

SOA를 위한 온톨로지 기반의 서비스 도출 방법론*

장효선** · 박세권*** · 류승완*** · 신동천****

A Methodology for Ontology-based Service Drawing for SOA*

Hyo-Sun Jang** · Sei-Kwon Park*** · Seung-Wan Ryu*** · Dong-Cheon Shin****

■ Abstract ■

Even though several methodologies for SOA(Service Oriented Architecture) have been proposed, in practical aspects most of them have some problems since they fail to propose specific policies in definition and identification of a service. This paper proposes a service modeling methodology, SOMO(Service Oriented Modeling using Ontology), which draws proper services in the process of defining and identifying services. SOMO defines a service ontology based on service definition and characteristics in SOA. The service drawing process consists of 3 steps : requirement analysis, service identification, and service definition. SOMO is expected to increase the degree of reuse and facilitates the definition and search of services by using service ontology. In addition, it clearly allows the definition and identification of services, satisfying the user requirements.

Keyword : Service Architecture, Service Modeling, Ontology, Service Drawing

논문투고일 : 2011년 01월 17일

논문수정완료일 : 2011년 05월 24일

논문게재확정일 : 2011년 06월 11일

* 이 논문은 2010년도 중앙대학교 학술연구비 지원에 의한 것임.

** 중앙대학교 정보시스템학과 석사과정

*** 중앙대학교 정보시스템학과 교수

**** 중앙대학교 정보시스템학과 교수, 교신저자

1. 서론

오늘날 세계의 경제 중심은 서비스 산업으로 빠르게 이동하고 있으며, 국가경제 뿐만 아니라 기업 경영에서도 서비스의 중요성이 높아지고 있다. 따라서 서비스 시스템에 과학적이고 체계적인 접근 방법을 도입해 생산성과 품질을 향상시키기 위한 서비스 사이언스가 주목을 받고 있다. 특히, IT 서비스 산업에서는 서비스 전략과 실행 방안을 제공할 수 있는 과학적이고 공학적인 방법론의 접근이 필요하다. 따라서 서비스를 기획하고 설계하여 제공한 후 운영과 평가 및 개선하기 위해서는 타당성 있는 모형이나 방법론이 필요하게 되며 최근 비즈니스 민첩성과 통합성, 효율성, 유지보수 비용을 최소화할 수 있는 IT 전략인 SOA(Service Oriented Architecture)가 가장 주목 받고 있다[16].

SOA를 적용하면 애플리케이션 전체나 일부가 서비스 개념으로 인식되어 재조합을 통해 새로운 비즈니스 애플리케이션을 빠르게 개발할 수 있게 되며, 이를 통해 현업 부서가 쉽고 빠르고 업무를 재구축할 수 있게 된다. 따라서 기존 IT 부서 중심의 컴퓨팅 환경이 비즈니스 중심으로 바뀌어 더욱 신속히 비즈니스 요구를 반영할 수 있게 된다. 이러한 이유 때문에 SOA는 기업 내 차세대 인프라 스트럭처의 핵심 기술로 자리 잡았다. SOA는 느슨한 연결(loosely coupled)을 통해 융통성을 증가시키며, 서비스 공급자, 서비스 요청자, 서비스 저장소라는 세 요소가 유기적으로 연계되어 있는 특징을 갖는다[14]. SOA의 핵심 특성에는 이러한 느슨한 연결 외에도 서비스 지향 비즈니스 모델링, 기민성 등을 들 수 있다.

많은 기업들이 SOA에 관심을 가지고 있으나 실제적인 도입에는 조심스러운 모습이다. 기업들의 소극적인 자세에는 몇 가지 이유가 있다. 첫 번째로 기존의 성공적인 SOA 사례가 없다는 점을 볼 수 있다. 두 번째로 단기간 구축이 어려우며 컴포넌트 구축을 위한 시간과 비용이 과다하기 때문이다. 세 번째로 비즈니스 전문가와 솔루션 전문가

들이 없다는 점이다. 네 번째는 대부분의 인터페이스를 데이터베이스와 텍스트 처리에 의존한다는 현재의 IT 환경이다. 마지막으로 기술과 비즈니스가 접목될 수 있는 마땅한 SOA 방법론이 없다는 점이다.

현재까지 제안되고 있는 SOA 방법론 중 주요 SOA 방법론으로는 SODA(Service Oriented Development of Applications)[18], SOAD(Service Oriented Analysis and Design)[19], SOUP(Service Oriented Unified Process)[17], 그리고 SOMA(Service Oriented Modeling and Architecture)[12] 등이 있다. 그러나 이러한 SOA 개발 방법론들은 서비스를 정의하고 식별을 위해 기준을 명확하게 제시하고 있지 않고 있어 실제 시스템의 개발에 적용하는데 어려움이 있다. 또한, 기존의 서비스 식별 및 정의에 대한 관한 연구들은 서비스 식별 기법에만 초점을 맞추었고, 구체적인 절차나 객관적인 기준을 제시하지 못했다[1, 7, 10].

서비스 정의 단계는 SOA 수립 과정의 가장 핵심 단계이며, SOA를 통해 제공될 후보 서비스들을 이끌어내고 정의하는 과정이라고 볼 수 있다. 따라서 SOA 기반의 시스템이 효율적이고 성공적으로 자리 잡기 위해서는 올바른 서비스 구성이 필수적이며 이를 위해 구체적인 서비스의 정의 및 식별 방법에 관한 연구가 필요하다. 본 논문에서는 서비스 정의 및 식별 과정에서 적절한 서비스를 도출하기 위하여 서비스 온톨로지를 이용한 서비스 도출 방법론인 SOMO(Service Oriented Modeling using Ontology)를 제안한다.

SOMO는 서비스 온톨로지를 이용함으로써 서비스 식별에 적절한 기준을 제시하였으며, 서비스 도출 프로세스는 서비스를 도출하기 위한 구체적인 절차를 담고 있다. 따라서 서비스 도출에 있어서 서비스 식별을 좀 더 용이하게 해주고, 서비스 온톨로지를 이용함으로써 서비스의 재사용성과 유연성을 높여줄 수 있다. 따라서 본 논문은 서비스 온톨로지를 정의하고, 사용자 요구사항 분석을 기반으로 서비스 온톨로지를 이용하여 서비스를 식별 및

정의하는 단계로 이루어져 있다.

본 논문의 구성은 다음과 같다. 제 2장에서는 본 논문에서 서비스 식별 및 정의에 사용하기 위해 제안하는 서비스 온톨로지를 소개한다. 제 3장은 SOMO의 프로세스로 구성되며 요구사항 분석, 서비스 식별, 서비스 정의로 이루어진다. 제 4장에서는 SOMO를 평가하기 위하여 기존의 SOA 방법론과 비교 평가한다. 마지막으로 제 5장에서는 결론을 내린다.

2. 서비스 온톨로지

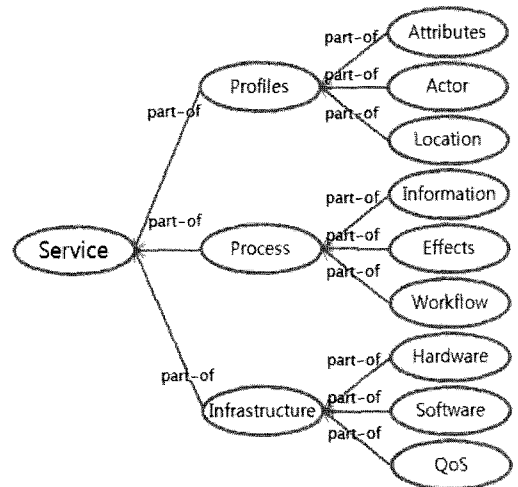
온톨로지에 대한 명확한 정의는 내려지지 않았지만 일반적으로 “개념화에 대한 명시적인 명세사항”이라는 정의를 사용한다[15]. 온톨로지의 특징으로는 어휘사이의 논리적 형식보다 내용 중심으로 창조한다는 점이다. 또한, 정보의 처리방식이 단순 패턴 매칭이 아닌 내용 중심적이기 때문에 정보의 통합과 공유가 가능하다는 점이 있다. 다음으로 사용자에게 맞게 특화되거나 전문 분야에 적합하게 세분화할 수 있다는 점이 있다. 마지막으로 온톨로지는 추론 기능을 제공한다는 특징이 있다.

본 논문에서는 서비스 정의와 식별을 위하여 서비스 도출 방법론에 온톨로지를 적용시킨다. 온톨로지는 클래스, 객체, 관계로 구성된다. 클래스는 개념이라 할 수 있으며 하위 클래스를 지닐 수 있고 객체를 포함하는 집합이다. 객체는 클래스의 인스턴스로 볼 수 있으며, 관계는 클래스들 간의 관계를 나타내며 is-a, has-a, part-of 등 클래스들 간에 존재하는 관계에 따라 여러 가지로 나타낼 수 있다.

