

다중 속성 의사결정에 의한 웹 서비스 선정 프로세스에 관한 연구

서 영 준[†] · 송 영 재^{††}

요 약

최근 웹 서비스 분야는 SOA(Services-Oriented Architecture)에 대한 관심의 증가와 B2B 시장의 성장으로 인해 차세대 IT 패러다임으로 급부상하고 있다. UDDI(Universal Description, Discovery and Integration)를 통한 서비스 발견은 오직 기능적 측면에만 기반을 두기 때문에, 서비스의 사용 빈도와 상호간의 신뢰성에 미치는 영향은 고려하지 않았다. 즉, 웹 서비스의 비기능적 측면인 품질은 소비자와 제공자 상호간에 성공을 위한 중요한 요인이 될 것이며, 이에 품질을 고려한 웹 서비스 선정 방법이 요구된다.

본 논문에서는 서비스 소비자의 입장에서 소비자가 원하는 최적의 품질을 제공하는 서비스를 찾도록 도와주는 에이전트 기반 품질 브로커 아키텍처와 선정 프로세스를 제안한다. 웹 서비스와 같이 분산되고 이질적인 환경에서 에이전트에 관한 이론들은 널리 받아들여지고 있으므로 제안하는 시스템 아키텍처에 적합하다.

본 논문에서는 웹 서비스 선정에 관한 기존 연구의 문제점을 개선하기 위해 평가과정에서 QoS와 CoS를 고려하였으며, 평가 방법으로 다기준 의사결정 기법들 중에서 웹 서비스 선정에 가장 적합하다고 판단된 PROMETHEE(Preference Ranking Organization MeTHod for Enrichment Evaluations)를 사용하였다. PROMETHEE는 비교 서비스들이 추가되거나 삭제되더라도 이원비교를 다시 수행해야 하는 문제를 극복할 수 있는 장점이 있다. 본 논문에서는 제시한 선정 프로세스를 검증하기 위하여 서비스 조합 시나리오를 갖는 사례 연구를 제시하였다. 사례 연구에서 웹 서비스 선정 프로세스는 소비자 관점의 품질 측정값과 정의된 서비스 레벨을 바탕으로 의사 결정 문제를 기술하였다.

키워드 : 웹 서비스, 웹 서비스 조합, 품질 측정, 품질 브로커, 다기준 의사결정기법, PROMETHEE

A Study on Selection Process of Web Services Based on the Multi-Attributes Decision Making

Young-Jun Seo[†] · Young-Jae Song^{††}

ABSTRACT

Recently the web service area is rapidly growing as the next generation IT paradigm because of increase of concern about SOA(Services-Oriented Architecture) and growth of B2B market. Since a service discovery through UDDI(Universal Description, Discovery and Integration) is limited to a functional requirement, it is not considered an effect on frequency of service using and reliability of mutual relation. That is, a quality as nonfunctional aspect of web service is regarded as important factor for a success between consumer and provider. Therefore, the web service selection method with considering the quality is necessary .

This paper suggests the agent-based quality broker architecture and selection process which helps to find a service providing the optimum quality that the consumer needs in a position of service consumer. A theory of agent is accepted widely and suitable for proposed system architecture in the circumstance of distributed and heterogeneous environment like web service.

In this paper, we considered the QoS and CoS in the evaluation process to solve the problem of existing researches related to the web service selection and used PROMETHEE(Preference Ranking Organization MeTHod for Enrichment Evaluations) as an evaluation method which is most suitable for the web service selection among MCDM approaches. PROMETHEE has advantages that solve the problem that a pair-wise comparison should be performed again when comparative services are added or deleted. This paper suggested a case study with the service composition scenario in order to verify the selection process. In this case study, the decision making problem was described on the basis of evaluated values for qualities from a consumer's point of view and the defined service level.

Key Words : Web Service, Web Service Composition, Quality Measurement, Quality Broker, MultiCriteria Decision Making, Preference Ranking Organization MeTHod for Enrichment Evaluations

1. 서 론

IT의 새로운 패러다임이라 할 수 있는 웹 서비스는 Standard-

Based, Services-Oriented Architecture에 대한 관심의 증가와 B2B 시장의 성장으로 인해 기업 환경에서 그 비중이 점점 증대하고 있다. 웹 서비스 시장은 향후 5년간 연평균 58%로 증가하여 2008년에는 110억 달러에 이를 것으로 예상되며, 특히 서비스 품질 평가를 지원하는 Hosted Service 영역은 2006년까지 전체 비즈니스 영역 중 18%를 차지하면

※ 이 논문은 2004년도 경희대학교 지원에 의한 연구 결과임.

† 준 회 원 : 경희대학교 컴퓨터공학과 박사과정

†† 종신회원 : 경희대학교 전자정보대학 교수

논문접수 : 2006년 2월 8일, 심사완료 : 2006년 6월 12일

서 꾸준히 성장할 것으로 전망되었다[1, 2].

그러나 웹 서비스가 활성화되기 시작하면서 해결해야 할 문제점들 또한 증가하고 있다. 이러한 문제들 중에서 초점이 되고 있는 요인은 웹 서비스 선정 문제이다. 현재의 웹 서비스 모델에서 UDDI(Universal Description, Discovery and Integration) 레지스트리는 웹 서비스에 대한 평가가 아닌 웹 서비스에 대한 설명만을 담고 있으며, UDDI 레지스트리의 48%가 잃어버리거나, 깨지거나, 정확하지 않은 정보를 포함하는 연결을 가지는 단점이 있다[3]. 많은 유사한 웹 서비스들 가운데 하나를 선택해야 할 때, 서비스 소비자는 일반적으로 웹 서비스의 품질 정보를 필요로 하게 된다. 비록 UDDI가 서비스 품질 정보를 제공하도록 설계된 것은 아니지만, UDDI 레지스트리들은 서비스 소비자들에게 편의를 제공하기 위해서 이러한 정보들을 포함하려 하는 추세이다. 또한, 애플리케이션이 다양한 웹 서비스들로 조립되는 시나리오에서는 소비자들이 웹상에서 복잡한 질의들을 처리해야 하는 어려움이 있다. 시간 낭비와 어려운 프로세스는 무시하더라도, 소비자는 더 나은 서비스 선택의 기회를 잃기 쉽다[4].

따라서 웹 서비스 선정 문제를 해결하기 위해서는 서비스 소비자들 간에 품질 속성들을 공유할 수 있는 수단이 필요하다. 공유된 품질 속성들을 사용하여, 서비스 소비자들은 필요한 서비스 구현들의 품질 평판을 알 수 있다. 따라서 소비자의 품질 요구에 가장 적합한 서비스 구현 선정은 가장 신뢰 받는 구현을 선택하는 것과 일치한다.

본 논문에서는 서비스 소비자들을 대신하여 최적의 품질을 제공하는 웹 서비스를 선정하는 에이전트 기반 품질 브로커 아키텍처와 선정 프로세스를 제안한다. 품질 브로커 아키텍처는 서비스 소비자들을 대신하여 자율적으로 행동하는 소프트웨어 에이전트와 함께 서비스 선정 작업을 자동화한다. 또한, 서비스 선정 알고리즘과 서비스 성능에 대한 품질 데이터를 동적으로 확보하기 위한 과정을 소개한다. 따라서 소비자의 애플리케이션은 각자의 품질 요구에 적합한 최적의 서비스를 동적으로 수행 시간에 선정할 수 있다.

