

유즈케이스 분석을 통한 효율적인 서비스 모델링 방법

신수미^o, 류범중, 신동구
 한국과학기술정보연구원
 {sumi.ybj,lovesin}@kisti.re.kr

Effective Service Modeling Method through Use case Analysis

Sumi Shin, Bum-jong Yu, Dong-gu Shin
 Korea Institute of Science and Technology Information

요 약

SOA에 대한 도입희망이 여러 분야에서 가속화 되고 있지만 서비스 모델링에 대한 구체적인 방법과 사례가 부족하며, 제안된 모델링 방법들은 비즈니스 분석에 집중되어 있거나 서비스의 도출에 대한 구체적 절차와 서비스 식별 기준을 명확하게 정의하지 않아 실제 시스템의 개발에 적용하는데 어려움이 있다. 본 논문에서는 효율적인 서비스 모델링을 위하여 지금까지 많은 시스템들이 구축에 활용되었던 방법론에 대한 결과물인 유즈케이스(Use Case)를 활용하여 프로젝트의 특성에 관계없이 적용될 수 있는 구체적인 서비스 정의 방법을 소개한다. 또한 이를 위한 서비스 구분과 서비스 아키텍처를 제안하고 제안된 서비스 정의 방법을 국가성과정보시스템의 구현에 적용한 사례를 소개한다

1. 서 론

1996년 가트너(Gartner)그룹에 의해 처음 소개된 SOA의 개념실현 가능성에 대한 기대가 확산되면서 SOA에 대한 도입희망이 예전보다 가속화 되고 있지만 다양한 분야의 구현사례는 부족한 형편이다. SOA 구현의 성공을 위해서는 요구사항 변화에 민첩하게 반응할 수 있도록 단일화되고 작업 중심으로 서비스를 정의하는 것이 중요하다. 그러나 지금까지 제안된 많은 서비스 정의 방법들은 구체적 절차와 서비스 식별 기준을 명확하게 제시하고 있지 않고 있어 실제 시스템의 개발에 적용하는데 어려움이 있다.

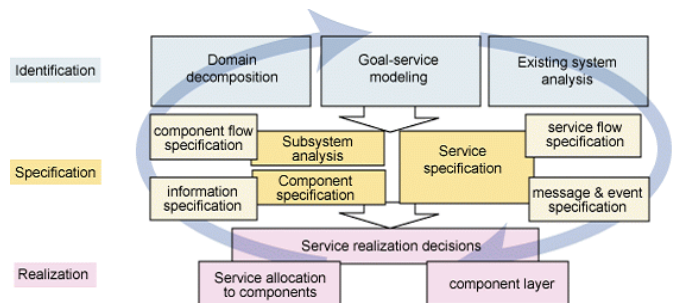
본 논문에서는 효율적인 서비스 정의 방안으로 지금까지 많은 시스템들이 구축에 적용되었던 유즈케이스(Use case) 모델링 결과를 서비스 정의를 위해 활용하는 방법을 제안하며 서비스 정의를 위한 요소로 시스템 구현 관점에서의 서비스 구분을 함께 제안한다. 본 논문의 구성은 2장에서 관련연구로 SOA기반 방법론을 살펴보고, 3장에서 유즈케이스를 이용한 서비스 모델링 방안을 제시하며, 4장에서 제시된 서비스 모델링 방법을 시스템 구축에 이용한 사례를 확인하고 5장에서 결과를 정리한다.

2. 관련연구

가트너(Gartner)에서 제안한 SODA(Service Oriented Development of Application)는 SOA를 구현하기 위해 서비스를 설계, 개발 및 조합 등 일련의 과정에 대한 원칙을 제공하며 서비스를 개발하고 조합하여 비즈니스

프로세스를 지원할 수 있는 애플리케이션 개발 모델 제공을 목표로 한다.