본 논문에서 제안하는 서비스 온톨로지는 SOA에서의 서비스 정의와 특징을 바탕으로 정의하였다. 우선 서비스는 어떤 기능들, 즉 비즈니스 프로세스를 수행하는 기능들의 집합이라고 할 수 있다. 따라서 서비스는 서비스를 수행하기 위한 기능들의 정보와 서비스를 수행 했을 때의 효과를 나타낼 수 있어야 한다. 또한 서비스는 서비스 공급자,

서비스 요청자, 서비스 저장소라는 세 요소가 유기적으로 연계 되어 있는 특징을 갖는다. 그러므로 서비스는 서비스를 행하는 주체들과 저장되어 있는 서비스의 위치를 알 수 있는 정보가 필요하다. 뿐만 아니라, 서비스는 액터 및 시스템의 다양한 요구 사항을 만족할 수 있도록 다양한 레벨 단위의 서비스를 제공하기 위해 서비스의 단위를 알 수 있는 정보도 필요하다.

따라서 서비스 온톨로지는 [그림 1]과 같이 프로파일(Profiles), 프로세스(Process), 인프라스트럭처(Infrastructure)로 구성된다. 또한, 각각의 요소들은 클래스로써 다른 클래스로 나누어지게 된다. 따라서 각 클래스가 다른 클래스의 일부분이 되기 때문에 관계는 part-of로 정의하였다. 각 클래스에는 클래스의 인스턴스인 객체가 들어가며 본 단계에서는 객체의 상위 개념인 클래스 단위에서의 설명만 하며, 각각의 클래스에 대한 설명은 다음과 같다.



[그림 1] 서비스 온톨로지

2.1 프로파일

프로파일은 서비스의 기본적인 정보를 담고 있으며 애트리뷰트, 액터, 로케이션으로 구성된다. 애트리뷰트는 서비스의 기본 속성들을 나타내며 서비스의 ID, 이름, 타입 등의 정보가 들어간다. 액터는

서비스를 행하는 주체들의 정보를 담고 있으며 서비스를 제공하는 서비스 제공자와 서비스를 소비하는 서비스 소비자 등의 정보가 들어간다. 로케이션은 서비스가 위치하는 위치 정보를 나타낸다.

2.2 프로세스

프로세스는 서비스의 정보와 효과 및 흐름을 담고 있으며 인포메이션, 이펙트, 워크플로우로 구성된다. 인포메이션은 서비스가 행하기 위해 필요한 정보, 즉 입력정보와 행하고 난 뒤의 정보인 출력 정보를 담고 있다. 이펙트는 서비스의 효과를 나타내며 워크플로우는 서비스를 이루는 액티비티들의 순서와 정보를 나타내고, 서비스의 실행과 종료 를 위한 조건인 사전조건과 후행조건의 정보를 표현한다.

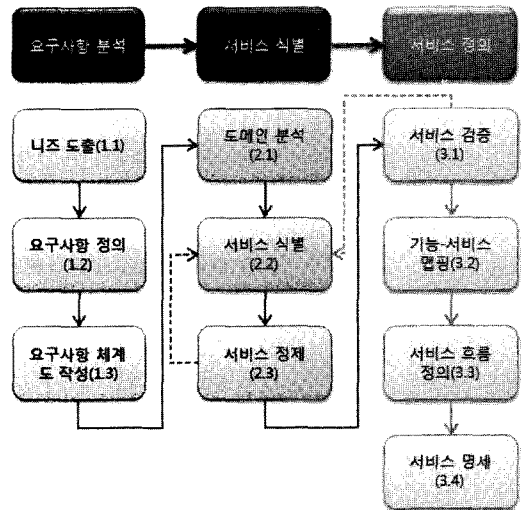
2.3 인프라스트럭처

인프라스트럭처는 서비스가 이루어지는 기반구조를 나타내며 하드웨어, 소프트웨어, QoS로 구성된다. 하드웨어는 서버, 입출력 장치, 저장소, 네트워크 등의 정보를 담고 소프트웨어는 플랫폼과 프로토콜 등의 정보로 이루어진다. 마지막으로 QoS는 서비스의 질을 나타내는 정보를 담고 있으며 서비스의 확장성, 용이성 등이 이에 속한다.

3. 서비스 도출 프로세스

본 논문에서 제안하는 서비스 도출 방법론인 SOMO의 프로세스는 [그림 2]와 같이 크게 요구사항 분석, 서비스 식별, 서비스 정의의 3단계로 이루어지며 각각의 프로세스는 세부 프로세스로 나누어진다. 요구사항 분석은 니즈를 도출하고, 니즈로부터 요구사항을 정의하여 요구사항 체계도를 작성하는 단계로 이루어져 있다. 서비스 식별은 서비스를 식별하기 위하여 도메인 분석을 수행하고, 도메인 분석 결과로부터 서비스를 식별하여 정제

하는 과정까지로 이루어진다. 서비스 정의는 식별된 서비스를 검증하고, 검증된 서비스들을 기능과 서비스의 맵핑과 서비스의 흐름을 정의하여 최종적으로 서비스를 명세하는 과정이다.

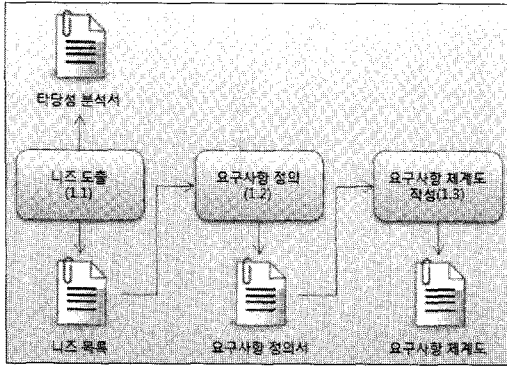


[그림 2] 서비스 도출 프로세스

3.1 요구사항 분석

사용자 요구사항을 만족하는 것은 모든 시스템에서 가장 기본이 되는 핵심적인 사항이다. 따라서 SOMO에서는 사용자 요구사항에 맞추어 서비스를 도출하기 위하여 [그림 3]에서와 같이 사용자 요구사항을 가장 먼저 분석한다. 요구사항 분석 단계에서는 이해당사자를 선정하고 이해당사자의 니즈로부터 타당성 분석을 통해 니즈를 정제하여 최종 니즈를 도출한다. 정제된 니즈로부터 요구사항을 도출하기 위하여 QFD(Quality Function Deployment, 품질기능 전개)[11] 분석기법을 이용한다. QFD 분석은 니즈와 기능과의 맵핑을 통해 요구사항을 만족시킬 수 있는 기능들을 찾아내어 요구사항으로 도출한다. 도출한 요구사항들은 계층적으로 상세히 나타내는 RBS(Requirement Break-down Structure)[13]를 작성한다. 이 단계에서 작성된 요구사항 체계도는 서비스를 식별하기 위해 수

행하는 도메인 분석 단계에서 사용된다.



[그림 3] 요구사항 분석 프로세스

3.1.1 니즈 도출(1.1)

니즈 도출 단계는 이해당사자들의 니즈로부터 타당성 분석을 통해 니즈를 정제하여 최종 니즈를 도출하는 과정이다. 우선, 시스템을 사용하기 위한 이해당사자들을 선정하고 이해당사자들로부터 니즈를 도출한다. 이 단계에서 나온 니즈는 정제되지 않은 니즈로 타당성 분석을 통해 니즈를 정제해야 한다. 타당성 분석을 통해 나온 정제된 니즈는 최종 니즈로 선택하게 된다.

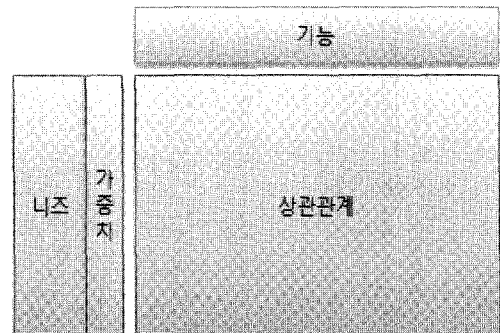
타당성 분석은 기술, 경제, 운용, 정책의 4가지 분야에 대해 수행한다. 기술적 타당성 분석은 니즈의 기술적인 측면에 있어서 트렌드 부합성, 정보기술 인프라 등 적용기술의 적합성과 사업수행 준비도, 위험 분석 등의 기술실현의 가능성 등에 대한 분석을 수행한다. 경제적 타당성 분석은 니즈의 경제적인 측면에서의 타당성을 분석하는 것으로 정성적 경제 타당성과 정량적 경제 타당성으로 나누어 분석한다. 운용 타당성 분석은 운용적인 측면에서 효과성, 적합성 등에 대한 분석을 수행한다. 마지막으로 정책적 타당성 분석은 니즈의 정책적인 부분에 있어서 필요성, 시급성 등에 대한 타당성 분석을 수행한다.

타당성 분석을 위해서는 기술, 경제, 운용, 정책 타당성 분석으로부터 도출된 각각의 결과를 종합하여 판단해야 한다. 이를 종합하기 위해서는 각

각의 기준, 특성, 또는 목표에 따라 중요도를 합리적으로 결정하여야 하며 본 논문에서는 이러한 종합적인 결론을 내리기 위하여 AHP(Analytic Hierarchy Process, 계층화분석법)[4] 분석기법을 이용한다. AHP 분석기법을 이용하여 각 기준들의 가중치를 매겨 분석한다. 타당성 분석의 세부 기준과 가중치는 업무의 특성에 맞게 정해야 하며, 타당성 분석의 기준 점수 또한 업무의 특성에 맞게 정해야 한다. 타당성 분석 결과 타당하다고 판단되어진 니즈를 최종 니즈로 도출하며 이는 요구사항 분석 단계에서 요구사항을 도출하기 위해 사용된다.