기존의 웹 서비스 선정 연구들 중에는 구체적인 서비스 선정 방법을 명시하지 않거나 소비자에 의한 비자율적인 선정이 이루어져 선정 과정이 어려워지는 문제점이 있었다. 또한, 자율적인 선정 과정이 이루어지더라도 선정 과정에서 다자간, 다중 속성을 고려하지 않거나, 현실과는 다르게 제공자들이 서비스마다 QoS(Quality of Service) 정보를 지원 하는 상황을 가정하는 문제도 있었다. 따라서 본 논문에서는 다자간, 다중 속성을 고려한 자율적인 서비스 선정 프로세스를 제안하며, 제공자들이 제공하지 못하는 서비스의 QoS 정보를 소비자들의 피드백으로 인한 QoS 정보 공유로 해결하고자 한다.

본 논문에서는 최적의 서비스를 선정하기 위한 구체적인 알고리즘으로 다기준 의사 결정 기법중 하나인 PROMETHEE를 이용하여 기술하였다. 웹 서비스 선정은 동시에 측정되고 평가되는 많은 품질 속성들이 요구되고, 한 품질 속성

간에 상충 관계(trade-off relationship)를 유발 할 수 있다[5]. 이러한 서비스 선정 환경에서 PROMETHEE는 MAUT, AHP와 같은 다른 기법과 비교하여 다음과 같은 장점들을 가지고 있다. PROMETHEE는 이원비교가 자동으로 수행되므로, 비교 대안(comparative alternative)이 추가되거나 삭제되더라도 AHP에서처럼 이원비교를 다시 수행해야 하는 문제를 극복할 수 있다. 따라서 유사 서비스의 수가 많고, 새로운 서비스가 추가되거나 삭제되는 경우 PROMETHEE는 다른 기법들보다 효율적으로 이원비교를 수행할 수 있다[6].

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구로서 품질 브로커와 웹 서비스 선정 연구 동향에 관한 이론적 배경을 소개하고, 3장에서는 에이전트 기반 품질 브로커의 아키텍처와 웹 서비스 선정 프로세스를 설명한다. 4장에서는 웹 서비스 조합 환경에서 서비스 선정 방법에 대하여 품질 모델과 의사결정 알고리즘을 제시한다. 5장에서는 사례 연구를 통해 선정 프로세스를 적용해 보며, 기존 웹 서비스 선정 프로세스 연구와의 비교 평가를 기술한다. 6장에서는 결론 및 향후 연구 방향에 대해 기술한다.

2. 관련 연구

본 장에서는 UDDI와 웹 서비스 선정에 관한 기존 연구들에 대하여 소개한다.

2.1 UDDI

UDDI(Universal Description, Discovery and Integration)는 웹 서비스에 관한 정보를 등록하고 이를 실시간 검색할 수 있는 공용 디렉터리 또는 프로토콜의 집합체이다[7]. 간단하게 웹 서비스를 찾아주고 홍보해주는 일종의 온라인 공중 전화번호부와 유사하다. 따라서 웹 서비스 제공업체는 자사의 웹 서비스를 디렉터리에 등록하고, 외부에서 웹 서비스를 검색하는데 UDDI를 이용한다. UDDI는 비즈니스와 해당 서비스에 대한 정보를 구조화된 방법으로 수용하기 위해 설계되어 그 자체도 웹 서비스로 구현되어 있다. 따라서 UDDI를 통해 개인이나 회사가 제공하는 웹 서비스에 대한 정보를 등록하고 검색할 수 있으며, 데이터는 표준 분류법을 사용하여 분류할 수 있으므로 카테고리별로 정보를 찾을 수 있다.

오늘날 웹 서비스 레지스트리를 위한 실질적인 표준 규격으로 기능을 하고 있는 UDDI는 2000년 9월 MS, IBM, 그리고 Ariba 등에 의해 프로젝트 형태로 추진되었다. 2001년 8월까지 UDDI 비즈니스 레지스트리에는 5200개 이상의 비즈니스와 7800개의 서비스가 등록되어 있으며 매일 새로운 비즈니스와 서비스들이 등록되고 있다.

현재 널리 이용되고 있는 버전 2 스펙은 서비스 검색이 쉽고 비용 대 효과 면에서 우수하여 웹 서비스를 증진시키는 장점이 있다. 그러나 개방성 때문에 다음과 같은 단점[8]이 있다. 첫째, 등록 정보 데이터의 신뢰성에 관한 걱정이다. 즉, 누가 등록했는지 알 수 없는 정보를 신뢰해 비즈니스를

전개하기 힘들다는 것이다. 둘째, 전 세계에 공개할 수 있는 서비스 밖에 등록되어 있지 않다. UDD 레지스트리의 본질이 공개성에 있으므로, 어느 누구에게나 액세스 되어도 문제 되지 않는 내용의 웹서비스만 등록되어 있는 상황이다.

2.2 웹 서비스 선정

현재 웹 서비스 선정(web service selection)에 관한 연구는 여전히 초기 연구에 머물러 있으나, 그 필요성 및 중요성은 매우 커지고 있다.

de Moor[9]의 연구에서는 웹 서비스 선정이 소프트웨어 개발 주기 과정에서 실행되는 선정 메커니즘을 제안하였다. 서비스 선정을 소프트웨어 개발자에게만 맡기지 않고 가상 커뮤니티(virtual community)의 멤버들도 포함시켰다. 필요한 정보는 커뮤니티의 멤버들로부터 개발 하는 동안 얻을 수 있다. 그러나 선정 프로세스에서 어떻게 정보가 사용되는지에 관한 명시적인 내용은 기술되지 않았다.

Shuping[3]의 연구에서는 기존 UDDI 모델에 품질 정보를 포함하여 서비스의 품질 정보로 웹 서비스를 발견하는 모델을 제안하였다. 제안된 연구에서는 기존 웹 서비스 모델의 구성원들 외에 서비스 제공자의 서비스 등록 이전에 QoS 요구를 검증할 수 있는 웹 서비스 QoS 증명자(web service QoS certifier)를 추가하였다. 웹 서비스 QoS 증명자는 서비스 제공자에게 전달받은 QoS 요구를 검사하여 보증하거나 만족스럽지 못한 경우 품질을 하향 조정하여 증명자의 저장소에 등록한다. QoS 보증이 증명자에 의해 발행되면 제공자는 서비스의 기능적 명세와 연관된 품질을 함께 UDDI 레지스트리에 등록한다. UDDI 레지스트리는 보증의 존재를 검사하기 위해 증명자와 통신하며, 성공적인 경우 서비스를 등록한다. 소비자에 의해 웹 서비스가 발견되면, WSDL과 보증된 QoS 정보가 소비자에게 전달된다. 그러나 서비스 선정이 여전히 소비자에 의해 이루어지므로, 만약 수천 개의 서비스가 검색된다면 선정 과정이 어려워진다. 서비스 소비자의 기능과 품질 요구를 만족하는 가장 적합한 서비스를 선정하는 것은 품질 브로커와 같은 자동화 된 시스템 모델에 의해 수행되는 것이 바람직하다.

Liu[10]의 연구에서는 웹 서비스 선정이 실행 시간 중에 QoS 연산과 정제에 기반하여 실행되는 동적인 선정 메커니즘을 제안하였다. 제안한 시스템은 웹 서비스 선정을 위한 확장 가능한 QoS 연산 모델을 사용한다. 웹 서비스는 너무 다양하므로 단일의 정적 모델은 모든 QoS 매개변수들을 정의할 수 없으며, 반면에 한 서비스를 위한 특정 도메인(domain-specific)의 매개변수는 다른 서비스에 완전하게 적용할 수 없다. 따라서 수행 가격(execution price), 수행 기간(execution duration), 평판(reputation)을 포함한 많은 일반적인 품질 기준을 정의하였다. 수행 가격은 서비스 요청자가 서비스를 사용하기 위해 서비스 제공자에게 지불해야 하는 비용이다. 수행 기간은 서비스를 호출하여 그 결과를 되돌려 받는데 걸리는 시간이다. 평판은 사용하는 특정 웹 서비스에 대해 사용자에게 의해 기술될 수 있는 매개변수이다.

