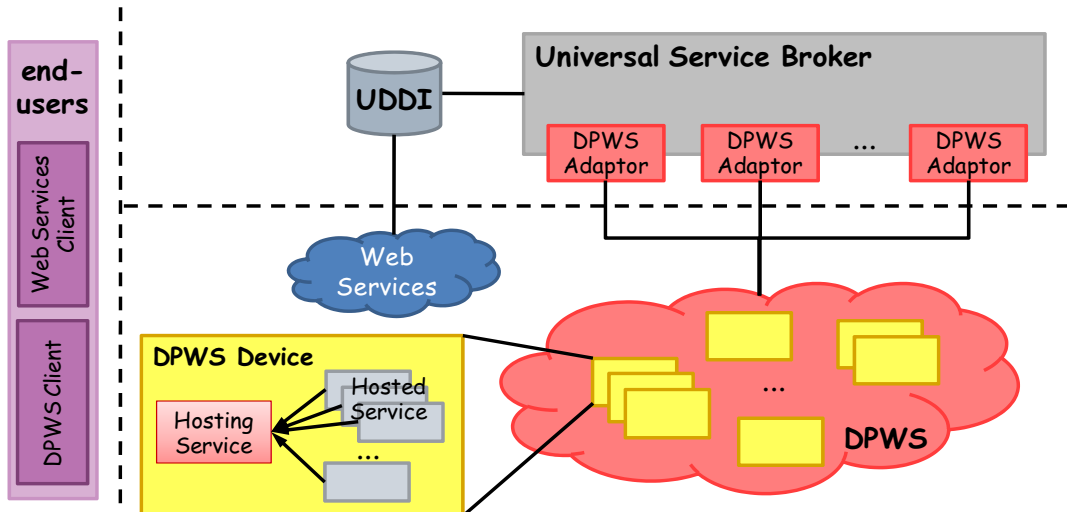
본 장에서는 양방향 DPWS 어댑터에 대해 소개한다. 양방향 DPWS 어댑터는 DPWS 와 웹서비스의 이질적인 특성을 반영하여 상호운용을 제공한다.

### 3.1 양방향 DPWS 어댑터 구조

양방향 DPWS 어댑터는 서비스 지향 아키텍처 패러다임 기반으로 접근하고 있기 때문에 일괄적인 표준 (e.g., SOAP-over-UDP)을 통해 상호운용을 제시한다. 그림 1 은 양방향 DPWS 어댑터를 포함한 상호운용을 지원하기 위한 환경으로 서비스 지향 아키텍처와 유사하게 3 가지로 분류된다. 서비스 사용자, 서비스 제공자, 서비스 중개자로 서비스 사용자는 논리적으로 DPWS 의 서비스 사용자와 웹서비스 사용자로 나뉜다. 하지만, 양방향 DPWS 어댑터를 통해 서비스 사용자는 DPWS 와 웹서비스에 상관없이 서비스를 제공받는다. 서비스 제공자는 웹서비스와 DPWS 의 서비스이며, 서비스 브로커는 서비스와 서비스 사용자를 연결하는 중개자 역할을 한다. 사용자에게 서비스의 형태에 관계없이 서비스를 제공하고 DPWS 의 서비스와 웹서비스가 서로를 연동하여 사용할 수 있는 상호운용을 지원한다.

### 3.2 양방향 DPWS 어댑터 구성 요소

양방향 DPWS 어댑터는 DPWS 와 웹서비스의 이질적인 특성을 반영한 상호운용성을 제공하기 위해 5 가지의 요소로 구성된다.