3.1.2 요구사항 분석(1.2)

요구사항 분석 단계는 타당성 분석 후 정제된 니즈들로부터 요구사항을 도출해내는 과정이다. 니즈로부터 요구사항을 도출하기 위하여 QFD 분석기법을 이용한다. QFD 분석기법은 [그림 4]와 같이 니즈와 기능과의 상관관계를 나타내는 매트릭스로 나타낸다. 또한, 니즈의 가중치는 앞서 분석한 타당성 분석 결과 값을 가중치로 정한다. QFD 분석기법에서 니즈와 기능의 상관관계는 니즈와 기능의 관련도를 나타내며, 1, 3, 5점 척도나 1, 3, 9점 등의 척도로 나타낼 수 있다. QFD 분석기법을 이용하여 니즈를 만족시킬 수 있는 기능들을 찾아내어 기능을 요구사항으로 도출한다. 요구사항 도출 시에는 만족시킬 수 있는 니즈의 수와 니즈들의 가중치, 그리고 니즈의 합계 등을 고려해야 한다.



[그림 4] 니즈-기능 QFD 분석

[그림 5]는 니즈와 기능과의 QFD 예를 보여주고 있다. 니즈의 가중치는 앞서 분석한 타당성 분석 결과 값이다. 니즈와 기능과의 상관관계는 1, 3, 5점 척도로 나타내고 있으며, 가로축의 니즈의 합계는 니즈를 만족시킬 수 있는 기능들의 합으로 높은 값을 나타내면 니즈를 만족시킬 수 있는 기능이 많다는 것을 의미하고, 낮은 값을 나타내면 니즈를 만족시킬 수 있는 기능이 적다는 것을 의미한다. 세로축의 합계는 기능이 만족시킬 수 있는 니즈들의 합으로 높은 값을 나타내면 기능이 만족시킬 수 있는 니즈가 많다는 것을 의미하고, 적은 값을 나타내면 만족시킬 수 있는 니즈가 적

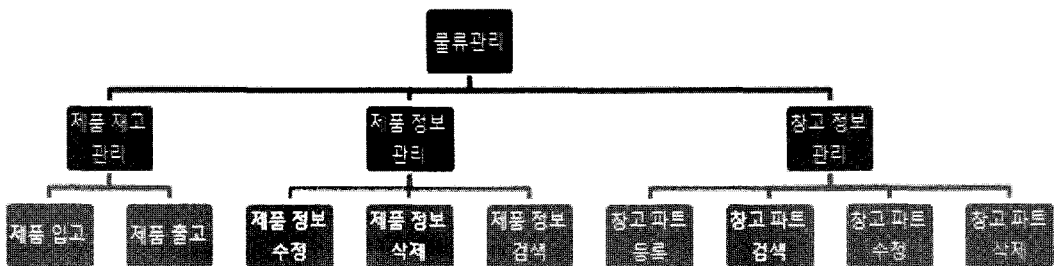
다는 것을 의미한다.

3.1.3 요구사항 체계도 작성(1.3)

요구사항 체계도 작성 단계에서는 요구사항 정의 단계에서 정의한 요구사항들을 상세하고 계층적으로 나타낼 수 있는 RBS를 작성한다. RBS는 요구사항들을 그룹화하고, 상세히 나타낸다. 이 단계에서 나오는 RBS는 서비스를 식별하기 위한 도메인을 분석하는 도메인 분석 단계에서 To-Be 모형이 되기 위한 바탕이 된다. [그림 6]은 RBS의 예를 보여주고 있다. 물류관리는 제품 재고 관리, 제품 정보 관리, 창고 정보 관리 세 개의 요구사항으

니즈	가중치	기능						합계
		시스템 로그인	제품 입고	제품 출고	제품 등록	제품 검색	제품 수정	
전체제품에서 현재 입고된 제품 검색	4.4	3	3			5		11
입고일 별 정보 검색 및 수정	4.1	3	3			5		11
검색된 정보 수정	4.2	3					5	8
수정 한 직원의 이름 등록	3.2	3					3	6
전체제품에서 현재 출고할 제품 검색	3.9	3		3		5		11
출고일 별 정보 검색 및 수정	4	3		3		5	5	16
모든 제품 검색	4.8	3				5		8
정보 등록, 수정, 삭제	4.3	3			5	3	5	21
신제품 정보를 받아서 물류시스템에 등록	4.1	3			5			8
물류 관리 시스템에 등록이 되어 있지 않은 제품은 입고 업무 진행 불가	4.5	3	5					8
합계		30	11	6	10	28	18	5

[그림 5] 니즈-기능 QFD 예

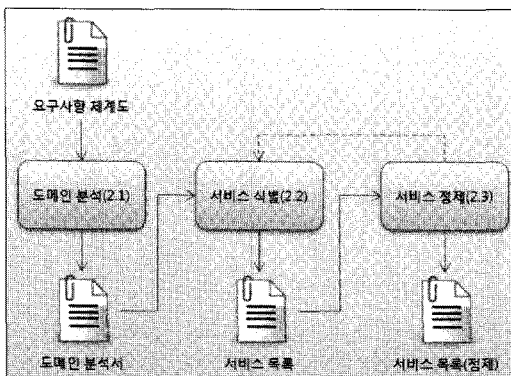


[그림 6] RBS 예

로 구분되며 각각은 또한 세부 요구사항으로 구분될 수 있다.

3.2 서비스 식별

SOA 구현 시 가장 중요한 것은 서비스를 식별하는 것이다. 따라서 서비스 식별 단계는 SOA 구현의 가장 핵심적인 단계라고 할 수 있다. 이 단계는 SOA 구현의 가장 핵심적인 단계인 서비스 식별 단계이므로 전체 프로세스 중 가장 중요한 단계라고 할 수 있다. 서비스 식별 단계는 [그림 7]에서 볼 수 있듯이 도메인 분석 단계와 서비스 식별, 서비스 정제로 구성되어 있다. 도메인 분석 단계에서는 서비스가 될 수 있는 서비스의 도메인을 분석하는 것으로 이 단계에서는 프로세스 분류체계를 정의한 후 FFBD(Functional flow block diagram, 기능흐름분석)[13]를 작성하게 된다. 서비스 식별 단계에서는 서비스를 식별하기 위한 단위와 서비스 식별 기준을 정의한 후 앞서 분석한 서비스 도메인으로부터 서비스를 1:1 매핑 관계, 결합 관계, 분할 관계로 식별한다. 식별한 서비스는 체계적으로 분류하고, 서비스 식별 단계를 반복하는 과정을 통해 서비스를 정제한다.



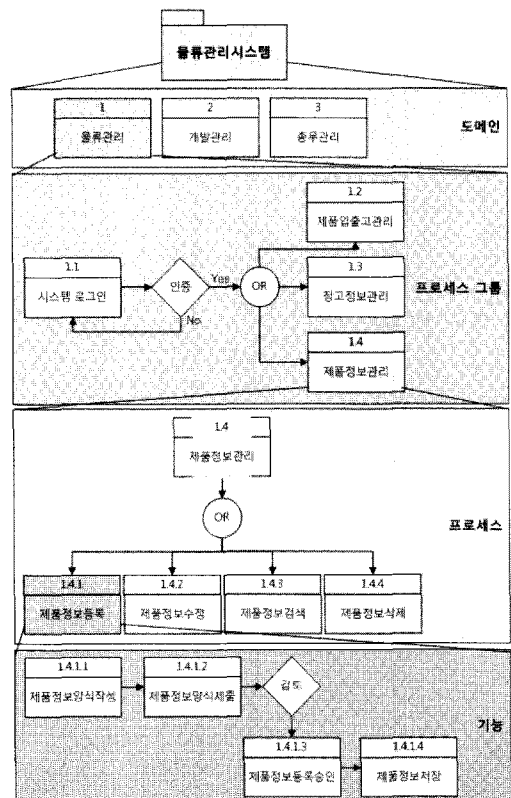
[그림 7] 서비스 식별 프로세스

3.2.1 도메인 분석(2.1)

도메인 분석 단계는 서비스가 될 수 있는 서비스의 도메인을 분석하는 단계이다. 도메인 분석을 위

해서는 먼저 프로세스 분류 체계를 정의한 후 앞서 작성한 RBS를 기반으로 기능 흐름을 분석한다. 기능 흐름 분석은 FFBD를 앞서 정의한 분류 체계를 기준으로 작성한다. 프로세스 분류 체계는 프로세스를 수행하기 위하여 프로세스 관점으로 업무를 체계화할 수 있도록 프로세스 레벨링 기준을 수립하는 것이다. 이 기준은 기업이나 기관의 환경과 특성에 맞추어 정의한다. 일반적으로 프로세스 분류 체계는 1레벨부터 5레벨까지 정의한다[5].

프로세스 분류체계를 정의한 후 기능 흐름 분석을 수행한다. 기능 흐름 분석은 요구사항 분석 단계에서 정의한 RBS를 반영한 To-Be 모형을 작성하며, 기능의 계층적인 분해도와 기능이 실행되는 차례를 서술하는 FFBD를 작성한다. FFBD는 기능과 프로세스를 그룹화 하여 계층적으로 나타내게 되며 기능적인 측면에서 작성하게 된다.



