

물리 효과 기반 서비스 디스커버리

신용진*, 김도현*, 이영희*, 최우진**
*한국정보통신대학교 공학부
**㈜케이티 인프라 연구소
e-mail : rustystar@icu.ac.kr

Physical Effect Based Service Discovery

Yong-Jin Shin*, Do-Hyun Kim*, Young-Hee Lee*, Woo-Jin Choi**
*Dept. of Engineering, Information and Communications University
**Infra Laboratory, Korea Telecom

요 약

본 논문은 대체 서비스 디스커버리를 위한 효과 기반 명세 모델과 매칭 방식을 제안한다. 효과 기반이라 함은 시맨틱 웹의 IOPE 모델에서 효과(effect)를 재 정의 하여 서비스 및 디바이스를 명세 하는데 사용한다는 말이다. 여기서 효과는 인간이 인지할 수 있는 오감을 사용하며, 오감의 인지는 센서를 통하여 한다. 오감을 이용하여 서비스를 탐색 함으로써, 우리는 서비스의 가용성을 높일 수 있다. 매칭은 서비스를 세 가지의 타입으로 나눈 후 각 타입에 해당하는 항목별로 유사도를 계산하고 각 항목들의 합으로 최종 유사도를 도출해 낸다. 본 연구에서 제안한 효과 기반 서비스 디스커버리는 기존의 시맨틱 서비스 디스커버리보다 서비스의 가용성을 높여 줄 것으로 기대한다.

1. 서론

서비스 투명성(Service Transparency)의 제공은 유비쿼터스 컴퓨팅 환경에서 중요한 이슈 중 하나이다. 이것은 유저의 환경이 바뀌어도 기존에 사용하던 서비스를 유저의 직접적 관여 없이, 마치 동일한 환경에서 쓰는 것처럼 제공하는 유비쿼터스 환경의 중요한 요구사항이며 이를 제공하기 위한 중요한 요소가 서비스 디스커버리(Service Discovery) 기술이다.

이 기술은 유비쿼터스 미들웨어의 요소 기술이며, 유저 혹은 유저의 응용이 원하는 서비스를 즉시적으로 찾아주는 역할을 한다[1]. 찾아준다는 것은 일반적으로 서비스에 대한 참조정보를 전달해 주는 것이며, 이를 실행한 응용은 참조정보를 활용해 서비스에 접근하거나 서비스를 호출한다. 서비스 디스커버리 프로토콜들은 대상으로 하는 응용 및 도메인에 따라 접근하는 서비스 명세 방식, 매칭 방식, 기반 전송 프로토콜 등이 다양하지만 목적은 동일하다고 할 수 있다.

현재 널리 적용되고 있는 UPnP[2], Jini[3], Bonjour[4], Bluetooth SDP[5]와 같은 서비스 디스커버리 프로토콜들은 대부분 간단한 서비스의 타입(type)이나 아이디(ID) 기반의 검색만을 지원한다. 이것은 컴퓨팅 측면에서 상당히 효율적이지만 서비스를 미리 약속된 단순한 타입이나 아이디만으로 표현함으로써 서비스 및 질의(query) 명세(description)의 표현력을 저하시키고 의미 기반(semantic-based)의 디스커버리를 어렵게 한다. 의미 기반 서비스 디스커버리 기술들[6,7,8,9,10]은 서비스의 의미(semantic)를 보다 풍부하게 표현하기 위해 서비스를 다양한 하위 개념(concept)과 그 개념들간의 관계(relation)를 가진 그래프로 표현하고 검색

시에 서비스를 구성하는 하위 개념과 그들간의 관계로부터 의미를 유추하여 보다 의미적인 서비스 검색을 하는데 초점을 맞추고 있다[7,11,12,13]. 대부분의 의미기반 서비스 디스커버리 기술들은 서비스 모델을 표현하는 서비스 온톨로지를 활용하고 있다.