이러한 QoS 매개변수들은 서비스 제공자로부터 정보를 얻거나 클라이언트 측의 수행 모니터링에 의해 결정될 수 있다. 선정을 수행하기 위해, 시스템 중심의 QoS 레지스트리는 클라이언트들로부터 수집된 데이터를 얻어 웹 서비스 데이터의 행렬에 저장한다. 행렬의 행과 열은 각각 웹 서비스와 QoS 매개변수를 표현한다. 그리고 정규화와 같은 데이터 연산을 수행한다. 클라이언트는 레지스트리를 액세스 할 수 있으며 클라이언트가 선호하는 매개변수에 기반하여 서비스가 주어진다.

Tao[11]의 연구에서는 웹 서비스 소비자와 제공자들 사이에 QoS 브로커 모듈을 정의한 QCWS(QoS-Capable Web Service) 아키텍처를 제안하였다. QoS 브로커는 서비스 제공자들에 대한 QoS 정보를 수집하고, 소비자를 위해 서비스 선정 결정을 한다. 서비스 선정 과정에서 소비자의 QoS 요구를 만족하기 위해 후보 서비스를 제공하는 공급자 서버와 협상을 한다. Tao의 연구에서는 제공자의 서버에서 서비스마다 QoS 정보를 제공한다고 가정하고 있으나, 실제로 오늘날의 대부분의 웹 서비스 제공자들은 서비스를 QoS 지원과 함께 설치하지 않는다는 사실을 고려하지 않았다.

Patrick[12]의 연구에서는 협상 전략중 하나인 logrolling을 사용하여 웹 서비스 제공자와 소비자 간에 더욱 선호되는 이슈와 덜 선호되는 이슈를 거래함으로써 두 이슈(QoS와 CoS) 사이의 전체 가치를 증가시킬 수 있는 최적의 조합을 제시하였다. 이 모델에서 QoS는 성능 지향 능력(거리, 시간), CoS(Cost of Service)는 QoS를 확보하기 위해 요구되는 리소스(자본, 하드웨어, 소프트웨어, 네트워크 대역폭)에 관련되며, 두 이슈를 수량화하기 위해 토큰 기반 접근법(token-based approach)을 제안 하였다. 제안된 토큰 기반 접근법에서 리소스는 QoS 토큰 단위로 측정되고 Dollar는 CoS 토큰 단위로 측정된다. 일단 QoS와 CoS가 토큰으로 측정되면, 웹 서비스 제공자와 소비자 사이의 토큰 거래가 가능해지며, 그 결과로 리소스의 효율적인 할당이 가능하다. 그러나 QoS와 CoS에 포함된 모든 세부 이슈들을 제외한 채 오직 두 이슈 그룹만을 고려하고 있어, 다자간(multi-party), 다중 이슈(multi-issue)를 다루지 못하는 단점이 있다.

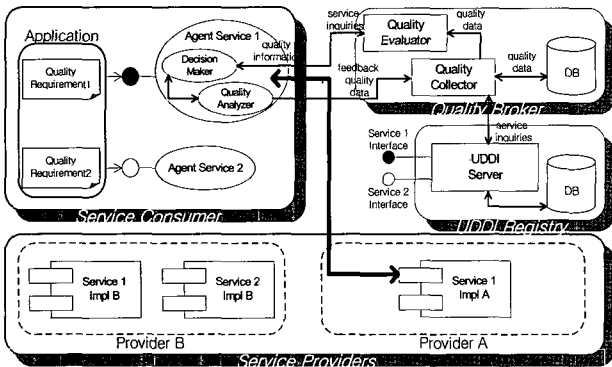
Julian[13]의 연구에서는 소비자와 웹 서비스 사이에 상호 작용을 모델링하기 위해 QoS 정보에 기반한 동적인 선정 방법을 제안하였다. 제안된 연구에서는 소비자가 다수의 구문적으로 동일한 웹 서비스들 중에서 어느 것이 최상의 서비스를 제공할 것인지에 대하여 추론하도록 허용하기 위해 규칙 기반 전문가 시스템(rule-base expert system) 또는 나이브 베이저안 분류자(naive Bayesian classifier)를 사용하였다. 또한, 공유되는 경험 정보를 얻기 위해서 사용자측 확장 접근법(client-side augmentation approach)을 선택하였다. QoS 속성에 관련된 경험 정보는 가용성, 신뢰성, 수행 시간이며, 공용으로 액세스 할 수 있는 QoS 포럼에 저장된다. QoS 포럼은 시맨틱 모델들의 벡터를 리턴 하며, 추론 엔진은 그들로부터 경험 정보를 추출한다. 추출된 경험 정

보와 가중치들과의 집합인 평가 방정식을 통해 가장 높은 가중치의 합을 갖는 서비스를 선택하였다. 그러나 Julian의 연구에서는 웹 서비스 선정 문제가 과거 경험의 관점으로만 다루어졌으며, 가격은 고려하지 않아 협상 전략에 사용되지 않았다.

3. 품질 브로커 및 서비스 선정 프로세스

3.1 품질 브로커의 아키텍처

본 절에서는 최적의 서비스 선정을 위해 품질 속성에 기반을 둔 에이전트 기반 브로커 아키텍처를 제안한다. 제안된 아키텍처에는 (그림 1)과 같이 서비스 공급자, 소비자, UDDI 레지스트리로 구성되는 기존 웹 서비스 모델에 추가적으로 품질 정보를 공유하는 기능이 확장된 품질 브로커가 포함된다.



(그림 1) 품질 브로커의 아키텍처

3.1.1 클라이언트 에이전트(Client Agent)

에이전트는 품질 요구에 정확히 일치하는 서비스를 발견하지 못할 때 다양한 의사 결정 알고리즘을 사용하여 최적의 서비스를 선정할 수 있다. 서비스 선정이 이루어지면, 에이전트는 모니터링을 통해 품질 정보를 수집하며 품질 브로커에 피드백 한다. 에이전트들에 의해 계속적으로 피드백 되는 서비스의 품질 정보는 향후 선정 과정에서 소비자 자신뿐만 아니라 다른 소비자에게도 유용한 정보가 된다.

의사 결정기(Decision Maker)는 에이전트 모듈의 핵심이다. 소비자의 애플리케이션을 위한 서비스 선정과 호출을 처리한다. 애플리케이션의 서비스 요청과 품질 요구를 받은 후에, 의사 결정기는 적합한 서비스를 찾기 위해 브로커에 의뢰한다. 하나 이상의 서비스 구현 후보가 발견되고 품질 요구와 후보들의 품질 측정값이 일치하지 않는다면, 의사 결정 알고리즘으로 대표되는 선정 알고리즘이 가장 적합한 하나를 선택하기 위해 사용된다. 일단 한 후보가 선정되고 호출되면, 품질 분석기(Quality Analyzer)는 해당 서비스의 품질 속성 값을 검사하고 문제가 있다면 다음 순위의 서비스 후보를 식별, 호출한다. 사용이 완료되면 해당 서비스 구현의 품질 데이터를 품질 브로커의 품질 수집기로 피드백 한다.