[그림 8] FFBD의 예

[그림 8]은 FFBD의 예를 보여주고 있다. 예제에서는 프로세스 분류체계를 프로세스 도메인-프로세스 그룹-프로세스-액티비티로 정의하였고, FFBD는 프로세스 분류체계에 맞추어 작성하였다. 따라서 물류관리시스템은 세 개의 도메인으로 나뉘며 각각의 도메인은 프로세스 그룹, 프로세스, 기능으로 나뉘게 된다.

3.2.2 서비스 식별(2.2)

서비스 식별 단계에서는 분석한 서비스 도메인을 바탕으로 서비스가 될 수 있는 서비스를 식별한다. 이 단계에서 식별하는 서비스는 정제되지 않았기 때문에 최종 서비스라 볼 수 없고, 서비스 후보로 볼 수 있다. 서비스 식별을 위해서는 우선적으로 정의해야 할 사항이 두 가지가 있다.

첫 번째는 서비스를 식별할 단위를 정하는 것이다. 도메인 분석 단계에서 정의한 프로세스 분류체계에서 어느 단위를 서비스로 볼 것인지를 정해야 한다. 서비스는 하나의 단일 서비스일수도 있

지만, 여러 개의 서비스가 모여져서 하나의 서비스가 될 수도 있다. 따라서 기업이나 기관에서의 업무 특성에 따라 어느 단위에서 서비스를 식별할 것인지를 결정한다. 대부분의 경우 하나의 프로세스이거나 여러 개의 프로세스가 모여서 서비스가 되기 때문에 프로세스 단위에서 서비스를 식별하는 것이 일반적이다. 따라서 본 논문에서는 프로세스를 서비스를 식별하기 위한 기준 단위로 본다.

두 번째는 서비스를 식별할 때의 기준이다. 본 논문에서는 서비스를 식별하는데 사용하기 위하여 서비스 온톨로지를 제안하였으며 식별 기준은 서비스 온톨로지의 속성인 프로파일, 프로세스, 인프라스트럭처로 정의한다. 프로세스가 정의한 기준의 속성을 충족시킬 수 있을 경우 체크를 하게 되고, 기준이 모두 체크되었을 경우 서비스로 식별하게 된다.

프로세스로부터 서비스를 식별하는 경우는 총 3가지이다. 첫 번째는 프로세스와 1:1 매핑으로 식별되는 경우로 하나의 프로세스와 하나의 서비

번호	이름	프로파일	프로세스	인프라스트럭처	관계	서비스
1.2.1	제품입고정보등록	✓	✓	✓	1:1	제품입고정보등록 서비스
1.2.2	제품입고정보수정	✓	✓	✓	1:1	제품입고정보수정 서비스
1.2.3	제품입고정보삭제	✓	✓	✓	1:1	제품입고정보삭제 서비스
1.2.4	제품입고정보조회	✓	✓	✓	1:1	제품입고정보조회 서비스
1.2.5	제품출고정보등록	✓	✓	✓	1:1	제품출고정보등록 서비스
1.2.6	제품출고정보수정	✓	✓	✓	1:1	제품출고정보수정 서비스
1.2.7	제품출고정보삭제	✓	✓	✓	1:1	제품출고정보삭제 서비스
1.2.8	제품출고정보조회	✓	✓	✓	1:1	제품출고정보조회 서비스
1.3.1	창고정보등록	✓	✓	✓	1:1	창고정보등록 서비스
1.3.2	창고정보수정	✓	✓	✓	1:1	창고정보수정 서비스
1.3.3	창고정보검색	✓	✓	✓	1:1	창고정보검색 서비스
1.3.4	창고정보삭제	✓	✓	✓	1:1	창고정보삭제 서비스
1.3.5	창고정보조회	✓	✓	✓	1:1	창고정보조회 서비스
1.4.1	제품정보등록	✓	✓	✓	1:1	제품정보등록 서비스
1.4.2	제품정보수정	✓	✓	✓	1:1	제품정보수정 서비스
1.4.3	제품정보삭제	✓	✓	✓	1:1	제품정보삭제 서비스
1.4.4	제품정보검색	✓	✓	✓	1:1	제품정보검색 서비스

[그림 9] 서비스 식별의 예

스가 연결 되는 경우이다. 이 경우에는 프로세스가 독립적인 업무의 특성을 갖는다. 두 번째는 결합의 경우로 여러 개의 프로세스가 하나의 서비스로 정의되는 경우이다. 이 경우 하나의 프로세스가 너무 작게 나누어져 있거나 여러 개의 프로세스가 모여서 하나의 업무를 수행할 경우에 해당된다. 마지막은 분할의 경우로 하나의 프로세스에서 여러 개의 서비스가 정의되는 경우이다. 이 경우는 하나의 프로세스가 너무 크게 정의되어 있거나, 여러 개의 서비스로 정의함으로써 재사용성을 높일 경우 발생한다.

[그림 9]는 서비스 식별을 하기 위한 예를 보여주고 있으며, 식별 기준은 서비스 온톨로지의 속성인 프로파일, 프로세스, 인프라스트럭처로 정의하였다. 프로세스가 정의한 기준의 속성을 충족시킬 수 있을 경우 체크를 하게 되고, 기준이 모두 체크되었을 경우 서비스로 식별하게 된다. 또한, 프로세스로부터 서비스로 식별할 경우의 관계를 함께 써주며 모든 관계가 1:1 관계인 것을 볼 수 있다.

3.2.3 서비스 정제(2.3)

서비스 정제 단계는 식별한 서비스들을 정제하는 과정이다. 먼저, 식별한 서비스들 중 다른 프로세스이거나 이름은 다르지만 서비스 온톨로지의 나머지 속성들이 같은 경우에는 동일한 서비스로 볼 수 있는 경우를 찾는다. [그림 10]은 다른 프로세스에 있지만 서비스 온톨로지의 속성이 동일하여 하나의 서비스인 로그인 서비스로 식별한 경우를 나타낸 예이다.

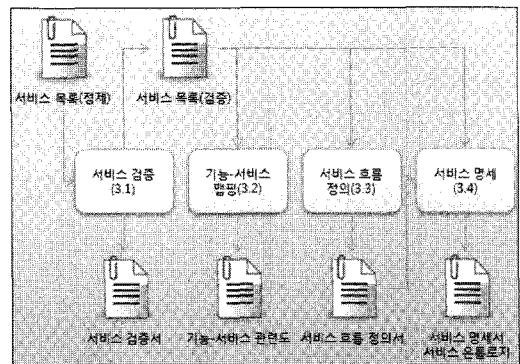
번호	이름	프로파일	프로세스	인프라스트럭처	서비스
1.1	시스템 로그인	동일	동일	동일	로그인 서비스
2.1	시스템 로그인				
3.1	시스템 로그인				

[그림 10] 서비스 정제의 예

다음으로 서비스는 액터 및 시스템의 다양한 요구 사항을 만족하기 위하여 다양한 레벨 단위의 서비스를 제공하게 된다. 이를 위하여 식별한 서비스들을 체계적으로 분류하며 레벨 단위는 업무 특성에 맞추어 정하게 된다.

3.3 서비스 정의

서비스 정의 단계는 식별되고 정제된 서비스에 대해서 검증과 흐름 정의를 통해서 서비스를 명세하는 단계이다. [그림 11]과 같이 서비스 정의 단계는 서비스 검증, 기능-서비스 맵핑, 서비스 흐름 정의, 서비스 명세 단계로 구성되며 앞서 분석한 서비스 식별 단계에서 나온 산출물인 서비스 목록을 사용하게 된다. 서비스 검증 단계에서는 식별한 서비스들을 검증하여 최종적으로 서비스를 결정하게 된다. 검증 작업이 완료 되면 최종적인 서비스 목록이 나오게 되고, 이 서비스 목록을 가지고 기능과 서비스의 맵핑 작업을 통해 기능을 수행하기 위한 서비스들을 판단하고, 서비스 흐름을 정의한다. 서비스 흐름 정의까지 완료 한 후에는 서비스를 명세하게 된다. 서비스 명세는 서비스 명세서와 앞서 정의한 서비스 온톨로지 요소에 맞추어 서비스 온톨로지로 작성하게 된다.



[그림 11] 서비스 정의의 프로세스

3.3.1 서비스 검증(3.1)

서비스 검증 단계에서는 식별된 서비스를 평가

하여 서비스를 최종 결정하는 검증의 과정을 거친다. 서비스 검증을 위해서는 사용자 요구사항을 만족하는지에 대해 분석하고, 식별된 서비스들을 평가 기준에 맞춰 평가하게 된다. 앞서 분석한 사용자 요구사항을 모두 만족하는지에 대해 평가를 위해서는 요구사항과 서비스를 맵핑시켜야 한다. 맵핑 작업 후 만족하지 않는 사용자 요구사항이 있을 경우에는 사용자 요구사항을 만족시키는 서비스를 추가시키는 작업을 고려해야 한다.

