온톨로지 기반 SOA 개발 방법론

최고봉, 박세권, 류승완, 명광식
중앙대학교 정보시스템학과

Ontology Based SOA Development Methodology

Choi, Kobong, Park, Seikwon, Ryu, Sungwan, Myung, kwangsik
chung-ang Univ.
E-mail : ca_ckb@hanmail.net. psk3193@cau.ac.kr, ryu@cau.ac.kr, mks45@post.cau.ac.kr

요 약

최근 비즈니스 환경이 빠르게 변화함에 따라 기업의 민첩한 대응이 요구되고 있으며, 서비스지향 아키텍처(Service-Oriented Architecture(SOA))가 유력한 대안으로 제시되고 있다. SOA는 서비스를 공유하고 재사용 할 수 있도록 정보시스템을 구축하는 소프트웨어 설계 방법론으로써, 급변하는 비즈니스 환경에 맞도록 민첩한 대응이 가능하도록 한다. 이러한 SOA의 장점으로 인해 현재 많은 기업에서 SOA를 이용한 시스템 개발을 진행하고 있다. 그러나, 이러한 SOA를 적용한 시스템 개발 노력이 계속 시도되고는 있지만 SOA기반의 표준화된 방법론은 아직 정립되지 않은 실정이다. 따라서, SOA를 이용한 시스템 개발은 현재 많은 어려움에 직면하고 있다.

본 논문에서는 표준화된 방법론으로써 온톨로지 기반의 SOA 개발방법론을 제시한다. 제안하는 방법은 온톨로지를 이용하여 서비스를 서비스 프로파일과 서비스 모델로 구분하여 정의하여 서비스 식별을 위한 기본 프레임워크를 제시한다. 또한 식별된 서비스들을 온톨로지 개념을 이용하여 검색, 재사용 할 수 있는 방법론을 제시하여 서비스 재활용도를 높일 수 있도록 한다.

본 논문에서 제안하는 온톨로지 기반의 SOA 개발 방법론은 현재 사용중인 다른 개발방법론과 비교 분석하였다. 비교 결과 본 논문에서 제안하는 온톨로지 기반 SOA 개발방법론은 현재까지 제안된 다른 방법보다 효율적이고 체계적인 개발 결과를 제공하는 것으로 나타났다. 본 논문에서 제안하는 연구 결과는 SOA에 표준방법론의 개발의 기반을 제공하여, 서비스의 의미론적 접근에 대한 시도 등 향후 연구에 중요한 방향성을 제공할 수 있을 것으로 기대 된다.

1. 서론

최근 비즈니스 환경은 외부의 변화 및 내부의 요구사항에 대해 보다 빠른 대응을 요구하고 있다. 이는 기업의 성공적인 운영을 위해 예측되어진 변화 외에 예측하지 못한 변화에 대해서도 빠르게 대응 할 수 있는 능력을 요구하고 있기 때문이다. IT환경 또한 기업의 판단에 따른 비즈니스 변화에 대해 빠르게 대처 할 수 있도록 요구하고 있으나 기존의 IT방식으로 비즈니스 환경의 변화요구에 맞게 IT환경을 재구성하기에는 너무 많은 시간과 비용이 소요 된다.[1] 이에 대한 대응책으로 서비스 지향 아키텍처(Service Oriented Architecture (이하 SOA))가 제시되었다.

SOA는 사용자 애플리케이션의 기능을 서비스

형식으로 전달하는 분산 시스템 구축의 한 접근 방법인데 느슨한 결합(loosely coupling)을 통해 유연성을 높이고, 표준 인터페이스를 기반으로 함으로써 이기종 환경에서의 비즈니스 민첩성을 높일 수 있다.[2]

1996년 가트너 그룹에서 SOA를 발표한 이후 SOA에 대한 많은 연구가 이루어졌고 SOA 기반으로 많은 기업에서 프로젝트를 수행해왔다. 하지만 아직까지 SOA에 대한 표준 방법론이 존재하지 않아 많은 어려움이 뒤따르고 있다. 이에 본 논문에서는 온톨로지를 기반으로 한 SOA방법론을 제안하고자 한다.