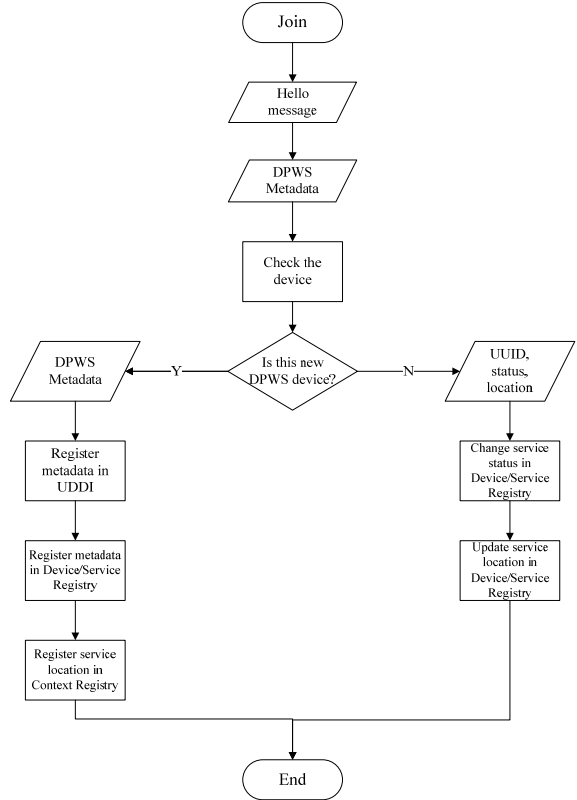
(그림 1) 양방향 DPWS 어댑터의 상호운용을 위한 환경

- DPWS 클라이언트 확장 - DPWS의 디바이스와 서비스 정보를 포함하는 메타 정보를 얻기 위해 양방향 DPWS 어댑터는 DPWS 클라이언트 모듈을 확장한다. DPWS 클라이언트는 DPWS 서비스의 정보를 얻기 위해 WS-MetadataExchange와 WS-Transfer를 이용한다. 이를 통해, DPWS의 디바이스 및 서비스의 메타 정보를 얻어 서비스의 효율적이고 동적인 검색을 지원한다.
- 발견 프록시 확장 - DPWS 클라이언트 모듈을 확장한 것과 달리, 웹서비스를 검색하기 위해 WS-Discovery의 DP (Discovery Proxy)를 확장한다. DPWS 클라이언트가 웹서비스를 사용하기 위해서는 Probe Message를 이용하는데 클라이언트 특성상 많은 메시지의 교환이 수반된다. 이는 DP를 통해 증대되는데 DPWS에서는 선택 사항이다. 본 논문에서는 교환되는 메시지의 양을 줄이고, 효율적인 검색을 위해 필수 요소로 존재한다.
- 가상 웹서비스 - DPWS는 SOAP 기반의 디바이스 레벨의 웹서비스 프로토콜을 지원하므로 DPWS 서비스를 가상의 웹서비스로 변환하지 않는다. 하지만, Jini나 HAVi와 같은 미들웨어의 서비스는 가상의 웹서비스로 변환하여 상호운용을 지원한다.
- 이벤트 및 알림 - 서비스의 동적인 검색을 위해 서비스의 상태 정보를 레지스트리로 관리한다. 따라서, 서비스의 상태 정보가 변경이 될 경우 이벤트와 알림을 통해 처리한다. 유비쿼터스 환경에서는 유동성에서 생기는 다양한 문제를 서비스의 상태 정보를 관리함으로써 동적인 서비스 검색을 제공한다.
- 메시지 전달 - DPWS와 웹서비스 및 서비스 사용자와의 상호운용을 위해 생기는 메시지 전달을 담당한다. DPWS는 SOAP-over-UDP이므로 메시지 변환 기능이 필요하지 않으나, 타 미들웨어에서는 필요하다.

### 3.3 양방향 DPWS 어댑터 기능

DPWS 어댑터의 구조와 구성 요소가 수행하는 기능에 대해 서술한다. 양방향 DPWS 어댑터는 6가지의 중요한 특징을 갖는다.

- 서비스 지향 아키텍처 - 서비스 지향 아키텍처는 이질적인 시스템의 통합 및 재사용 등을 위해 비즈니스 측면으로 연구되었다. 서비스 지향 아키텍처 패러다임의 기반인 DPWS는 느슨한 연결, 쉬운 통합, 서비스 조합, 비동기, QoS 등을 제공한다. 이러한 특징을 통해 양방향 DPWS 어댑터에 적용한다.