3.1.2 품질 브로커(Quality Broker)

품질 브로커는 UDDI 레지스트리에 요청한 서비스 인터페이스를 구현하는 서비스들을 질의하고, 에이전트에 의해 피드백 되는 품질 데이터를 공유한다.

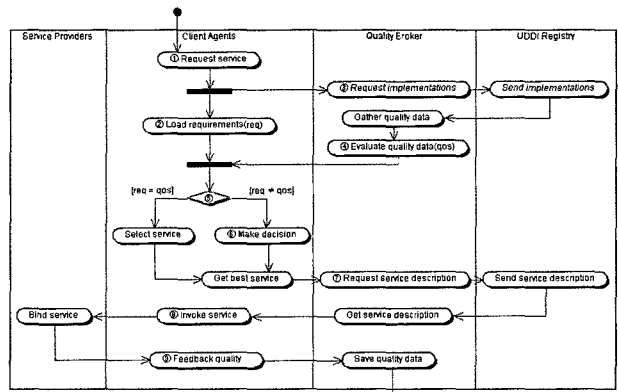
품질 수집기(Quality Collector)는 에이전트가 요청한 서비스에 대한 구현 목록을 얻기 위해 UDDI 레지스트리를 검색하며, 목록상의 서비스 구현에 해당하는 품질 데이터를 찾기 위해 품질 데이터베이스를 검사한다. 품질 수집기는 품질 데이터의 수집이 완료된 이후에 그 정보를 품질 평가기에 보낸다. 또한, 에이전트에 의해 피드백 되는 서비스의 품질 데이터를 수집하여 품질 데이터베이스에 저장된다.

품질 평가기(Quality Evaluator)는 서비스에 대한 품질 속성 값을 얻기 위해 품질 데이터를 기반으로 측정한다. 품질 데이터는 응답 시간의 경우 서비스 전송, 호출 시각이며 가용성의 경우 성공적 실행 건수, 전체 호출 건수에 해당한다. 에이전트의 의사 결정기는 서비스 선정 과정 동안 품질 속성 값을 사용한다.

3.2 서비스 선정 프로세스

(그림 2)에서는 구획 면으로 제안한 아키텍처의 구성원들을 구분하고 구성원들 간의 상호 작용을 통한 서비스 선정 절차를 액티비티 다이어그램을 사용해 기술하였다.

- 1단계. 서비스 소비자는 필요한 서비스마다 에이전트를 실행화 한다. 에이전트는 품질 브로커에 특정 인터페이스를 구현하는 서비스를 요청한다.
- 2단계. 에이전트는 소비자의 품질 요구 정보를 로드한다.
- 3단계. 품질 브로커는 특정 서비스 인터페이스를 구현하는 서비스 구현들을 검색하기 위해 UDDI 레지스트리에 질의한다. 품질 브로커는 검색된 서비스 구현에 대한 품질 데이터를 수집한다.
- 4단계. 수집된 품질 데이터를 바탕으로 측정 방법에 따라 품질 측정값을 구한다.
- 5단계. 에이전트가 소비자의 품질 요구와 품질 측정값을 비교한다. 만약 일치하는 서비스를 발견한다면 해당 서비스 구현을 선정하며, 그렇지 못할 경우 의사 결정 과정을 통해 최적의 서비스를 선정한다.



(그림 2) 서비스 선정 프로세스

- 6단계. 의사 결정 과정에서는 품질 속성 값들을 PROMETHEE와 같은 의사 결정 알고리즘에 적용하여, 서비스 구현들의 순위번호 관계를 계산한다. 순호름량을 비교하여 그 값이 가장 큰 서비스 구현을 최적의 서비스로 선정한다.
- 7단계. 품질 브로커는 UDDI 레지스트리에 서비스 명세서를 요청한다.
- 8단계. 에이전트는 리턴 된 명세서 정보를 바탕으로 해당 웹 서비스를 호출한다.
- 9단계. 에이전트는 선정된 웹 서비스의 품질 속성들을 모니터링한 뒤, 품질 브로커에 피드백 한다. 저장된 품질 데이터들은 향후 서비스 선정 과정에서 반복적으로 사용된다.

4. 웹 서비스 조립 환경에서 서비스 선정 방안

4.1 품질 모델

웹 서비스 품질 모델은 표준화 되어 있지는 않으나, 본 논문에서는 관련 연구[3, 14, 15]에서 정의한 웹 서비스 품질 모델을 참고하였다.

사용상의 품질은 크게 성능 및 안전성 측면 품질, 트랜잭션 지원 측면 품질, 미들웨어 측면 품질, 비용과 상호운용성 측면 품질, 보안 측면 품질로 나뉜다. 웹 서비스의 특성상 웹 서비스를 소비자가 사용하는 시점에서의 품질이 서비스 선정에 있어서 중요한 결정 요소로 작용하므로, 본 논문에서는 성능 및 안전성 측면 품질과 비용 측면 품질만을 고려하였다.

성능 및 안전성 측면 품질은 제공되는 웹 서비스가 얼마나 성능이 뛰어나고, 안전적으로 서비스를 제공하고 있는지

를 의미한다. 대표적인 성능 관련 품질 속성으로는 요청을 보내고 응답을 받는데 걸린 시간을 의미하는 응답 시간(response time)과 주어진 시간동안 제공된 웹 서비스 요청 수를 의미하는 처리율(throughput)이 있다. 안전성 관련 품질 속성으로는 서비스가 이용 가능한지를 나타내는 정도를 의미하는 이용가능성(availability)과 서비스 신뢰성에 관련하여 특정 시간대에서 실패 확률을 의미하는 신뢰성(reliability)이 있다. 마지막으로 비용 측면 품질은 서비스 요청 비용(cost)을 의미한다.

소비자는 서비스 선정 시에 품질 속성마다 gold, silver, bronze와 같은 차별화 된 서비스 레벨을 선택할 수 있으며 [16], <표 1>은 선택된 품질 속성들의 측정 방법[15]과 서비스 레벨의 범위를 나타낸다.

4.2 의사 결정 알고리즘

본 논문에서는 5가지 품질 속성 정보를 기반으로 최적의 서비스를 선정하기 위한 의사 결정 알고리즘은 PROMETHEE [6, 17]를 기반으로 기술하였다.

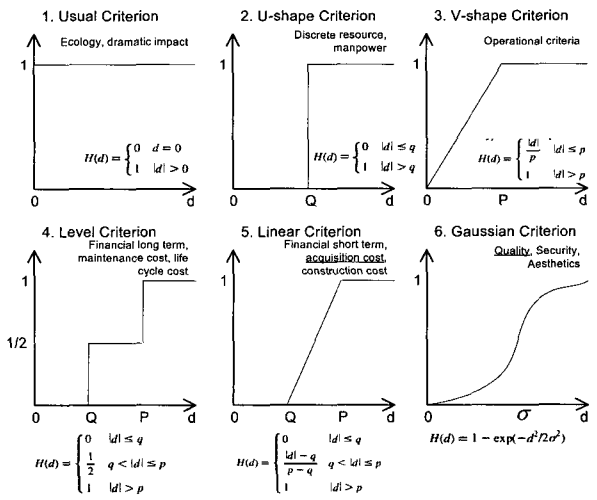
1 단계로 평가기준별 최저/최고(Min/Max), 가중치(weight), 선호 함수(preferance function), 임계치(threshold)를 결정한다. 최저/최고항목에서 최고(Max)는 평가기준의 값이 클수록 해당 웹 서비스 선정에 긍정적인 영향을 주는 지표의 의미하고, 최저(Min)는 반대의 경우를 의미한다. 가중치는 서비스 소비자들의 의견에 의해 결정된다. 선호 함수(preferance function)는 (그림 3)과 같이 여섯 가지로 정의되고 있으며 [18], 각 함수는 평가기준의 종류에 따라 선택된다.