서비스 디스커버리는 유비쿼터스 환경에서 유저 및 응용이 필요로 하는 서비스가 가용하지 않을 때 대체하여 제공할 수 있는 서비스를 찾아주는 것을 말한다. 현재 의미기반 서비스 디스커버리 프로토콜은 서비스의 입력(Input), 출력(Output), 선행조건(Precondition), 효과(Effect) (IOPE) 같은 기능적인 측면(functional aspect)을 나타내는 개념에 의미정보를 부가(annotation)적으로 제공하여 보다 풍부한 의미정보를 제공하는 방식이며, 이것은 다양한 의미표현을 기반으로 한 서비스 가용성 증대의 여지를 남겨두고 있다.

본 연구에서는 보다 높은 서비스 가용성을 보장하기 위한 효과 기반 서비스 디스커버리 기술과 매칭 방식을 제안한다. 제안된 서비스 디스커버리 기술의 핵심은 검색을 위해 서비스의 IOPE 개념을 활용함과 동시에 효과라는 개념의 기존 의미와 본 연구에서 재해석된 새로운 개념으로 확장하여 검색에 활용한다는 것이다. 여기서 제안하는 효과 개념은 서비스의 수행 시 공간 혹은 인간에게 미치는 영향을 의미하며 인간의 오감을 기준으로 구분된다. 본 연구에서는 기능적 측면의 의미정보 표현에 초점을 맞춘 서비스 모델에 효과 개념을 추가적으로 보완 활용함으로써 보다 높은 서비스 가용성을 보장하는 효과 기반 서비스 디스커버리 프로토콜을 제안한다. 본 연구에서 제안된 매칭 방법의 핵심은 서비스를 세가지 타입으로 나눈 후

각 타입별로 유사도를 계산을 한다는 것이다. 여기서 타입은 서비스들 간의 관계에 따라 구분 되며, 이는 차 후 근사 매칭을 하는데 사용되고, 근사 매칭을 함으로써 각각의 서비스가 유저의 질의를 얼마나 만족하는지 알려 준다.

서비스 검색 프로토콜 제안을 위해 본 논문에서는 동기 및 디자인 시 고려사항을 알아보고 (2 장) 효과 기반 서비스 디스커버리를 위한 서비스 명세 및 질의 명세 방법을 포괄하는 서비스 표현 모델과 서비스 매칭 방법(3 장)을 제시한 후, 결론을 내리고 향후 계획을 논의(4 장) 하며 논문을 끝마친다.

2. 동기 및 디자인 고려 사항

서비스 가용성(service availability), 인간 친화적 사용 방법(Human-familiar usage convention): 웹 서비스 vs. 유비쿼터스 서비스 대부분의 웹 서비스가 물리적인 위치에 제한을 받지 않고 응용간의 상호작용(interoperation)에 초점을 맞추어 설계되는 반면, 유비쿼터스 환경에서의 서비스는 물리적으로 유저의 근처(proximity)로 제한되는 경우가 대다수이며 응용간의 상호작용뿐만 아니라 인간과의 상호작용 또한 중요한 비중을 차지하고 있다는 점에서 차이가 있다 [1]. 유비쿼터스 환경에서의 서비스 디스커버리 기술은 위치 정보와 인간이라는 요소를 중요한 고려 사항으로 포함해야 한다.

웹 서비스는 위치 정보가 중요하지 않은 소프트웨어 기반 서비스가 대부분이며 여러 사용자가 동시에 이용하는 것이 가능하다. 웹 서비스 환경에서는 서비스의 가용성이 큰 문제가 되지 않는다. 오히려 유저의 질의에 적절한 서비스가 너무 많은 것이 문제가 될 수 있다. 이에 반해 사무실과 같은 로컬 환경에 종속된 서비스가 대다수인 유비쿼터스 환경에서는 서비스를 제공할 수 있는 서비스 자원 자체가 물리적인 위치에 종속되어 있으므로 서비스의 가용성이 중요한 이슈가 될 수 있다. 더구나 다수의 유저가 함께 생활하는 환경에서 공유되기 힘든 서비스를 가정한다면 가용성의 증대는 더욱더 중요한 요소가 된다.