본 논문에서 제안하는 SOA 방법론은 BPM을 통해서 비즈니스 프로세스를 정제하고 온톨로지를 이용하여 서비스를 식별함으로써 서비스의 재사용성과 유연성을 높였다.

본 논문의 구성은 다음과 같다. 2.1장에서는 SOA의 개요와 온톨로지 개요 그리고 BPM의 개요를 간단히 살펴보고 온톨로지와 BPM의 SOA와의 연관성에 대해서 설명하였으며 2.2장에서는 제안하는 방법론의 아키텍처와 방법론의 절차에 대해서 설명한다. 2.4장에서는 본 방법론의 검증을 3장 결론 순으로 구성된다.

2. 본론

2.1 관련연구

2.1.1SOA 개요

SOA란 기존 애플리케이션들이 가진 기능을 비즈니스적인 의미를 가지는 기능 단위로 묶고 표준화된 호출 인터페이스를 통해 서비스라는 소프트웨어 컴포넌트 단위로 재조합한 후, 이 서비스들을 서로 조합(Orchestration)해 업무 기능을 구현한 애플리케이션을 만들어내는 소프트웨어 아키텍처이다.[3]

서비스의 정의는 특정 기술이나 플랫폼에 종속되지 않는 표준 인터페이스 기반으로 기능을 가지고 있는 단위로써, 느슨한 결합의 특성을 지니며 상호 조합이 가능한 단위이다. 서비스는 여러 특징을 가지고 있는데, 주요 특징으로는 느슨한 결합, 조합성, 조밀성 그리고 발견 가능성을 들 수 있다.

또한, 서비스 공급자, 서비스 요청자, 서비스 저장소라는 세요소가 유기적으로 연계되어 있는 특징을 갖는다.[4]

2.1.2 온톨로지(Ontology) 개요

온톨로지란 용어 사이의 관계를 정의하고 있는 일종의 사전과 같은 것으로 Gruber의 정의에 의하면 "온톨로지란 공유된 개념에 대한 정형화된 명세이다."[5]

온톨로지를 SOA에 적용시킴으로써 얻을 수 있는 장점으로는 서비스의 정의와 식별이 유연해질 수 있으며 보다 명확한 서비스 정의가 가능하다는 점이다.[6]

2.1.3 비즈니스 프로세스 관리(Business Process Management, (BPM))

기업 경영환경이 빠르게 변화하고 시장에 대한 민첩한 대응이 기업의 중요한 경쟁력이 됨에 따라 BPM은 실시간으로 비즈니스 프로세스를 변경하고, 개별적으로 관리되었던 비즈니스 프로세스를 수명주기 전체에 걸쳐서 일관성이 있게 관리하기 위한 경영관리 기법으로 등장하였다.[7]

BPM에서는 모든 비즈니스 프로세스를 17가지 방법으로 나누어 설명하고 있다. 그리고 이 17가지 프로세스를 다시 단순 프로세스와 복잡한 프로세스로 구분 하고 있다.[8]

2.2 제안하는 개발방법론

2.2.1 서비스 구성 요소

SOA의 적용을 위해서는 서비스를 정의하고 식별해야 하는데, 본 논문에서 제안하는 서비스 구성요소는 그림 1과 같다.

서비스는 서비스의 특성을 나타내는 서비스 프로파일과 서비스 형식을 나타내는 서비스 모델로 구분할 수 있다. 특성을 나타내는 프로파일은 네 가지 구성요소를 갖는데, 파라메터, 위치 정보, 서비스 수준, 액터이다. 네 가지의 요소의 자세한 설명은 아래와 같다.