IBM의 SOMA(Service-Oriented Modeling and Architecture)는 SOA를 지원하는 모델링, 분석 설계 기술과 활동을 포함하며, IBM에서 제안한 각 레이어 내의 요소를 결정 하도록 한다.[1] SOMA는 비즈니스 프로세스를 기능을 중심으로 서브시스템으로 분해하는 하향식(Top-down) 방식과 이미 가지고 있는 시스템의 분석을 통해 비즈니스 프로세스에서 수용 가능한 기능을 가진 후보 서비스를 도출하는 상향식(Bottom-up) 방식을 제안하며, 하향식이나 상향식으로 식별되지 않은 다른 서비스를 찾아내는 목표 서비스 모델링(Goal-Service Modeling)을 수행하도록 한다.



[그림 1] SOMA 방법론

그러나 제안된 SOA지원 방법론들은 서비스 구체적인 서비스 정의방법에 대한 제안은 부족하여 구체적 구현 기술 적용 및 활용에 대한 방안이 필요 하다.

3. 서비스 모델링 방안

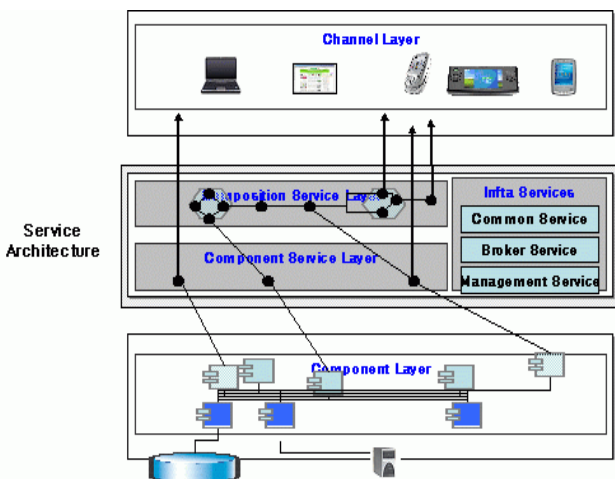
3.1 서비스 아키텍처 제안

효과적으로 서비스를 모델링 하기 위해서는 서비스 집합의 논리적 구성을 나타내는 서비스 아키텍처의 정의와 서비스의 구분이 우선 필요하다. 본 논문에서 제안하는 서비스 아키텍처는 [그림2]와 같이 비즈니스 서비스 레이어와 컴포넌트 서비스 레이어, 인프라 서비스 레이어를 갖는다. 비즈니스 서비스 레이어는 비즈니스 흐름에 따라 서비스를 조합하는 계층이며 SOA의 핵심개념인 서비스를 제공하는 컴포넌트 서비스 계층, 그리고 이들 계층에 공통서비스를 제공하는 인프라 서비스 레이어로 구성하였다.

서비스는 구분은 아키텍처의 레이어를 고려하였다. 아키텍처의 3개 레이어 중 개발을 위한 정의 대상이 되는 서비스는 비즈니스 서비스 레이어와 컴포넌트 서비스 레이어에 놓이는 서비스로 각각 비즈니스 서비스와 컴포넌트 서비스로 구분하였다.

컴포넌트 서비스는 구현 형태에 따라 3가지 형태를 갖는다.

- 신규 컴포넌트 서비스: 신규로 작성되는 서비스
- 재사용 컴포넌트 서비스: 내 외부에 존재하는 web service를 사용하는 서비스
- 래핑(Wrapping) 컴포넌트 서비스: 웹서비스 인터페이스를 사용하지 않은 기능을 사용하기 위하여 어댑터 등을 사용하여 래핑한 서비스



[그림2] 서비스 아키텍처

3.2 서비스 정의방법

지금까지 많은 시스템의 구축에서 활용된 방법론에서 정의하고 있는 산출물을 활용하여 서비스를 정의하게 되면 비즈니스의 특성이나 프로젝트에 대한 특성에 대한 영향이 적은 일반적인 서비스 모델링 방법을 수립할 수 있다. 본 사용자와 시스템의 상호작용을