동기: 서비스 가용성, 인간 친화적 사용 방법 제안하는 효과 개념은 서비스의 실행이 환경에 미치는 영향(impact)을 표현함에 있어 인간의 인지감각(perceptual sense)인 오감(five sense) - 시각(sight), 청각(hearing), 촉각(touch), 미각(taste), 후각(smell) - 을 기본으로, 서비스의 실행 효과를 인간이 인지하는 관점에서 기술한다는 데 의의가 있다. 여기서 대상으로 하는 서비스는 인간을 위한 것이며, 따라서 인간이 인지할 수 있는 서비스의 실행 효과는 서비스를 찾는 데 상당히 유용할 수 있다.

효과 기반의 서비스 디스커버리 프로토콜의 설계를 위해 고려해야 할 이슈 및 요구사항은 다음과 같다.

서비스 가용성 vs. 적절성(appropriateness) 약속된 서비스 카테고리나 타입기반의 서비스 디스커버리 기술들이 가지는 장점은 검색된 서비스의 정확성 및 적절성이 높다는 것이다. 서비스를 좀 더 추상적/일반적으로 기술함으로써 얻을 수 있는 서비스의 높은 가용

성은 낮은 적절성을 감수한 자연스러운 결과라 할 수 있다. 본 연구에서 도입하는 오감 기반의 효과 개념은 모든 서비스를 오감을 기준으로 명세함으로써 부정확한 서비스를 찾아 줄 가능성이 높다. 따라서 제안하는 효과 개념은 기존의 IOPE 모델과 함께 상황에 맞게 활용되거나 서로를 보완하는 방식으로 설계되어야 한다.

유저의 역할. 효과 기반의 서비스 디스커버리를 위해 사용자의 서비스에 대한 요구사항과 서비스 명세가 효과에 대한 명세 항목을 가지고 있어야 한다. 이는 서비스의 명세나 질의가 더 많은 정보를 필요로 함을 의미하며 유저나 서비스 제공자는 해당 개념이 의미하는 바를 잘 이해하고 있어야 하며 이는 곧 유저의 부담을 의미한다. 서비스 질의 및 명세부분의 효과 개념을 어떻게 유저의 부담을 최소화하여 잘 채워줄 것인가에 대한 고려가 필요하다. 본 연구에서는 사용자가 직접 값을 채우는 방식과 사용자의 직접적인 관여 없이 명세부분을 채우는 방식을 모두 고려한다.

3. 물리 효과 기반 서비스 디스커버리

3.1 환경과 효과 모델

환경과 효과 모델은 다음과 같은 식에 따라 서비스 효과로 나타나게 된다.

- (1) $Effect(service_i) = E$ execution of service_i
- (2) $E' = \{e_i \mid i \text{ is the index of element that consist of } E'\}$
- (3) $E = \{e_i \mid i \text{ is the index of element that consist of } E\}$

위 식에서 E 는 환경을 나타내며 이는 하나 이상의 환경(온도, 습도, 소음, 조도 등) e_i 로 이루어진다. 각각의 환경은 다양한 센서를 통하여 관찰 된다. e_i 는 상태 값인 s_i 를 갖고 있다.

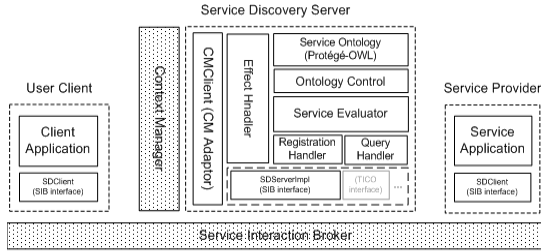
$$Effect(service_i) = \{s_i - s' \mid i \text{ is the index of element that consist of } E'\}$$

E 는 서비스를 실행하기 전의 환경이고 E'는 서비스를 실행한 후의 환경이다. 위 식들을 통하여 우리는 서비스의 물리적 효과를 알아낼 수 있다.