예를 들어, 비용 같은 양적인 기준은 반정량적(상, 중, 하 척도) 기준도 계단식 함수 형태를 가지는 까닭에, (그림 3)의 type 5번 linear 함수가 선택된다[19]. 품질은 품질 속성에 가장 자연스러운 선호 형태를 가지는 type 6번 gaussian 함수가 선택된다[20, 21].

여기서 P, Q, σ는 평가기준별 선호 함수의 구체적 형태를 결정하기 위해 필요한 preference, indifferent, gaussian 임

<표 1> 품질 속성의 측정 방법과 레벨

품질 속성	측정 방법	서비스 레벨
Performance	Response Time	서비스 전송 시작-서비스 호출 시작 Response Time ms
	Throughput	지정 시간동안의 서비스 호출 건수 Throughput requests/s
Safety	Availability	P(성공적인 실행 건수 / 전체 호출 건수) Availability probability
	Reliability	1-PAvailability Reliability probability
Cost	Cost	C(Enactment)+C(Licensing) Enactment: 관리와 모니터링 비용 Licensing: 라이선스 비용 Cost EUR



(그림 3) 선호 함수와 함수식

계치를 의미한다. P는 한 대안이 다른 대안에 비해 선호되는 지점을 말하며, 가장 희망하는 선호도의 끝 값과 넘지 말아야 할(strictly not beyond) 값 사이의 차이로 계산된다. Q는 두 대안들이 동일한 것으로 고려되는 간격을 정의하며, 가장 희망하는 선호도의 끝 값과 받아들일 수 있는(acceptable) 값 사이의 차이로 계산된다[22]. 마지막으로, σ 는 indifferent 임계치와 preference 임계치 사이의 값으로서 계산된다.

2단계로 선호 지수(preference index)를 계산한다. 먼저 품질 기준별로 선호함수(preference function) 값을 구해야 하며, 기준 k의 선호함수 값 $p_k(WS_i, WS_j)$ 는 두 서비스의 점수 차이를 선호 함수식에 적용하여 얻을 수 있다. 선호 지수 $\Pi(WS_i, WS_j)$ 는 WS_j 에 비교하여 WS_i 에 대한 소비자의 선호 강도를 의미한다. 선호 지수는 (식 1)과 같이 선호함수 값에 평가기준별 가중치를 곱하여 계산된다.

$$\Pi(WS_i, WS_j) = \sum_{k=1}^K w_k \cdot p_k(WS_i, WS_j) \quad (식 1)$$

3단계로 선호의 선호유출량(leaving flow), 선호유입량(entering flow), 순흐름량(net flow)을 계산한다. WS_i 와 WS_j 에 대한 평가기준별 선호지수 $\Pi(WS_i, WS_j)$ 를 모두 합하여 두 서비스간의 선호지수를 계산한다. 다음 (식 2)로부터 (식 4)와 같은 선호유출량(Φ^+), 선호유입량(Φ^-), 그리고 선호유출량과 선호유입량을 차감한 순흐름량(Φ)을 계산한다. $\Phi^+(WS_i)$ 는 WS_i 가 나머지 모든 서비스들을 지배하는(dominating) 정도를 나타내며, $\Phi^-(WS_i)$ 는 WS_i 가 나머지 모든 서비스들에 의해 지배되는(dominated) 정도를 나타낸다. WS_i 가 선호유출량이 높고 선호유입량이 낮을수록 나머지 서비스들보다 우월하다는 것을 나타낸다. 따라서 $\Phi(WS)$ 가 클수록 해당 서비스가 우월한 대안임을 의미한다.

$$\Phi^+(WS_i) = \frac{1}{N} \sum_{n=1}^N \Pi(WS_i, WS_n) \quad (식 2)$$

$$\Phi^-(WS_i) = \frac{1}{N} \sum_{n=1}^N \Pi(WS_n, WS_i) \quad (식 3)$$

$$\Phi(WS_i) = \Phi^+(WS_i) - \Phi^-(WS_i) \quad (식 4)$$

4단계로 순위선호 관계(outranking relation)를 평가한다. 각 서비스의 순흐름량을 기준으로 서비스들의 우선순위를 도출한다. 출발점에 있는 서비스는 가장 우월한 서비스를 의미하며, 종점에 있는 서비스는 가장 열등한 서비스를 의미한다.

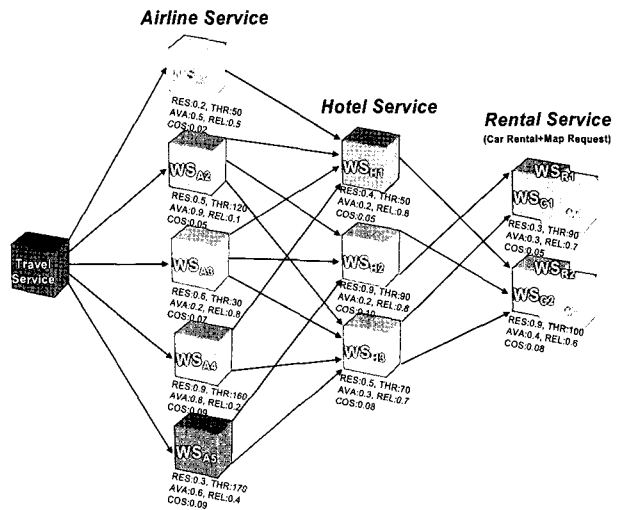
5. 사례 연구

5.1 시나리오

웹 서비스 조합(web service composition)의 대표적인 시

나리오는 (그림 4)와 같이 항공 티켓 예약, 호텔 예약, 자동차 렌트, 지도 요청 서비스를 조합하여 여행 계획 서비스를 구성하는 시나리오이다. 서비스 조합의 방향은 개별적인 웹 서비스들의 통합 순서를 가리킨다[23]. 서비스 조합은 수평, 수직, 또는 양 방향으로 조립될 수 있다. 수직 조합(vertical composition)은 서비스들이 의존 관계에 기반을 두어 순차적인 순서에 의해 조합되는 것을 의미한다. 예를 들면, 호텔 예약 서비스는 항공 티켓이 발급될 때까지 수행될 수 없으므로 수직 조합을 사용한다. 반면에 수평 조합(horizontal composition)은 독립적인 서비스들이 병렬로 조합되는 것을 의미한다.

예를 들면, 자동차 렌트와 지도 요청 서비스는 수평적인 방법으로 동시에 수행될 수 있다. 서비스 조합의 방향은 품질 속성의 측정에도 영향을 끼친다[24]. 예를 들면, 비용은 수평, 수직 조합 모두 합계로 측정된다. 그러나 응답 시간은 수평 조합에서는 조합된 서비스들 사이의 최대 응답 시간으로, 수직 조합에서는 합계로 측정된다.



(그림 4) 여행 계획 서비스

5.2 평가기준별 최저/최고, 가중치, 선호 함수, 임계치 결정

<표 2>는 다섯 가지 평가기준을 위한 의사결정표를 나타낸다. 대안으로는 (그림 4)의 여행 계획 서비스를 구성하는 세 개의 서비스(airline, hotel, rental service)를 대상으로 한다. 그 중 렌탈 서비스는 자동차 렌트와 지도 요청 서비스로 구성된 수평 조합 서비스이므로, 각 평가기준들은 두 서비스의 평가기준 값들로 계산된다. 응답 시간은 두 서비스의 최대 응답 시간으로, 처리율은 합계, 가용성은 최소 확률, 그리고 비용은 합계로 계산된다.