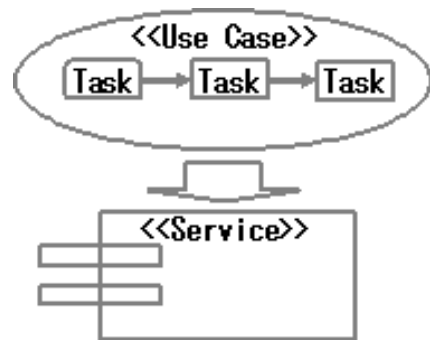
표현하며 시스템간, 사용자와 시스템간에 상호작용 해야 하는 기능이나 특성에 초점을 맞춘 유즈케이스를 활용하여 서비스를 정의하는 방법을 제안한다. 유즈케이스는 반드시 구현되어야 할 시스템의 목표가 될 수 있는데 이를 활용하여 서비스를 식별하면 각 actor 를 위해 시스템이 제공해야 하는 기능을 확인할 수 있어 비즈니스를 지원하는 기능 단위의 서비스 식별이 용이하고 독립 서비스도 쉽게 정의할 수 있다. 서비스 식별을 위해 유즈케이스를 분석하기 위해서는 1:1 매핑과 분할 방식을 이용한다.

서비스는 3.1 에서 정의한 바와 같이 컴포넌트 서비스와 비즈니스 서비스로 구분함으로써 1:1 매핑과 분할 방식을 적용할 수 있다.

1:1 매핑과 분할에 의한 서비스 도출방식은 아래와 같다.

- 1:1 매핑

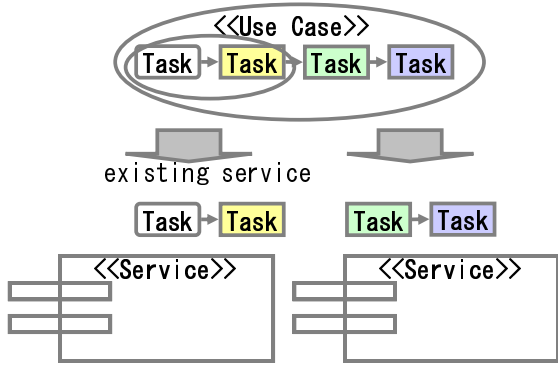
하나의 use case 를 하나의 service 로 도출하는 방법이다. use case 내의 task 집합이 다른 use case 가 갖는 task 의 집합과 의존관계 없이 중복되지 않고 독립적인 경우 하나의 서비스로 도출해낼 수 있다.



[그림 3] 1:1 Mapping

- 분할

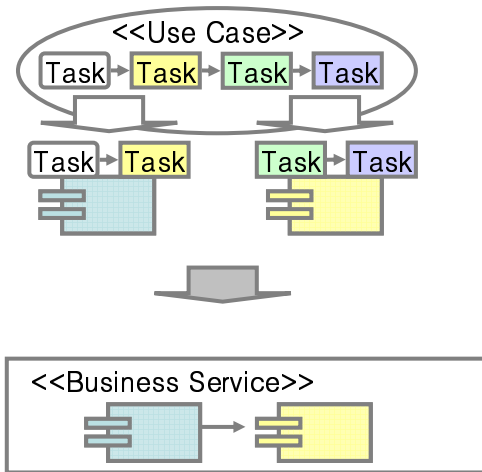
하나의 유즈케이스를 좀 더 작은 서비스 단위로 나누는 경우이다.



[그림 4] 분할

하나의 유즈케이스가 여러 개의 작업으로 표현된다고 볼 때, 서로 독립적으로 작용하는 작업에 대하여 유즈케이스내의 작업 집합이 다른 유즈케이스가 갖는 작업집합과 중복되는 경우나, 기존에 있는 서비스나 신규로 개발될 컴포넌트 서비스 중 유즈케이스의 일부 작업을 수행하는 서비스가 있을 경우 분리하여 두 개 이상의 서비스로 분할한다.