(그림 2) DPWS 디바이스 등장할 경우의 플로우차트

- 상호운용성 - 디바이스 레벨의 웹서비스인 DPWS가 웹서비스 프로토콜을 지원하지만, 유비쿼터스 환경에 적합하도록 유동성에서 생기는 문제점을 웹서비스와 호환되기 위해서 양방향 DPWS 어댑터를 제안하였다. 이를 통해, 부가적인 작업이 없이 이질적인 특성을 반영한 상호운용성을 제공한다.
- 일반적인 웹서비스 호출 (Generic Web Services) - 웹서비스를 호출하기 위해 다양한 타입과 파라미터로 구성된 형태에 적합하도록 변형해야 한다. 이를 해결하기 위해, 웹서비스 형태에 맞게 변환하여 호출하는 부분이다.
- 자가 구성 (Self-Organizing) - 양방향 DPWS 어댑터는 설계 단계와 실행 단계로 구분하여 자동적으로 구성되어 동작한다. 이에 대한 내용은 4장의 설계를 통해 자세히 설명한다.
- 동적 서비스 검색 - DPWS 서비스의 상태 정보를 레지스트리를 통해 관리하여 실제 사용 가능한 서비스만 검색하도록 제공한다.
- 메시지 수 감소 - DPWS에서 선택 사항인 DP를 필수 요소로 하여 DPWS 클라이언트의 수에 비례하여 증가하는 메시지에 대한 관리로 교환되는 메시지의 수를 감소시킨다.

#### 4. 양방향 DPWS 어댑터 설계

[9]은 웹서비스 클라이언트가 DPWS 서비스를 사용하는 단방향 DPWS 어댑터로 설계 단계와 실행 단계로 구분하여 설계를 다루고 있으므로 자세한 사항을 참고하고, 본 장에서는 웹서비스와 상호운용을 제공하기 위해 UDDI 를 포함한 양방향 DPWS 어댑터를 서술한다. 그림 2 는 양방향 DPWS 어댑터의 동작 중 대표적인 것으로 DPWS 디바이스가 네트워크에 등장할 경우, 단계별로 구분하여 동작하는 과정이다. 즉, DPWS 디바이스가 등장하면 기존에 네트워크에 있었던 디바이스인지 여부를 검사하여 UDDI 와 레지스트리에 등록하거나 수정한다.

#### 5. 양방향 DPWS 어댑터 구현

본 논문에서 다룬 DPWS 와 웹서비스의 상호운용을 지원하기 위한 양방향 DPWS 어댑터의 검증은 한다. 현재 DPWS 프로토콜을 따르는 디바이스가 존재하지 않기 때문에 PC 기반으로 가상의 DPWS 서비스를 만들어 간단한 회의 시나리오를 구현하였다. 구현에 사용된 시나리오는 다음과 같다.

회의 의장은 회의 장소에 들어서면 날씨를 제공하는 웹서비스를 통해 실내 온도가 조정된다. 회의에 필요한 문서 출력과 진행에 필요한 프로젝트 서비스를 이용한다. 프로젝터를 이용하여 회의가 시작되면 자동적으로 형광등의 조도가 조절된다.

- DPWS 서비스 – Heater, Printer, Projector, light
- 웹서비스 – Weather Information

구현에 필요한 환경은 다음과 같다.

- 운영체제
  - DPWS 클라이언트 – Windows Vista Ultimate K
  - DPWS 서비스 – Windows Server 2008
- 소프트웨어
  - WSD-API SDK
  - .NET Micro Framework 3.0
  - Visual Studio 2008 Team Suite
  - DPWS Explorer
- 구현언어 – C#

현재 DPWS 프로토콜을 지원하는 디바이스가 존재하지 않기 때문에 마이크로소프트의 WSD-API SDK [10]를 이용하여 DPWS 서비스를 만들었다. 또한, 양방향 DPWS 어댑터는 DPWS 클라이언트와 WS-Discovery 의 DP 를 확장하기 위해 각각 .NET Micro Framework [11]와 WS-Discovery Implementation [12]를 이용한다. 또한, 날씨 제공 웹서비스는 Axis 1.2 를 통해 구현하였다. 마지막으로 구현한 DPWS 서비스와 양방향 DPWS 어댑터의 메타 정보와 메시지를 확인하기 위해 DPWS Explorer [13]를 통해 검증하였다.