최저/최고 항목에서는 응답 시간, 처리율, 가용성은 값이 클수록 해당 웹 서비스 선정에 긍정적인 영향을 주는 지표이므로 최고(Max)로 설정한다. 응답 시간은 신뢰할 수 있는 서비스를 액세스 할 때 발생하는 것보다 실패가 발생할 때 더 낮아질 수 있다. 이 때문에, 응답 시간의 디폴트 가중치는 가용성 또는 신뢰성의 그것보다 더 낮은 경향이 있으나

〈표 2〉 평가기준을 위한 의사결정표

Criteria		RES	THR	AVA	REL	COS
Airline Service	WSA1	0.2	50	0.5	0.5	0.02
	WSA2	0.5	120	0.9	0.1	0.05
	WSA3	0.6	30	0.2	0.8	0.07
	WSA4	0.9	160	0.8	0.2	0.09
	WSA5	0.3	170	0.6	0.4	0.09
Hotel Service	WSH1	0.4	50	0.2	0.8	0.05
	WSH2	0.9	90	0.2	0.8	0.10
	WSH3	0.5	70	0.3	0.7	0.08
Car Rental	WSC1	0.3	60	0.3	0.7	0.03
	WSC2	0.9	80	0.4	0.6	0.02
Map Request	WSM1	0.2	30	0.7	0.3	0.02
	WSM2	0.8	20	0.8	0.2	0.06
Rental Service	WSR1	0.3	90	0.3	0.7	0.05
	WSR2	0.9	100	0.4	0.6	0.08
Min/Max	Min	Max	Max	Min	Min	
Weight	0.2	0.2	0.2	0.2	0.2	
Preference Function	Gaussian	Gaussian	Gaussian	Gaussian	Linear	
Indifference Threshold	-	-	-	-	0.06	
Preference Threshold	-	-	-	-	0.09	
Gaussian Threshold	0.8	112	0.7	0.7	-	

[13] 분석의 편의를 위하여 다른 평가기준들의 가중치와 동일하게 부여한다.

선호 함수는 비용을 제외한 모든 품질 속성들은 Gaussian형 함수를 사용하고, 비용은 Linear형 함수를 사용한다(그림 3 참조). 다음 단계로 의사 결정자는 품질 속성별 선택된 선호 함수의 구체적 형태를 결정하기 위해 선호 임계치를 부여한다.

비용의 indifference 임계치(Q)는 가장 희망하는 선호도의 끝 값은 0이며, 받아들일 수 있는 값은 Silver 레벨로 가정한다. 따라서 $Q = (0.06((0.04+0.07)/2)-0) = 0.06$ 이다. 비용의 preference 임계치(P)는 넘지 말아야 할 값을 Bronze 레벨로 가정한다. 따라서 $P = (0.09-0) = 0.09$ 이다. 이 결과는 다음과 같이 해석될 수 있다. 웹 서비스의 비용을 고려함에 있어서 0.06의 차이는 상관하지 않는다는 것이다. 그러나 0.09만큼 가격 차이가 발생한다면, 서비스 소비자는 가장 싼 서비스를 선택할 것임을 의미한다.

Gaussian형 함수에서 gaussian 임계치는 indifference 임계치와 preference 임계치 사이의 중간 값으로 계산한다. 분석의 편의상 indifference 임계치는 Silver 레벨을, preference 임계치는 Bronze 레벨을 사용한다. 예를 들어, 응답 시간의 경우 indifference 임계치(Silver 레벨의 평균값(0.6)-가장 희망하는 선호도의 끝 값(0))와 preference 임계치(Bronze 레벨의 평균값(0.9)-가장 희망하는 선호도의 끝 값(0))의 중간값으로 계산된다. 즉, $(Q(0.6)+P(0.9))/2 = 0.8$ 이다. 처리율의 경우 indifference 임계치(가장 희망하는 선호도의 끝 값(200)-Silver 레벨의 평균값(126)=74)와 preference 임계치(가장 희망하는 선호도의 끝 값(200)-Bronze 레벨의 평균값(50)=150)의 중간 값으로 계산된다. 즉, $(Q(74)+P(150))/2 = 112$ 이다.

가용성의 경우도 indifference 임계치(가장 희망하는 선호도의 끝 값(1)-Silver 레벨의 평균값(0.5)=0.5)와 preference 임계치(가장 희망하는 선호도의 끝 값(1)-Bronze 레벨의 평

균값(0.2)=0.8)의 중간값으로 계산된다. 즉, $(Q(0.5)+P(0.8))/2 = 0.7$ 이다. 계산된 선호 임계치는 선호 함수와 더불어 선호 함수의 값을 0과 1사이의 수치로 정규화 시킴을 알 수 있다.

5.3 선호 지수와 순흐름량 계산

(식 1)에 따라 선호 지수를 알기 위해서는 평가기준별 선호 함수 값을 계산해야 한다. 다음 <표 3>은 항공티켓 예약 서비스의 평가기준별 선호함수 값을 나타내었다. 예를 들어, 응답 시간(RES)인 경우의 WS_{A1} 과 WS_{A2} 의 선호 함수 값 ($p_{RES}(WS_{A1}, WS_{A2})$) 0.0679와 WS_{A2} 와 WS_{A1} 의 선호 함수 값 ($p_{RES}(WS_{A2}, WS_{A1})$) 0.0000은 <표 3>과 같이 계산되었다.

응답시간은 작을수록 웹 서비스 선정에 긍정적 영향을 주는 지표이므로, 두 서비스의 평가점수 차이인 $x(0.2-0.5=-0.3)$ 가 음수인 경우에는 Gaussian형 함수식에 x 와 gaussian 임계치(0.8)를 대입하여 0.0679라는 값을 얻는다. 반대로, 양수인 경우에는 선호되지 않아 0으로 계산한다.

또 다른 예로, 처리율(THR)인 경우의 WS_{A1} 과 WS_{A2} 의 선호 함수 값 ($p_{THR}(WS_{A1}, WS_{A2})$) 0.0000과 WS_{A2} 와 WS_{A1} 의 선호 함수 값 ($p_{THR}(WS_{A2}, WS_{A1})$) 0.1774는 다음과 같이 계산되었다. 처리율은 클수록 웹 서비스 선정에 긍정적 영향을 주는 지표이므로, 두 서비스의 평가점수 차이인 $x(50-120=-70)$ 가 음수인 경우에는 선호되지 않아 0으로 계산한다. 반대로 양수인 경우에는 Gaussian형 함수식에 x 와 gaussian 임계치(112)를 대입하여 0.1774라는 값을 얻는다.