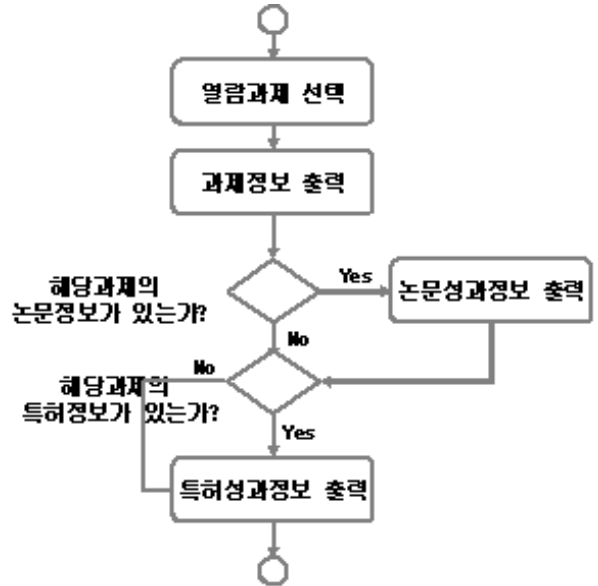
1:1 매핑이나 분할을 통해 도출된 서비스는 또 다시 다른 유즈케이스의 부분 작업이 될 수 있기 때문에 서비스 분석은 반복하여 점진적으로 시행하여야 한다. 분할 대상이 된 유즈케이스는 여러 컴포넌트 서비스들의 기능을 비즈니스 논리에 따라 통합한 서비스로 구현할 수 있도록 비즈니스 서비스로 정의한다. 비즈니스 서비스는 분할 대상이 된 유즈케이스를 1:1로 매핑하는데 이와 같이 비즈니스 서비스를 정의하면 비즈니스 서비스를 구성하는 서비스들의 집합을 바로 인지할 수 있다. 비즈니스 서비스는 다양한 서비스들의 조합을 통해 구축되기 때문에 서비스들의 흐름을 정의되어야 하는데 다수의 독립 작업으로 분리된 유즈케이스를 비즈니스 서비스로 식별하게 되면 각 작업간의 관계나 흐름이 미리 분석되었기 때문에 여러 서비스들의 기능 조합도 쉽게 정의할 수 있다.



[그림 5] 비즈니스 서비스

4. 적용 사례

제안한 서비스 식별방법을 적용하여 구현할 대상은 국가성과정보시스템으로 해당 시스템은 관련정보를 가진 유관기관으로부터 데이터를 수집하여 기초 데이터를 확보하며, 수집된 정보는 특허청의 특허 검색 웹서비스와 논문발간 학회시스템의 논문검색 API를 활용하여 검증할 수 있어야 한다. 수집되고 검증된 정보는 다양한 형태의 검색과 조회가 가능하여야 하며, 해당 논문과 특허가 특정 과제의 성과인 경우 타 응용시스템인 과제정보시스템과 연동하여 해당 과제의 상세정보를 보여줄 수 있어야 한다. 또한 특허나 논문의 저자에 대해서는 또 다른 응용시스템인 인력정보시스템과 연동하여 해당 인물의 상세정보를 보여줄 수 있어야 한다. 해당 요구사항을 분석하여 도출한 유즈케이스 중 일부로 논문 검증, 특허 검증, 과제별성과조회, 과제상세조회, 논문성과검색, 논문성과조회, 특허성과검색, 특허성과조회, 인력정보조회가 정의되었다. 정의된 유즈케이스를 이용해 1:1 mapping 방법으로 서비스를 도출하면 9 개의 서비스가 식별된다. 이제 서비스 분할을 위해 식별된 서비스를 일부 작업으로 가지고 있는 유즈케이스가 있는지 확인한다.