#### 6. 결 론

본 논문에서는 DPWS 와 웹서비스의 실질적인 특성에서 생기는 문제점을 해결하고 상호운용을 지원하기 위해 양방향 DPWS 어댑터를 제안했다. 이는 서비스 지향 아키텍처를 기반으로 설계 단계와 실행 단계에서 자동적으로 구성이 된다. 또한, 다양한 형태의 웹서비스 호출을 위한 복잡한 작업을 단순화하였으며, 서비스의 상태 정보를 포함한 동적 서비스 검색과 메시지 수의 감소의 이점을 제공한다.

양방향 DPWS 어댑터는 설계 단계의 네트워크에 구성이 될 경우와 실행 단계의 디바이스의 등장, 퇴장, 이벤트, 알림 등의 사항을 모두 처리하도록 설계되었다. 또한, 설계에 대한 검증을 위해 간단한 시나리오의 구현을 통해 실제 환경에서 제대로 운영되는지 확인하였다.

추후 양방향 DPWS 어댑터의 최적화와 서비스 사용자에게 다양한 서비스 검색 및 연동을 위한 방안을 제시할 필요가 있다. 이와 함께 WSUN 에서 다루고 있는 Jini 와 상호운용의 지원을 통해 유비쿼터스 환경에서의 실제적인 운영을 제공할 것이며, 상황 정보를 고려한 서비스 조합뿐만 아니라 트랜잭션, 보안, QoS 등의 연구로 유비쿼터스 웹서비스의 최적의 프레임워크를 제공할 것으로 기대된다.

#### 참고문헌

- [1] Sun Microsystems. JINI Specifications Archive v2.1, 2005
- [2] UPnP Forum. UPnP Device Architecture v1.0.1, 2 December 2003
- [3] HAVi Consortium. HAVi Specification v1.1, 15 May 2001
- [4] Chan, S. et. al: Devices Profile for Web Services, February 2006, Microsoft Developers Network Library
- [5] 임형준, 오일진, 황윤영, 이경하, 이강찬, 이승윤, 이규철, “유니버설 네트워크를 위한 웹서비스 프레임워크 설계 및 구현”, 한국정보과학회, 정보과학회논문지: 컴퓨터의 실제 및 레터 제 14 권 제 2 호, pp. 143-157, 2008. 4
- [6] Michael Ditze, Guido Kamper, Isabel Jahnich, Reinhard Bernhardt-Grison, "Service-based access to distributed embedded devices through the open service gateway", INDIN 2004, pp. 493-498, June 2004
- [7] OSGi Alliance. About the OSGi Service Platform, Technical Whitepaper Revision 4.1, 7 June 2007
- [8] Vittorio Miori, Luca Tarrini, Maurizio Manca, "An Open Standard Solution for Domestic Interoperability", IEEE Transactions on Consumer Electronics, vol. 52, NUMB 1, pp. 97-103, February 2006
- [9] 임형준, 오일진, 황윤영, 이강찬, 이승윤, 이규철, “유비쿼터스 웹서비스 환경의 DPWS 와 웹서비스의 상호운용성을 위한 DPWS 어댑터 설계”, 한국정보과학회, 추계학술대회 학술발표논문집(C) 제 34 권 제 2 호, pp. 90-95, 2007. 11
- [10] Microsoft. *Web Services on Devices API (WSDAPI)*. Microsoft Windows SDK for Windows Server 2008 and .NET Framework 3.5. 2, May, 2008.
- [11] Microsoft Connect. .NET Micro Framework 3.0.
- [12] Microsoft .NET Framework. WCF WS-Discovery Protocol Implementation. 21, December, 2006.
- [13] Web Services for Devices (WS4D), DPWS Explorer 2 Prototype, 26, February, 2008. <http://ws4d.org/>