선호 지수는 평가기준별 선호함수 값들을 합한 후 가중치를 곱하면 계산할 수 있다. 예를 들어, WS_{A2} 과 WS_{A3} 사이의 선호 지수는 아래 (식 5)에서와 같이 <표 3>의 두 서비스가 만나는 칸의 평가기준별 선호함수 값들을 모두 합한 후

〈표 3〉 평가기준별 선호함수 값

	WS_{A1}	WS_{A2}	WS_{A3}	WS_{A4}	WS_{A5}
WS_{A1}		0.0679 0.0000 0.0000 0.0000 0.0000	0.1175 0.0158 0.0000 0.0878 0.0000	0.3181 0.0000 0.0000 0.0000 0.3333	0.0078 0.0000 0.0000 0.0000 0.3333
WS_{A2}	0.0000 0.1774 0.1506 0.1506 0.0000		0.0078 0.2759 0.3935 0.3935 0.0000	0.1175 0.0000 0.0102 0.0102 0.0000	0.0308 0.0000 0.0878 0.0878 0.0000
WS_{A3}	0.0000 0.0000 0.0878 0.0000 0.0000	0.0000 0.0000		0.0679 0.0000 0.0000 0.0000 0.0000	0.0000 0.0000 0.0000 0.0000 0.0000
WS_{A4}	0.0000 0.3826 0.0878 0.0878 0.0000	0.0000 0.0618 0.0000 0.0000 0.0000	0.0000 0.4901 0.3074 0.3074 0.0000		0.0000 0.0000 0.0400 0.0400 0.0000
WS_{A5}	0.0000 0.4367 0.0102 0.0102 0.0000	0.0000 0.0949 0.0000 0.0000 0.0000	0.0679 0.5422 0.1506 0.1506 0.0000	0.2452 0.0040 0.0000 0.0000 0.0000	

가중치(0.2)를 곱하면 된다.

$$\Pi(WS_2, WS_3) = \frac{1}{5}(0.0078 + 0.2759 + 0.3935 + 0.3935 + 0) \quad (\text{식 5})$$

항공티켓 예약 서비스의 모든 서비스의 쌍 (WS_i, WS_j)에 상위 과정을 적용하여, 평가기준별 선호지수와 선호유출량, 선호유입량, 순흐름량을 계산한 결과는 다음 <표 4>와 같다. 각 WS_i 의 선호유출량과 선호유입량은 <표 4>의 각각 i 번째 행과 열의 모든 선호지수를 합하여 평균한 값이며, 순흐름량은 두 값을 차감한 값이다.

<표 4> 선호유출량, 선호유입량, 순흐름량 계산

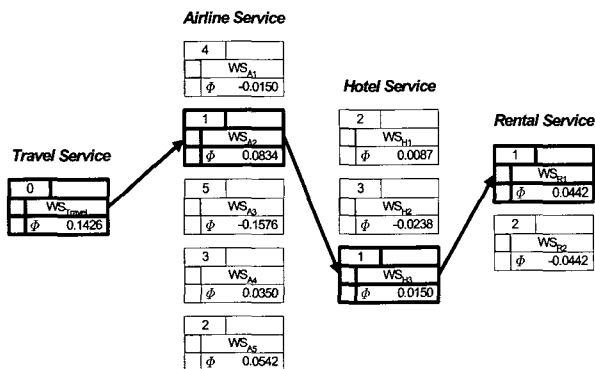
	WS_{A1}	WS_{A2}	WS_{A3}	WS_{A4}	WS_{A5}	Φ^+	Φ	Ranks
WS_{A1}		0.0136	0.0442	0.1303	0.0682	0.0641	-0.0150	4
WS_{A2}	0.0957		0.2141	0.0276	0.0413	0.0947	0.0834	1
WS_{A3}	0.0176	0.0000		0.0136	0.0000	0.0078	-0.1576	5
WS_{A4}	0.1116	0.0124	0.2210		0.0160	0.0903	0.0350	3
WS_{A5}	0.0914	0.0190	0.1823	0.0498		0.0856	0.0542	2

Φ^-	0.0791	0.0113	0.1654	0.0553	0.0314
----------	--------	--------	--------	--------	--------

5.4 순위선호 관계 평가

<표 4>의 순흐름량을 근거로 서비스들의 순위선호 관계는 다음 (그림 5)와 같다.

(그림 5)에서 여행 서비스(WS_{Travel})는 항공티켓 예약 서비스에서는 WS_{A2} , 호텔 예약 서비스에서는 WS_{H3} , 마지막으로 렌탈 서비스에서는 WS_{R1} 로 선정된 서비스가 최적의 서비스로 조합되었음을 알 수 있다. 만약 선정된 서비스가 서비스 호출 과정에서 문제가 발생한다면 순흐름량의 값에 근거하여 다음 순위의 서비스 후보를 호출한다.



(그림 5) 순흐름량에 근거한 최적의 서비스 조합

5.5 비교 평가

앞서 관련 연구에서 기술한 기존 연구들과 본 논문에서 제안한 연구와의 비교 평가는 <표 5>와 같으며, 각각의 특징을 살펴보면 다음과 같다.

<표 5> 서비스 선정 연구의 비교 평가

연구 특징	de Moor[9]	Shuping[3]	Liu[10]	Tao[11]	Julian[13]	제안 연구
사용 시점	설계시점	실행시점	실행시점	실행시점	실행시점	실행시점
정보 종류	사용자, 커뮤니티 요구사항	QoS 데이터	QoS 데이터	QoS 정보	특정 클라이언트의 데이터	QoS 속성
획득 방법	개발 주기에 사용자들을 포함	서비스 제공자의 등록	클라이언트의 피드백	서비스 제공자의 등록	클라이언트의 피드백	에이전트의 피드백
사용 방법	명시적 방법이 없음	명시적 방법이 없음	정규화	자원 할당 알고리즘	규칙기반 전문가 시스템, 나이트 베이지언 분류자	다기준 의사결정 기법

첫째, 선정 메커니즘의 사용 시점. de Moor의 연구를 제외하고는 비교 대상이 되는 모든 연구들은 실행 시간에 동적으로 웹 서비스를 선정하였다. 이에 반해 de Moor의 연구는 사용자와 커뮤니티 멤버들이 설계 시간에 선정 메커니즘을 수행하였다.

둘째, 선정 프로세스에 필요한 정보의 종류. de Moor의 연구를 제외하고는 대부분의 연구들은 QoS 데이터를 선정 프로세스에서 사용하였다. 본 논문에서 제안한 연구에서는 웹 서비스의 특성상 웹 서비스를 소비자가 사용하는 시점에서의 품질이 웹 서비스 선택에 있어서 중요한 결정 요소로 작용하므로, 성능, 안전성, 비용 관련 QoS 속성만을 고려하였다.

셋째, 정보의 획득 방법. Liu와 Day의 연구에서는 클라이언트가 직접 QoS 레지스트리 또는 포럼 시스템에 피드백 하는데 반해 제안한 연구에서는 에이전트를 사용해 브로커에 피드백 한다. 따라서 클라이언트를 대신해 자율적인 정보 피드백이 가능하다.

넷째, 정보가 선정 프로세스에 사용되는 방법. de Moor와 Shuping의 연구에서는 선정 프로세스에서 어떻게 정보가 사용되는 지에 관한 명시적인 설명을 하지 않았다. 이에 반해, Liu의 연구는 정규화와 같은 연산을 수행하며, Day의 연구에서는 규칙 기반 전문가 시스템 또는 나이트 베이지언 분류자를 통해 추론한다. 제안한 연구에서는 획득된 정보가 PROMETHEE와 같은 다기준 의사결정 기법에 사용되었다. PROMETHEE는 다른 기법들과 비교하여 선정 과정에서 유사 서비스들이 빈번히 추가되거나 삭제되더라도 이원비교를 다시 수행해야 하는 문제를 극복할 수 있는 장점이 있어 사용되었다.