[그림 12]는 사용자 요구사항과 서비스와 맵핑시킨 예를 보여주고 있다. 예제에서는 4개의 요구사항이 각각의 서비스와 맵핑되어 모두 만족하는 것으로 보이며, 이 경우에는 다른 서비스를 추가시키는 작업을 고려할 필요가 없게 된다.

다음으로 식별된 서비스들을 검증할 서비스 평가를 수행한다. 서비스 평가를 위해서는 우선 평가 기준을 정해야 한다. 평가 기준은 SOA를 하고자 하는 목적과 업무 특성에 맞게 정하고, 그에 따라 가중치를 매겨 평가한다. 예를 들어, [9]에서 제시한 평가 기준은 총 8가지로 비즈니스 목적 달성도, 재사용성, 독립성, 서비스 중복성, 제공성, 서비스 노출 여부, 서비스 관리 필요성, 분산성이다. 이 평가 기준과 기준 점수는 SOA 목적과 업무 특성에 맞게 정한다. 마지막으로, AHP 분석기법 등을 이용하여 평가 기준들의 가중치를 정하고, 서비스 모델러 및 시스템 담당자들과 함께 평가하게 된다. 평가 후 기준 점수 이상을 받은 서비스는 최종 서비스로 정의되며 기준 점수 이상을 받지 못한 서비스들은 다시 한 번 검토하여 최종 분류한다.

3.3.2 기능-서비스 맵핑(3.2)

기능-서비스 맵핑 단계에서는 기능들과 서비스들과의 맵핑을 통해 기능에 무슨 서비스가 대응하는지를 식별하게 된다. 이 단계를 거치면 기능을 수행하기 위한 서비스가 무엇인지를 판별해낼 수 있고, 기능을 수행하는데 필요한 서비스들의 우선순위를 판별해낼 수 있다.

기능에 대응하는 서비스를 식별하기 위해서는 QFD 분석기법을 통하여 기능과 서비스를 관련도에 따라 점수를 매겨 맵핑시킨다. QFD 분석은 서비스와 기능과의 상관관계를 나타낸다. QFD 분석을 통해 기능에 대응하는 서비스들의 관련도를 분석하여 기능을 수행하기 위한 서비스가 충분한지를 판별해내고 충분하지 않을 경우에 서비스 정제 과정을 통해 기능을 수행하기 위한 서비스를 추가하는 것을 고려해야 한다. 기능에 대응하는 서비스를 식별하기 위하여 기능과 서비스의 관련도를 QFD를 이용하여 점수로 나타내는 것은 기능을 수행하는 데 어떤 기능이 더 중요한지를 판별해내기 위함이다. 이를 통해서 기능을 수행하는 데 필요한 서비스들의 우선순위를 판별해낼 수 있게 된다.

3.3.3 서비스 흐름 정의(3.3)

서비스 흐름 정의 단계에서는 기능을 수행하는데 필요한 서비스의 흐름을 정의한다. 이 단계에서 정의되는 서비스 흐름은 시스템의 비즈니스 프로세스라고 볼 수 있으며, 이 단계를 통해 서비스의 전후관계 등의 특성을 파악할 수 있다. [그림 13]은 서비스 흐름 정의의 예를 보여주고 있다. 서

서비스 요구사항	제품입고신청 서비스	제품입고정보 등록 서비스	제품출고신청 서비스	제품출고정보 등록 서비스	제품정보수정 서비스	제품정보삭제 서비스
제품 입고	√	√				
제품 출고			√	√		
제품 정보 수정					√	
제품 정보 삭제						√

[그림 12] 사용자 요구사항-서비스 맵핑의 예

비스 흐름을 정의하기 위하여 기능을 수행하는 데 필요한 액터, 사전조건, 사후조건, 요구사항을 함께 정의하고 있다. 이는 서비스 흐름을 정의하기 위한 기본 정보이다. 이러한 서비스 흐름 정의를 통해서 제품입고관리를 위해서는 총무관리와 물류관리라는 액터가 필요하며, 제품입고관리를 수행하는 데 필요한 서비스들의 흐름을 알 수 있다. 또한, 로그인 서비스 후에 제품입고신청 서비스로 넘어가야 한다는 서비스의 전후관계 정보 또한 파악할 수 있게 된다.

3.3.4 서비스 명세(3.4)

서비스 흐름 정의와 서비스와 기능과의 맵핑 작업을 수행하면 서비스의 특성과 서비스의 기능들을 식별하게 된다. 서비스 명세 단계에서는 서비스와 서비스의 기능을 최종 결정하고 이를 명세화한다. 서비스 명세 작업의 산출물은 일반 문서로 명세하는 서비스 명세서와 미리 정의한 서비스 온톨로지 요소에 온톨로지로 작성하는 서비스 온톨로지로 구성된다. 이러한 작업은 서비스 식별부터 서비스 흐름 정의까지 서비스들의 반복적인 비교와 검증은 통해 정제하며 이루어진다. 온톨로지는 [그림 14]와 같이 서비스 속성들 뿐 아니라 서비스

들의 관계도 정의하게 된다. 서비스를 서비스 온톨로지로 작성함으로써 서비스의 용이성과 재사용성이 높아지게 된다.

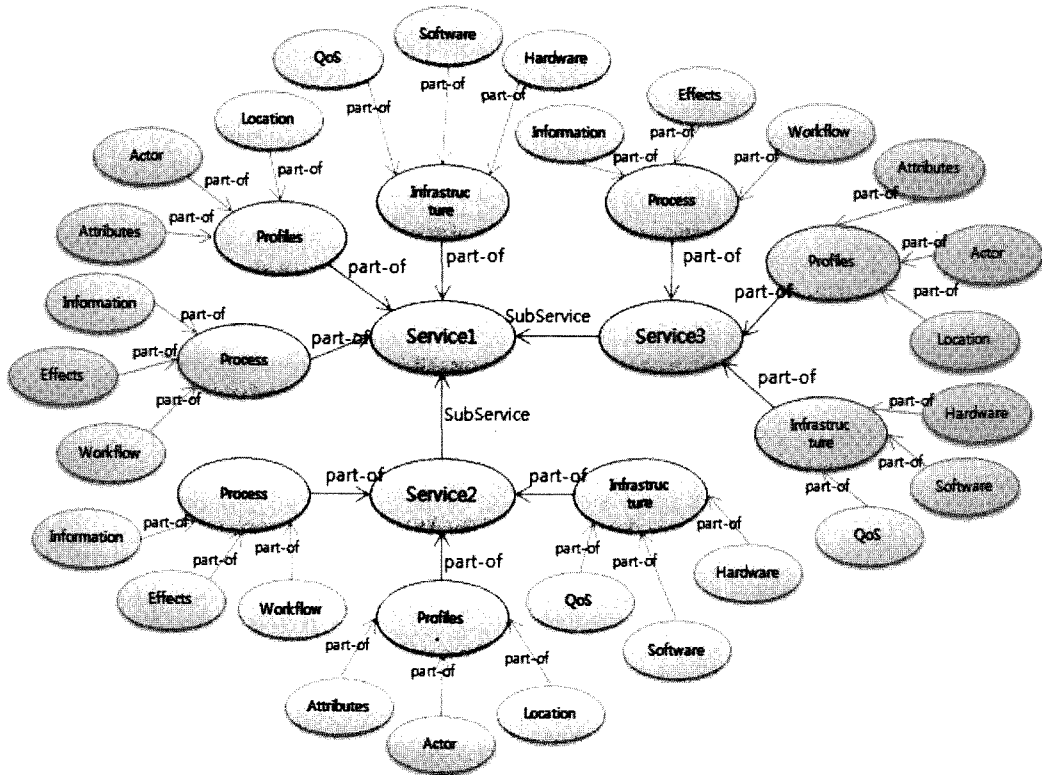
4. 평가

본 논문에서는 SOA를 위하여 온톨로지 기반의 서비스 도출 방법론인 SOMO를 제안하였다. 기존에 서비스 식별에 관한 연구가 이루어졌지만 대부분 연구 범위가 서비스 식별 한 부분에만 초점을 맞추고 있다. SOMO 또한 SOA 개발 방법론의 일부이지만 연구 범위가 구현 단계만 뿐 요구사항 분석 단계부터 서비스 정의 및 식별까지 이루어지므로 기존의 SOA 개발 방법론과 비교하여 평가하고자 한다.

우선, SOMO를 평가하기 위한 평가 기준을 정하기 위하여 기존의 연구들을 살펴보았다. [3]은 요구사항 분석 단계에서 유도되는 유즈케이스 모델을 기반으로 하는 웹 서비스 개발 절차 및 가이드라인을 포함하는 웹 서비스 개발 프로세스를 제안하였다. 제안하는 개발 프로세스와 기존의 SOA 방법론과의 비교를 위하여 5가지인 참조, 모델링 기술, 산출물, 프로세스, 서비스평가의 항목으로

기능	제품입고관리
액터	총무관리부, 물류관리부
개요 및 설명	제품 입고 신청이 들어오면 시스템에서 입고된 제품 정보를 검색하여 제품의 입고수량을 입력하여 수량을 증가시킨다.
사전조건	- 총무관리, 물류관리 권한으로 시스템에 로그인하여야 한다. - 입고될 제품 정보가 시스템에 등록되어 있어야 한다.
사후조건	- 현재 제품 수량에 입고된 제품의 수량을 누적시킨다.
서비스 흐름	<pre> graph LR A[총무관리] --> B[로그인 서비스] B --> C[제품입고신청 서비스] C --> D[물류관리] D --> E[로그인 서비스] E --> F[재충전보급 서비스] F --> G[제품입고등록 서비스] </pre>
특수 요구사항	- 물류 관리는 제품의 수량을 입력, 수정, 삭제할 수 있는 권한이다.