3.2 구조

본 연구에서는 인간이 인지 가능한 물리적 효과 기반 서비스 디스커버리를 위하여 그림 1 과 같은 구조를 가지는 미들웨어를 제안한다. 그림 1 은 클라이언트, 서버, 그리고 서비스성분으로 구성된다. 각각의 성분은 Service Interaction Broker 를 통하여 통신하며, 다양한 binding 을 제공해 준다. Registration 과 Query Handler 는 각각, 클라이언트로부터 요청된 쿼리를 처리하며 서비스를 광고 하기도 한다. 본 연구의 서비스 명세 모델은 OWL-Lite 와 protégé 를 사용하여 만들었다. Service Ontology 가 owl 파일을 저장하고 API 를 제공하며, Ontology Control 은 서비스 저장소에서 등록, 업데이트, 등록, 삭제 등을 한다. Service Evaluator 는 유저 쿼리와 서비스 저장소에 있는 서비스 명세가 얼마나 유사한지 계산한다. 본 연구의 서비스 디스커버리 미들웨어는 Context Manager(CM)를 통하여 상황인지 서비스 디스커버리를 지원한다. 현

재 CM 은 온도, 조도, 습도, 위치 등의 하위 레벨의 환경부터 수면, TV 시청 같은 상위 레벨의 상황도 제공한다. 본 논문에서는 위에서 설명한 모든 성분들을 다 다루지 않고 효과와 관련된 서비스 명세 모델과 Service Discovery Server 에 있는 Effect Handler 모듈만 다루기로 한다.

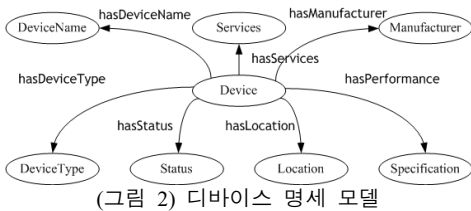


(그림 1) 제한하는 물리 효과 기반 서비스 디스커버리 구조

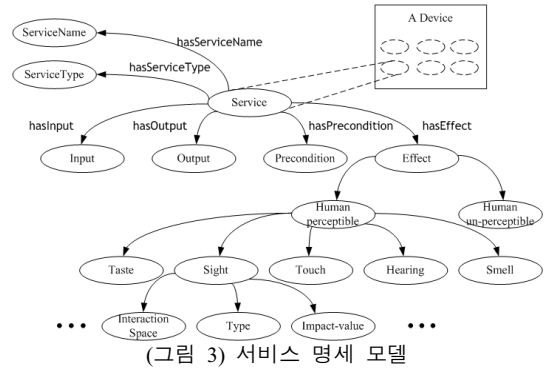
3.3 서비스 명세 모델

본 장에서는 디바이스와 서비스 명세 모델을 설명한다. 하나의 서비스는 하나 이상의 서비스를 지원해 준다고 가정하고 있으며, 디바이스 명세 모델은 그림 2 처럼 DeviceType, DeviceName, Manufacturer, Status, Specification, Service, 그리고 Location 으로 나눈다. 각각에 대한 예는 표 1 에서 찾아 볼 수 있다. 우리는 DeviceType 에 대해 분류 관계를 만들었고, 각각의 DeviceType 요소는 DeviceName, Manufacturer, Status, Specification, Service 등을 갖는 고유한 온톨로지를 갖는다.

그림 3 은 서비스 명세 모델을 나타내고 있다. 본 연구에서는 효과를 두 개의 카테고리(인간이 인지 가능, 인간이 인지 불가능)로 나누었다. 본 연구에서는 인간이 인지 가능한 효과, 물리적 효과, 만을 다루기로 한다. 각 감각은 하위 개념을 갖고 있는데, 이는 Interaction-Space, Type, 그리고 Impact-value 이다. Interaction Space 는 물리적, 지리적 범위 혹은 해당하는 효과의 범위를 말한다. Type 은 어떤 종류의 시각, 예를 들어 밝음, 등을 말한다. Impact-value 는 서비스가 얼마나 환경에 영향을 미치는 가를 의미하며, 이는 환경의 변화 정도를 뜻한다. 시각의 Type 이 brightness 인 경우, 100lx 증가 가 Impact-value 의 예가 될 수 있다.



(그림 2) 디바이스 명세 모델



(그림 3) 서비스 명세 모델

표 1 은 디바이스 비디오 카메라에 대한 명세 예시, 표 2 는 인간이 인지 가능한 오감을 이용한 TV 의 TurnOn 서비스 명세 예시 이다.