[그림 6] 과제별 성과조회 세부 시나리오

사례의 경우 과제별 성과조회 유즈케이스의 세부 작업이 [그림 6]과 같이 구분되는데 논문성과정보 출력과 특허성과정보 출력은 논문성과조회, 특허성과조회서비스에서 제공하는 기능이므로 해당 부분을 각각의 서비스로 분할하고 나머지 부분만 하나의 서비스로 정의한다. 즉 과제별 성과조회 유즈케이스는 신규로 개발되는 과제정보조회,

논문성과정보, 특허성과정보 서비스를 활용하여 구현할 수 있어 3 개 서비스로 분할 되었다. 과제별 성과조회 성과조회 유즈케이스는 3 개 서비스를 조합하는 비즈니스 서비스로 정의하였다. 이와 같이 1:1 mapping 방법과 분할 방법을 통해 그림의 유즈케이스를 분석하여 식별한 서비스는 표와 같으며 대부분의 서비스가 유즈케이스와 1:1 매핑되어 서비스로 도출된 것을 볼 수 있다.

[표 1] 유즈케이스분석에 따른 식별 서비스

유즈케이스	서비스명	정의방법
NDSL논문 검증	논문 검증 서비스	1:1 매핑
특허 검증	특허 검증 서비스	1:1 매핑
과제별성과조회	과제조회서비스	분할
	논문조회서비스	
	특허조회서비스	
과제조회	과제조회서비스	1:1 매핑
논문성과검색	논문성과검색서비스	1:1 매핑
논문성과조회	논문성과조회서비스	1:1 매핑
특허성과검색	특허성과검색서비스	1:1 매핑
특허성과조회	특허성과조회서비스	1:1 매핑
인력정보조회	인력조회서비스	1:1 매핑

식별된 서비스를 서비스 아키텍처에 표현하면 그림과 같다. 서비스 아키텍처에 서비스를 표현함으로써 독립적인 기능을 가진 서비스 간의 종속관계를 파악할 수 있으며 하나의 서비스의 변화가 시스템에 미치는 영향을 알기 쉽게 해준다.

5. 결론

제안된 많은 SOA 개발 방법론이 서비스 도출 방안으로 비즈니스 프로세스를 분석하거나 또는 구체적인 서비스 식별방안을 제안하고 있지 다양한 성격의 프로젝트에 적용하는데 한계가 있다.

본 논문에서 제안한 많은 시스템들이 구축에 적용되었던 방법론의 산출물을 활용하는 방법으로 유즈케이스를 분석하여 서비스를 도출하는 방식은 비즈니스 모델링과 서비스 모델링을 분리하고, 다양한 프로젝트에 적용될 수 있는 유연성을 확대할 수 있다. 유즈케이스를 분석하여 서비스를 도출해 본 결과 서비스 정의 기준이 명확히 정의됨으로써 서비스 식별기준의 모호함이 상당부분 해소되었으며, 시스템의 기능에 필요한 서비스가 누락되는 경우 없이 모든 요구사항을 만족하는 시스템을 구현할 수 있었다. 아울러 타 서비스에 종속적인 작업이 모두 구분되어 서비스 조합하는 형태로 정의됨으로써 유사기능의

중복개발이 방지되었다.

그러나 본 연구에서 제안한 유즈케이스 분석에 의한 서비스 도출 방식에도 서비스 크기가 오로지 유즈케이스 에 의해서만 정의된다는 한계가 있었다. 향후에는 식별된 서비스를 적절한 레벨로 정의하는 단계를 추가하여 적절한 레벨로 조정할 수 있는 방안에 대한 연구를 보완하고자 한다.

[참고문헌]

- [1] Arsanjani, A.: Service-Oriented Modelling and Architecture (SOMA), IBM developer-Works 2004, <http://www.ibm.com/developerworks/webservices/library/ws-soa-design1>
- [2] Erl, Thomas (2005). Service-Oriented Architecture: Concepts, Technology, and Design (Hardcover). Prentice Hall
- [3] Zimmermann, Olaf., Schlimm, Pestel., 2005. Analysis and Design Techniques for Service-Oriented Development and Integration. In INFORMATIK 2005 - Informatik LIVE!. INFORMATIK 2005.