[그림 13] 서비스 흐름 정의의 예



[그림 14] 서비스 온톨로지

비교하였다.

[8]은 SOA 시스템 구축을 위한 관점지향 서비스 식별 기법을 제안하였다. 제안하는 서비스 식별 기법을 평가하기 위하여 기존 서비스 식별 기법과 비교 평가하였고, 비교 평가 기준은 접근 방식, 연구 범위, 산출물, 각 단계 산출물 연관 관계, 비 기능적 요구사항 명시 여부, 시스템 구조 변경 시 용이성으로 6가지이다.

[2]는 서비스 공급자와 서비스 요청자의 관점을 반영하여, SOA를 위한 서비스 지향 개발 프로세스를 제안하였다. 제안한 프로세스의 평가를 위해서는 기존의 객체지향방법 및 CBD와 비교 분석과 기존의 SOA 방법론들과의 비교 분석을 행했다. 기존의 SOA 방법론과의 비교 분석의 비교 기준은 모델링기술, 산출물, 프로세스 지원, 서비스 식별 방법, 서비스 구현 전략 결정 과정이다.

[6]은 SOA를 구현하기 위한 개발 방법론과 그 과정에서 사용할 수 있는 구체적인 절차들을 정의하였다. 제안한 방법론을 평가하기 위하여 기존에 이루어져 왔던 객체지향 개발 방법론과 CBD 방법론과의 비교 분석과 기존의 SOA 방법론과의 비교 분석을 행했다. 기존의 SOA 방법론과의 비교 분석은 제안자, 영향 받은 기술, 모델링 기술, 산출물, 라이프사이클 지원 정도, 서비스 식별, 서비스 구현 전략 결정의 7가지 기준을 가지고 비교하였다.

<표 1>은 SOA 개발 방법론이나 서비스 식별 방법론에 관련한 기존의 연구들과 제안하는 방법론을 비교하기 위하여 정한 비교 기준들을 정리한 표이다.

기존의 연구들에서는 제안하는 SOA 방법론을 기존의 SOA 방법론과의 몇 가지 기준을 가지고 정성적으로 비교 분석하였다. 하지만, 객관적으로

〈표 1〉 기존의 연구와 평가 기준들

기존 연구	비교 기준
[3]	참조, 모델링 기술, 산출물, 프로세스, 서비스 평가
[8]	접근 방식, 연구 범위, 산출물, 각 단계 산출물 연관 관계, 비 기능적 요구사항 명시 여부, 시스템 구조 변경 시 용이성
[2]	모델링기술, 산출물, 프로세스 지원, 서비스 식별 방법, 서비스 구현 전략 결정 과정
[6]	제안자, 영향 받은 기술, 모델링 기술, 산출물, 라이프사이클 지원 정도, 서비스 식별, 서비스 구현 전략 결정

정확한 평가를 위해서는 정량적인 평가가 이루어져야 한다. 따라서 본 논문에서는 기존의 연구들의 비교 기준을 참고하여 평가 기준을 정성적인 평가를 위한 비교 기준과 정량적인 평가를 위한 평가 기준으로 나누었다. <표 2>는 정성적인 비교 기준과 정량적인 평가 기준과 내용들을 정리한 표이다.

우선, 정성적인 평가를 위한 비교 기준은 총 5가지로 기존의 방법론의 제안자, 참조기술, 모델링 기술, 접근 방식, 산출물이다. 제안자는 방법론을 제안한 단체, 혹은 개인이고 참조기술은 방법론을 제안하는데 영향 받은 기술이다. 또한, 모델링 기술은 방법론에서 사용한 모델링 기술을 기술하며,

접근 방식은 서비스에 접근하는 방식이 상향식인지, 하향식인지 또는 양방향식인지를 기술한다. 마지막으로 산출물은 방법론에서 산출물에 대한 정의를 언급했는지를 기술한다.

다음으로 정량적인 평가를 위한 평가 기준은 총 5가지로 프로세스 상세성, 서비스 식별 상세성, 유연성, 재사용성, 사용자 요구사항 반영도이다. 프로세스 상세성은 방법론의 프로세스들이 얼마나 상세하게 정의되었는지를 평가하고, 서비스 식별 상세성은 서비스 식별 방법에 대해 얼마나 상세하게 정의 되었는지를 평가한다. 또한, 유연성은 시스템 구조 변경 시 용이성을 평가하며 재사용성은 기 구축 모듈에 대한 활용도를 평가한다. 마지막으로 사용자 요구사항 반영도는 사용자 요구사항의 반영 정도를 평가한다.

<표 3>은 정의한 비교 기준에 따라 기존의 SOA 방법론과 SOMO를 비교한 표이다. 참조 기술을 보면 SODA와 SOMO는 CBD와 SOA를 참조하여 정의하였으며, SOAD는 OOAD, EA, BPM를 참조하였다. SOMA는 기존의 SOAD를 참조하였으며 SOUP는 두 가지 기술인 RUP와 XP의 특징을 살려 정의하였다. 모델링 기술은 SOAD와 SOMO를 제외하고는 특별히 언급하지 않았으며, SOAD는 UML 기술을 사용하였고, SOMO는 온톨로지를 사용하였다. 접근 방식은 기존의 SOA 방법론들은 대개 상

〈표 2〉 평가 기준 내용

	기준	내용
비교 기준	제안자	방법론을 제안한 단체나 개인을 기술
	참조 기술	방법론을 제안하는데 영향 받은 기술
	모델링 기술	방법론에서 사용한 모델링 기술
	접근 방식	서비스에 접근하는 방식
	산출물	산출물에 대한 정의를 언급하였는지의 여부
평가 기준	프로세스 상세성	방법론의 프로세스들이 얼마나 상세하게 정의되었는지 평가
	서비스 식별 상세성	서비스 식별 방법에 대해 얼마나 상세하게 정의되었는지 평가
	유연성	시스템 구조 변경 시 용이성을 평가
	재사용성	기 구축 모듈에 대한 활용도를 평가
	사용자 요구사항 반영도	사용자 요구사항의 반영 정도를 평가

〈표 3〉 기존의 SOA 방법론과 SOMO와의 비교

	SODA	SOAD	SOMA	SOUP	SOMO
제안자	Gartner	IBM	IBM	Knual Mittal	CAUIS
참조 기술	CBD, SOA	OOAD, EA, BPM	SOA	RUP, XP	CBD, SOA
모델링 기술	언급 없음	UML	언급 없음	언급 없음	온톨로지
접근 방식	양방향식	양방향식	양방향식	양방향식	상향식
산출물	언급 없음	언급 없음	언급 없음	언급 없음	단계별 산출물 정의

향식과 하향식을 결합한 양방향식을 사용하였고, SOMO는 기능 단계부터 서비스를 도출하는 상향식 접근 방식을 이용하였다. 마지막으로 기존의 SOA 방법론들은 산출물에 대한 언급이 없지만, SOMO는 단계별 산출물을 정의하여 나타내었다.

정량적 평가를 위해서는 정의한 평가 기준들의 가중치를 선정하고 평가하여야 한다. 평가 기준들의 가중치는 각각의 쌍대 비교를 통해 중요도를 매기는 AHP 분석 기법을 이용하여 산출한다. 평가 기준들의 가중치와 평가 점수를 매기기 위하여 현업의 IT 종사자 16명과 IT학과 대학원생 7명에게 설문조사를 수행했고, 객관성을 확보하기 위하여 각 평가 점수에 대한 간단한 이유를 함께 조사하였다.

평가 기준들의 가중치는 AHP 분석 기법을 이용하는데, 이를 위해서는 각 평가 기준들의 쌍대 비교가 필요하다. 본 논문에서는 설문 조사 결과 값과 의견들을 종합하여 평가 기준들의 쌍대 비교 값을 정하였다. <표 4>는 각 평가 기준들의 쌍대 비교 값을 바탕으로 AHP 분석을 한 결과 최종적으로 나온 가중치를 정리한 표이다. 쌍대 비교 시에 사용자 요구사항이 반영되지 않으면 의미가 없다는 의견이 많았기에 사용자 요구사항 반영도의 가중치가 가장 높게 나온 것을 볼 수 있다. 또한, SOA에서는 서비스가 기본이 되기 때문에 서비스 식별 상세성이 사용자 요구사항 반영도 다음으로 높은 것을 볼 수 있다. 반면, 프로세스 상세성은 다른 기준들에 비해 낮은 가중치를 얻었는데, 이는 프로세스가 아무리 상세하게 정의되었다고 해도 다른 기준들을 만족하지 않으면 SOA 방법론에

있어서 가치가 없기 때문이다.