6. 결론과 향후 연구과제

본 논문에서는 최적의 품질을 제공하는 서비스를 선정하고자 하는 서비스 소비자의 품질 요구를 만족시키는 에이전트 기반 품질 브로커에 대한 아키텍처와 선정 프로세스를 제안하였다. 제안된 아키텍처에서 에이전트는 품질 브로커에 저장된 품질 데이터를 바탕으로 서비스 선정 문제를

PROMETHEE와 같은 다기준 의사결정법으로 모형화하여 애플리케이션에 최적의 서비스를 제시하였다. PROMETHEE에 기반을 둔 의사결정 알고리즘은 이원비교가 자동으로 수행되므로, 비교 서비스의 증가에서 발생할 수 있는 이원비교 재수행 문제에 대한 해결책을 제공할 수 있다.

본 논문에서 제시한 선정 프로세스를 검증하기 위하여 서비스 조합을 시나리오로 갖는 사례 연구를 제시하였다. 소비자 입장에서 품질 속성들을 선택하고 측정값을 계산하여 서비스 선정 알고리즘에 적용한 뒤 서비스마다 순위번호 관계를 도출하였다.

향후 연구방향은 서비스 소비자뿐만 아니라 공급자의 품질 제공에 기반하여 서비스 선정 문제를 협상할 수 있는 아키텍처를 제시하고 이를 이용하여 서비스들을 체계적으로 비교, 평가할 수 있는 구체적인 알고리즘 및 프로세스를 마련하는 것이다.

참 고 문 헌

- [1] Sandra Rogers, "Web Services Software 2004-2008 Forecast," IDC, Apr., 2004.
- [2] Ovum, "Web Services Market Overview," Ovum Research Report, Sept., 2002.
- [3] S. Ran, "A model for web services discovery with QoS," ACM SIGecom Exchanges, Vol.4, Issue.1, pp.1-10, Spring, 2003.
- [4] Mourad Ouzzani, Athman Bouguettaya, "Efficient Access to Web Services," IEEE Internet Computing, Vol.8, No.2, pp.34-44, Mar., 2004.
- [5] K.H.Bennett, and others, "A Broker Architecture for Integrating Data Using a Web Services Environment," ICSOC, Vol.2910, pp.409-422, 2003.
- [6] 민재형, 송영민, "PROMETHEE를 이용한 다기준의사결정", 한국경영과학회 추계학술대회 논문집, pp.229-232, 2003.
- [7] Tom Bellwood, Luc Clément, David Ehnebuske, Andrew Hatley, Maryann Hondo, Yin Leng Husband, Karsten Januszewski, Sam Lee, Barbara McKee, Joel Munter, Claus von Riegen, "UDDI Version 3.0 Published Specification," July, 2002. (See http://uddi.org/pubs/uddi_v3.htm)
- [8] 한국 쉐어 마이크로시스템즈, "UDDI 레지스트리의 가능성 탐구", Sun microsystems monthly newsletter, pp.18-25, 2003.
- [9] Aldo de Moor, Willem-Jan van den Heuvel, "Web service selection in virtual communities," In 37th Hawaii International Conference on System Sciences, Jan., 2004.
- [10] Yutu Liu, Anne Ngu, Liangzhao Zheng, "QoS computation and policing in dynamic web service selection (to appear)," Proceedings of the WWW 2004, May, 2004.
- [11] Yu, T., Lin, K.-J., "The Design of QoS Broker Algorithms for QoS-Capable Web Services," Proceedings of the e-Technology, e-Commerce and e-Service Conference, pp.17-24, 2004.
- [12] Patrick C. K. Hung and Haifei Li, "Web Services Discovery Based on the Trade-off between Quality and Cost of Service: A Token-based Approach," ACM SIGecom Exchanges, Vol.4, No.2, pp.21-31, Aug., 2003.
- [13] Julian Day, Ralph Deters, "Selecting the Best Web Service," In Proceedings of the 14th Annual IBM Centers for Advanced Studies Conference(CASCON), pp.293-307, Oct., 2004.
- [14] 허정희 외, "웹 서비스 품질관리 동향 및 도입전략 연구", 한국전산원 연구보고서, Dec., 2003.
- [15] Chintan Patel, Kaustubh Supekar, Yugyung Lee, "A QoS Oriented Framework for Adaptive Management of Web Service based Workflows," LNCS, Vol.2736, pp.826-835, Sept., 2003.
- [16] Asit Dan, Heiko Ludwig, Giovanni Pacifici, "Web Services Differentiation with Service Level Agreements," White Paper, IBM Corporation, May, 2003.
- [17] Brans, J. P. Vincke, "A Preference Ranking Organization method(The PROMETHEE Method for Multiple Criteria Decision-Making)," Management Science, Vol.31, No.6, pp.647-656, 1985.
- [18] Chou T. Y., W. T. Lin, C. Y. Lin, W. C. Chou, "Application of the PROMETHEE technique to determine depression outlet location and flow direction in DEM," Journal of Hydrology 287, pp.49-61, 2004.
- [19] Linkov, I. A. Varghese, S. Jamil, T.P. Seager, G. Kiker, and T. Bridges, "Multi-criteria decision analysis: A framework for structuring remedial decisions at contaminated sites," In Linkov, I. and Ramadan, A. eds. Comparative Risk Assessment and Environmental Decision Making. Kluwer, pp.15-54, 2004.
- [20] John A., Jr. Lawrence, Barry Alan Pasternack, "Applied Management Science: A Computer-Integrated Approach for Decision Making," John Wiley, USA, 1998. (See <http://www.poms.ucl.ac.be/etudes/notes/qant2100/Promethee%20Shape.ppt>)
- [21] G. Waeyenbergh, B. Vannieuwenhuysse, L. Pintelon, "A model to determine the cleanliness measurement interval in an automotive paint shop," Journal of Quality in Maintenance Engineering, Vol.10, No.1, pp.37-46, 2004.
- [22] Kodikara, P.N., Perera, B.J.C., Kularathna, M.D.U.D., "Stakeholder Preference Modelling In Multi-Objective Operation Of Urban Water Supply Systems-A Case Study On Melbourne Water Supply System," International Congress on Modelling and Simulation, pp.1539-1545, 2005.
- [23] Lina Zhou, Dongsong Zhang, "Trust-supported Automatic Web Service Composition," The Third Workshop on e-Business(Web 2004), Dec., 2004.

[24] Aversano, L., Canfora, G., Ciampi, A., "An algorithm for Web Service discovery through their composition," Proceedings of the IEEE International Conference on Web Services, 2004.



서 영 준

e-mail : yjseo@khu.ac.kr
1999년 경희대학교 전자계산공학과 (공학사)
2001년 경희대학교 전자계산공학과 (공학석사)
2003년 경희대학교 전자계산공학과 박사과정 수료

2002년~현재 경희대학교 전자정보대학 강사
관심분야: 웹 서비스, CBSE, 소프트웨어 재사용 등



송 영 재

e-mail : yjsong@khu.ac.kr
1969년 인하대학교 전기공학과(공학사)
1976년 일본 Keio University 전산학과 (공학석사)
1979년 명지대학교 전산학과(공학박사)
1971년~1973년 일본 Toyo Seiko 연구원

1982년~1983년 미국 Maryland University 전산학과 연구교수
1984년~1989년 경희대학교 전자계산소장
1989년~1990년 일본 Keio University 전산학과 객원교수
1993년~1995년 경희대학교 교무처장
1996년~1998년 경희대학교 공과대학장
1998년~2000년 경희대학교 기획조정실장
2001년~2002년 경희대학교 산업정보대학원장
1976년~현재 경희대학교 전자정보대학 교수
관심분야: 웹 서비스, CBSE, CASE 도구, 소프트웨어 재사용