<표 1> 비디오 카메라 디바이스 명세 예시

Concept	Instance
DeviceType	'Video Camera'
DeviceName	'CoolFix'
Manufacturer	'Samsung Electronics'
Status	'TurnOn'
Specification	'CCD', '3648x2734', '30'
Service	'TurnOn', 'Monitor', 'Capture', 'TurnOff'
Location	'F617'

<표 2> 효과를 이용한 TV 의 서비스 명세 예시

Concept	Instance	
ServiceName	'TurnOn'	
ServiceType	'Video/Image Displaying'	
Input	'Channel'	
Output	N/A	
Precondition	'TurnOff'	
Effect	Sight	InteractionSpace('Bedroom partition 2'), Type('Brightness'), ImpactValue('Inc:100lx')
	Hearing	InteractionSpace('Bedroom'), Type('Sound'), ImpactValue('Inc:20dB')

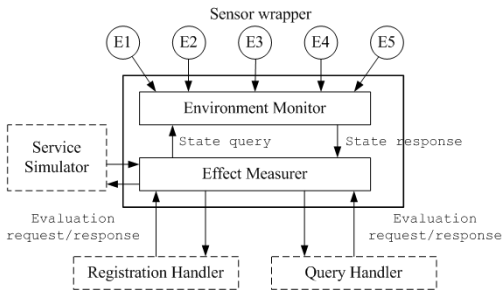
3.4 서비스 명세에서의 효과 표현

본 장에서는 서비스 제공자가 서비스 명세의 효과를 어떻게 채워 넣는지를 설명한다. 예를 들어 집안에 "Lightning" 서비스 제공자가 있다고 가정하자. 그는 서비스가 집의 밝기를 변화 시킬 것이라고 예상된다. 하지만 어느 정도나 밝게 또 얼마나 멀리 영향을 미칠지는 알 수 없다. 앞에서 언급 하였듯이 효과는 Interaction Space, Impact-value 그리고 Type 정보를 필요로 한다. 본 연구에서는 다양한 센서를 이용하여 서비스의 효과를 측정하고 측정된 값을 이용하여 서비스 명세를 채워 넣는다. 이러한 방식은 두 가지의 장점을 가지는데, 하나는, 인간의 측정보다 더 객관적으로 측정할 수 있으며, 또 다른 하나는, 인간이 서비스의 효과를 추측하는 부담을 없앨 수 있다.

그림 4 는 Effect Handler 의 작동 메커니즘을 설명한

다. Environment monitor 는 센서로부터 들어오는 정보를 저장한다. 각각의 Effect Measurer 는 서비스의 실행으로 인한 환경의 변화를 계산하여 환경의 상태를 평가한다. 새로운 서비스가 광고를 하면, Registration handler 가 Effect Measurer 에게 효과 측정을 요청 한다.

그 후, Effect Measurer 는 Service Simulator 에게 서비스의 실행을 요청하고, Environment Monitor 에게 현재의 상태를 체크 하라고 요청한다. Service Simulator 가 서비스의 실행이 종료 되었다고 Effect Measurer 에게 신호를 보내면, Effect Measurer 는 환경의 변화를 체크한다. 그에 따라 상응하는 효과가 서비스 명세에 채워지고 이를 Registration Handler 에게 준다. 그 후 Registration Handler 는 서비스 명세를 서비스 저장소에 등록한다. 쿼리 처리 과정은 서비스 광고 처리와 조금 다르다. 먼저 Query Handler 가 쿼리로부터 Service Type 값을 얻어오고, Service Type 을 기반으로 Effect Measure 가 효과를 평가 한다.