〈표 4〉 평가 기준들의 가중치

평가 기준	가중치
프로세스 상세성	0.07
서비스 식별 상세성	0.26
유연성	0.14
재사용성	0.14
사용자 요구사항 반영도	0.40
합계	1

평가기준의 가중치를 정한 다음은 각 SOA 방법론들을 정의한 평가 기준에 따라 평가를 해야 한다. 평가는 가중치와 마찬가지로 설문조사로 이루어졌으며 <표 5>는 앞서 도출한 평가 기준들의 가중치와 설문 조사 결과 값들의 평균을 정리한 표다. 또한, 최종적으로 평가 기준들의 값은 미리 정의한 평가 기준들의 가중치를 원 점수에 곱하여 분석한다. <표 6>은 이러한 평가 값들에 가중치를 곱하여 정리하여 보여주고 있다.

결과로 나온 점수들을 보기 쉽게 비교하기 위하여 그래프로 나타내었다. [그림 15]는 각 평가 기준들마다 방법론들의 점수를 비교한 그래프를 보여주고 있다. 그래프를 보면 SOMO가 다른 기존의 SOA 방법론에 비해서 높은 점수를 얻은 것을 볼 수 있다. 우선 프로세스 상세성에서는 SOMA와 SOMO가 다른 방법론들에 비해 높은 것을 알 수 있다. SOMA는 기존의 방법론인 SOAD를 확장시킨 방법론으로 SOA 구현을 위한 전체 프로세스를 비즈니스 모델링부터 구현, 개발까지 나누어

〈표 5〉 기존의 SOA 방법론과 SOMO와의 평가 값들의 평균

평가기준	방법론	가중치	SODA	SOAD	SOMA	SOUP	SOMO
프로세스 상세성		0.07	2.22	2.48	3.13	2.74	3.65
서비스 식별 상세성		0.26	2.13	2.83	3.09	2.52	4.09
유연성		0.14	2.91	3.17	3.22	3.26	4.26
재사용성		0.14	2.91	3.04	3.22	3.26	4.35
사용자 요구사항 반영도		0.40	2.13	2.96	3.13	2.78	4.13

〈표 6〉 〈표 5〉의 값들에 가중치를 곱한 결과

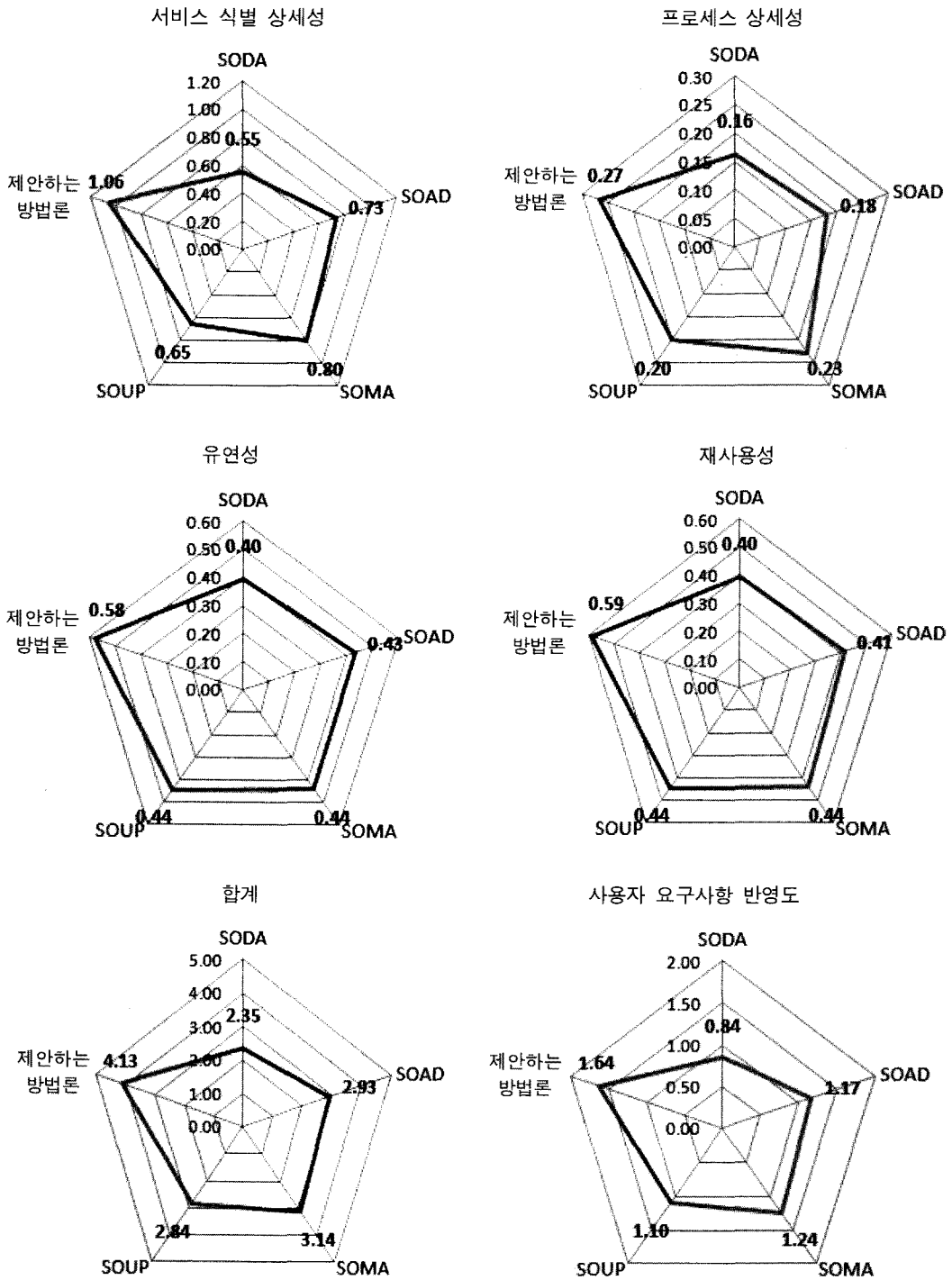
평가기준	방법론	가중치	SODA	SOAD	SOMA	SOUP	SOMO
프로세스 상세성		0.07	0.16	0.18	0.23	0.20	0.27
서비스 식별 상세성		0.26	0.55	0.73	0.80	0.65	1.06
유연성		0.14	0.40	0.43	0.44	0.44	0.58
재사용성		0.14	0.40	0.41	0.44	0.44	0.59
사용자 요구사항 반영도		0.4	0.84	1.17	1.24	1.10	1.64

설명하였으며 각 프로세스를 세부 프로세스로 나누었다. 따라서 프로세스 상세성에서 기존의 다른 SOA 방법론보다 높은 점수를 받은 것으로 분석되며, SOMO는 전체 프로세스를 세부 프로세스로 나누었을 뿐만 아니라 각 프로세스마다 예제를 들어 설명하며, 각 프로세스에 나오는 산출물들을 정의하였기에 다른 SOA 방법론에 비해 높은 점수를 얻을 수 있었다. 하지만, SOMO는 SOA 구현에 있어 전체적인 프로세스를 제안한 것이 아닌 서비스 도출에 초점을 맞추었기 때문에 다른 SOA 방법론에 비해 상대적인 점수는 높지만, 절대적인 점수는 높지 않은 것으로 보인다.

서비스 식별 상세성에서도 SOMA와 SOMO가 높은 점수를 얻은 것을 볼 수 있는데, SOMA는 SOA 구현을 위한 전체 프로세스를 세부 프로세스로 나눔으로써 서비스 식별 상세성에서도 기존의 다른 SOA 방법론보다 높은 점수를 얻을 수 있었지만, SOMO에 비해서는 높은 점수를 얻지 못했다. 이는 SOMO는 서비스 식별에 있어서 서비스 온톨로지를 제안함으로써 서비스 식별을 좀 더 구체적이고 명확하게 언급했기 때문으로 보인다.

유연성과 재사용성에서는 기존의 SOA 방법론들이 비슷한 점수를 얻은 것을 볼 수 있다. 이는 일반적인 SOA에서 얻을 수 있는 유연성과 재사용성에 대한 점수를 주었기 때문이라고 분석되어진다. SOMO가 가장 높은 점수를 얻을 수 있었던 것은 SOA에 있어서 핵심이 되는 서비스를 서비스 온톨로지를 이용함으로써 서비스의 재사용성과 유연성을 높일 것으로 예상되어지기 때문으로 보인다.

사용자 요구사항 반영도에서는 SODA가 다른 SOA 방법론에 비해 낮은 점수를 받았고, SOMO가 다른 SOA 방법론에 비해 높은 점수를 받은 것을 볼 수 있다. 이는 SODA의 경우 개발자 관점에서 요구사항을 분석하기 때문에 사용자 요구사항 반영도가 낮은 것으로 분석되어진다. 반면, SODA를 제외한 기존의 SOA 방법론들은 비즈니스 요구사항 관점에서 요구사항을 분석하기 때문에 SODA보다는 사용자 요구사항 반영도에서 높은 점수를 얻을 수 있었지만, SOMO보다는 낮은 점수를 얻은 것을 볼 수 있다. SOMO가 사용자 요구사항 반영도에서 높은 점수를 얻은 것은 SOMO가 사용자로부터 요구사항을 도출하는 사용자 관점에서 요



[그림 15] 기존의 SOA 방법론과 SOMO와의 평가 결과 비교 그래프

구사향을 분석하기 때문으로 보인다.

마지막으로 합계에서 SOMO가 높은 점수를 얻을 수 있었던 것은 다른 기준들에 대해서도 높은 점수를 얻었지만, 특히 높은 가중치를 가지고 있는 서비스 식별 상세성과 사용자 요구사항 반영도에서 높은 점수를 얻었기 때문으로 보인다.

5. 결 론

본 논문에서는 SOA를 위하여 온톨로지를 이용한 서비스 도출 방법론인 SOMO를 제안하였다. 먼저 서비스 재사용성과 유연성을 높이기 위하여 서비스 정의와 특징을 기반으로 서비스 온톨로지를 정의하였다. 서비스 정의 및 식별에 미리 정의된 서비스 온톨로지를 이용한다. 본 논문에서 사용하는 온톨로지는 클래스, 인스턴스, 관계로 구성되며 클래스는 프로파일, 프로세스, 인프라스트럭처로 구성된다. 서비스 온톨로지를 이용한 SOMO는 요구사항 분석, 서비스 식별, 서비스 정의 3단계로 이루어지며 각 단계마다 세부 활동으로 정의되며 각 단계마다 산출물을 정의하였다.

본 논문에서 제안한 서비스 도출 방법론이 갖는 가장 큰 의미는 서비스의 재사용성 및 유연성을 높임으로써 서비스의 품질 개선과 유지보수성을 개선할 수 있다는 점이라 할 수 있다. 또한 서비스 정의 및 식별 방법에 대한 구체적인 절차를 적용함으로써 소프트웨어 개발 시간을 단축시킬 수 있을 뿐만 아니라 SOA 개발 방법론의 개선에 기여할 것이다.

본 논문은 다음과 같이 두 가지의 향후 연구 과제가 있다. 첫째, 본 논문에서는 서비스 온톨로지를 정의할 때, 서비스 온톨로지의 세부 속성들은 자세하게 정의하지 않았다. 따라서 향후 연구에서는 서비스 온톨로지의 세부 속성들을 명확히 정의하고, 업무 특성에 맞추어 조정할 필요가 있다. 다음으로 본 논문은 SOA 기반의 소프트웨어 개발 방법론 중 서비스 도출에 초점이 맞추어졌으나 SOA 기반의 소프트웨어 개발을 위해서는 서비스

를 SOA 기반의 시스템에 적용시키기 위한 방법론이 필요하다. 이를 위해서는 도출한 서비스 온톨로지를 서비스 컴포넌트로 설계하는 물리적인 모델링 단계부터 서비스 컴포넌트들을 실제로 개발하는 개발 단계와 서비스 컴포넌트를 실제 시스템에 배치하고 운영하는 구현 단계 등이 필요하다. 따라서 서비스 컴포넌트를 설계 및 개발 하는데 필요한 기술과 서비스 컴포넌트를 개발하기 위한 언어 등을 정의하며 서비스 컴포넌트를 배치하는 방식 등에 대한 연구가 필요하다.

참 고 문 헌

- [1] 김유경, "유스케이스 재구성을 통한 서비스 식별", 「한국전자거래학회지」, 제12권, 제4호(2007), pp.145-163.
- [2] 김유경, 윤홍란, "SOA를 위한 서비스지향 개발 프로세스", 「한국전자거래학회지」, 제12권, 제2호(2007), pp.75-93.
- [3] 김유경, 윤홍란, 박재년, "웹 서비스 개발 프로세스에서 구현전략 결정을 위한 평가 지침", 「한국정보과학회논문지 : 소프트웨어 및 응용」, 제33권, 제5호(2006), pp.460-469.
- [4] 박용성, 박태근, 「의사결정론 : AHP를 위한」, 자유아카데미, 2001.
- [5] 심우혁, "사용자 관점의 SOA 서비스 모델링 기법", 숭실대학교(석사학위논문), 2009.
- [6] 윤홍란, "MASOD : SOA를 위한 서비스 지향 개발 방법론", 숭실대학교(박사학위논문), 2006.
- [7] 이현주, 최병주, 이정원, "서비스 지향 아키텍처를 위한 컴포넌트기반 시스템의 서비스 식별", 「한국정보과학회지」, 제35권, 제2호(2008), pp.70-80.
- [8] 정인호, "SOA 시스템 구축을 위한 관점지향 서비스 식별 기법", 숭실대학교(석사학위논문), 2009.
- [9] 최진호, 이진미, 임상현, "SOA 구현을 위한 프로세스 기반 서비스 정의 방법론에 관한 연구",

- 「한국품질경영학회지」, 제38권, 제1호(2010), pp.1-9.
- [10] 한상우, 박선희, 노재호, “Service Oriented Architecture 적용을 위한 서비스 식별 기법”, 「한국정보과학회지」, 제24권, 제11호(2006), pp.27-31.
- [11] Akao, Y., *Quality Function Deployment*, Sönkeherrmann Albers AndreasEd. Productivity Press, 1990.
- [12] Arsanjani, A., “Service-oriented modeling and architecture”, *IBM developer works*, 2004.
- [13] Blanchard, B. and W. Fabrycky, *Systems Engineering and Analysis*, Ed. Prentice Hall, 2006.
- [14] Brown, A., S. Johnston, and K. Kelly, *Using service-oriented architecture and component-based development to build web service applications*, Rational Software Corporation, whitepaper, 2002.
- [15] Gruber, T., “Toward principles for the design of ontologies used for knowledge sharing”, *International Journal of Human Computer Studies*, Vol.43, No.5(1995), pp.907-928.
- [16] Krafzig, D., K. Banke, and D. Slama, *Enterprise SOA : Service-Oriented Architecture Best Practices*, Prentice Hall, 2005.
- [17] Mittal, K., “Service Oriented Unified Process (SOUP)”, *IBM Journal*, <http://www.kunalmittal.com/html/soup.html>, 2005.
- [18] Nigam, S., *Service Oriented Development of Applications (SODA) in Sybase Workspace*, Sybase Inc. whitepaper, 2005.
- [19] Zimmermann, O., P. Krogdahl and C. Gee, “Elements of service-oriented analysis and design”, *IBM developerworks*, 2004.

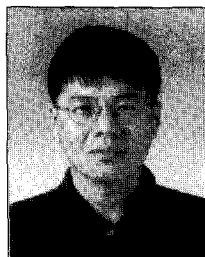
◆ 저자 소개 ◆

**장 효 선 (yuz@hanmail.net)**

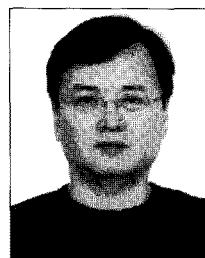
중앙대학교 정보시스템학과를 졸업하고, 현재 중앙대학교 정보시스템학과 석사과정에 재학 중이다. 주요 관심분야는 데이터베이스, 서비스, 온톨로지 등이다.

**박 세 권 (psk3193@cau.ac.kr)**

서울대학교 공과대학과 대학원 산업공학과에서 1978년과 1981년에 공학사(BS)와 공학 석사(MS)를 취득하였으며, Texas A&M 대학교 대학원 산업공학과에서 1985년에 산업공학공학 박사(Ph.D.)를 취득하였다. 1978년부터 1981년까지 한국전자통신연구소에서 연구원으로 근무하였으며, 1985년부터 1987년까지 한국전자통신연구원 통신망계획부에서 선임연구원으로, 1987년부터 1990년까지 농촌경제연구원 농림수산부 소프트웨어하우스 실장(수석연구원)으로 농업농촌정보화 하부구조 구축 연구를 수행하였다. 1990년부터 현재까지 중앙대학교 정보시스템학과에 재직 중이며 연구 관심분야는 시스템공학 등이다.

**류 승 완 (ryu@cau.ac.kr)**

고려대학교 산업공학과에서 1988년과 1991년에 각각 공학사와 공학 석사를 취득하였으며, 뉴욕주립대(SUNY at Buffalo) 산업공학과에서 2003년에 공학박사를 취득하였다. 1991년부터 1993년까지 LG전자 영상미디어연구소에서 주임연구원으로 근무하였으며, 1993년부터 2004년까지 한국전자통신연구원 이동통신연구단에서 선임연구원으로 근무하면서 2세대 CDMA 디지털 이동통신, 3세대 이동통신 IMT-2000, 4세대 이동통신 시스템 연구를 수행하였다. 2004년부터 중앙대학교 정보시스템학과에서 교수로 재직 중이며, 2004년부터 2007까지 한국전자통신연구원 이동통신연구단에서 초빙연구원으로 이동통신 연구를 수행하였다. 주요 연구 관심분야는 이동통신시스템 설계 및 성능분석, 무선 MAC 프로토콜, 차세대 이동통신 서비스 및 비즈니스 모델 개발 등이다.

**신 동 천 (dcshin@cau.ac.kr)**

서울대학교 컴퓨터공학과를 졸업한 후 한국과학기술원 전산학과에서 석사와 박사학위를 각각 취득하였다. 1991년부터 한국전산원에서 선임연구원으로 근무한 후 1993년부터 중앙대학교 정보시스템학과에서 교수로 재직 중이다. 최근의 주요 관심분야는 데이터베이스 설계 및 관리, IT 서비스 모델링, 기업정보시스템 등이다.