(그림 4) Effect Handler

3.5 서비스 평가

서비스 평가 방법은 모든 개념들의 가중치가 반영된 유사도의 합이다.

$$\text{Similarity}(R, A) \equiv \sum_{i=1}^n W_i \times \text{SimilarityScore}_i(R_i, A_i)$$

R 은 개별 요구사항인 Ri 의 집합이고, A 는 서비스 각 명세들인 Ai 의 집합이다. (Ri, Ai) 쌍은 하나의 개념 Ci 를 나타낸다. 각각의 유사도를 계산하는 함수 SimilarityScorei 는 개념에 따라 다르다. 각각의 개념을 나누어 유사도를 계산함으로써, 더욱 정확한 유사도를 얻을 수 있다. 왜냐하면, 각각의 개념은 서로 다른 성질을 가졌기 때문이다. 가중치 요소 Wi 는 각 개념들이 얼마나 중요한지를 나타낸다.[10]에서는 개념을 세가지 - 분류관계를 가진 개념, 분류관계를 가지지 않은 개념, 자료형식의 제한 - 로 나누어 유사도를 계산 하는데, ServiceType 과 Type 이 분류관계를 가진 개념, Input 과 Output 이 분류관계를 가지지 않은 개념, Impact-value 는 자료형식의 제한 에 해당되며, Location 과 Interaction Space 의 유사도 측정을 위해서는 위치 모델이 필요 하다.

4. 결론

서비스 자원 자체가 물리적인 위치에 종속되어 있

고, 다수의 사용자가 함께 생활하는 환경에서 서비스가 공유되기 힘든 상황이 많아 집에 따라, 서비스의 가용성을 증대 시키는 것은 중요한 요소가 되고 있으며, 이에 본 연구에서 우리는 효과기반 명세모델을 제안 하였다. 효과 기반 명세모델은 인간의 인지각각인 오감을 효과개념에 이용하여 서비스의 실행 효과를 인간이 인지하는 관점에서 기술한다는 데 의의가 있으며, 이는 서비스 가용성을 높일 수 있다고 평가된다.

참고 문헌

- [1] Feng Zhu, Matt W. Mutka, and Lionel M. Ni, "Service Discovery in Pervasive Computing Environments", Pervasive Computing, IEEE, Volume 4, Issue 4, Oct.-Dec. 2005 pp.81 - 90
- [2] Universal Plug and Play Forum. "Universal Plug and Play Device Architecture Version 0.91", <http://www.upnp.org>, March 2000.
- [3] Sun. Technical White Paper, "Jini Architecture Overview", <http://www.sun.com/jini/>, 1999.
- [4] Apple. "Bonjour", <http://developer.apple.com/networking/bonjour/index.html>
- [5] Bluetooth SIG, Specification of the Bluetooth System – Core, Version 1.0B volume 1, 1999. Part E.
- [6] John Miller, Kunal Verma, Preeda Rajasekaran, Amit Sheth, Rohit Aggarwal, Kaarthik Sivashanmugam, "WSDL-S: Adding Semantics to WDSL", white paper, university of Georgia, 2004
- [7] Dipanjan Chakraborty, Filip Perich, Sasikanth Avancha, and A.Joshi, "Dreggie: Semantic service discovery for m-commerce applications." In workshop on Reliable and Secure Applications in Mobile Environment, Symposium on Reliable Distributed Systems, 2001
- [8] Jeffery Hau, William Lee and John Darlington, "A Semantic Similarity Measure for Semantic Web Services", in Web Service Semantics Workshop 2005 at WWW2005. 2005. Chiba, Japan
- [9] Jaeger M C, Rojec-Goldmann G, Liebethuth C et al. "Ranked matching for service descriptions using OWL-S", InProc. Communication in Distributed System (KiVS05), Kaiserslautern, Geany, 2005, pp.91- 102
- [10] Bandara, A., Payne, T., de Roure, D., Gibbins, N., Lewis, T. "Semantic resource matching for pervasive environments: The approach and its evaluation", Technical report, School of Electronics & Computer Science, University of Southampton, 2008
- [11] Sousa, J. P., and Garlan, D. "Aura: an Architectural Framework for User Mobility in Ubiquitous Computing Environments", Proceedings of the 3rd Working IEEE/IFIP Conference on Software Architecture August 25-31, 2002. pp. 29-43.
- [12] Tomasz Rybicky, "Semantic Service Discovery in Pervasive Computing", International Conference on Pervasive Service 2008 (ICPS'08), July 6-10, 2008, pp.69 – 72
- [13] A. Toninelli, A. Corradi, R. Montanari, "Semantic-based Discovery to Support Mobile Context-aware Service Access", Computer Communications archive Volume 31 , Issue 5., pp. 935-949