- **파라메터**

  서비스는 서비스의 기능을 실현하기 위해 서비스와 관련된 데이터를 주고받는다. 이때 주고 받

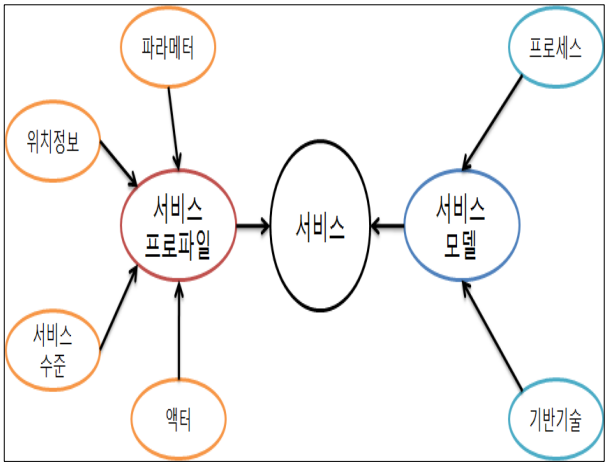

그림 1 서비스 구성 요소

는 데이터들에 대한 정보를 담고 있는 부분이다.

- **위치 정보**

  SOA의 특징 중 서비스의 발견성과 조합성은 필요한 서비스를 찾아서 재사용할 수 있도록 한다. 재사용하고자 하는 서비스가 어디에 위치하는지 하는지를 알아야 하는데 그에 대한 정보가 포함된다.

- **서비스 수준**

  비즈니스 프로세스 상에서의 서비스들은 일정 수준 이상으로 실행되어야 한다. 예를 들어 웹페이지에서 “로그인 서비스는 24시간 이용 가능해야 한다.” 라고 요구사항을 정의하게 되면 로그인 서비스는 언제나 이용할 수 있는 가용성 측면에서의 서비스 수준을 정의 할 수 있다. 하지만 만약 “로그인 서비스가 24시간 제공되며 사용자 인증은 1초 이내에 완료 되어야 한다“ 라고 요구사항을 정의하면 로그인 서비스는 24시간이라는 가용성외에도 응답속도 1초 이내 라는 응답속도에 대한 서비스 수준도 정의해야 한다.

  따라서 서비스 수준은 서비스의 서비스 수준을 개발 요구사항에 따라 정량적으로 설명하는 부분이다.

- **액터**

  서비스는 서비스를 제공하는 제공자와 서비스를 소비하는 소비자로 크게 구분 할 수 있으며 또한 세부적으로는 각각의 기능을 실현하기 위해서 서비스가 이용하는 참여자들이 존재 한다. 따라서 서비스에 누가 참여하는지 등의 정보를 포함하고 있어야 한다.

앞서 살펴본 네 가지 서비스 특성은 서비스의 형태를 결정하지는 못하지만, 서비스의 기본 정보를 담고 있어 서비스 재사용시 필요한 정보를 제공한다. 예를 들면 로그인 서비스를 실행하기 위해서 필요한 데이터나 액터, 서비스의 위치와 서비스 수준은 알 수 있지만, 로그인 서비스가 어떤 절차를 이루어져 있는지는 알 수 없다.

이런 서비스의 절차에 대한 정보를 담고 있으며 그에 따라 서비스를 어떤 서비스라고 구분할 수 있게 해주는 정보는 서비스를 구성하는 또 다른 요소인 서비스 모델에서 설명하고 있다. 서비스 모델은 서비스의 형태를 결정 할 수 있는 서비스 프로세스와 서비스의 기반기술로 구분되는데, 각각의 설명은 아래와 같다

- **프로세스**

  프로세스는 서비스가 가지고 있는 목적을 실행하기 위한 방법을 포함 하고 있다. 예를 들어 로그인 서비스는 로그인을 위한 사용자의 데이터를 입력받는 부분과 입력받은 사용자의 데이터를 서버의 저장 내용과 비교하는 부분, 비교 결과에 따라 로그인을 수락하거나 거절하는 부분의 세 가지 프로세스로 구성되어 있다.

  다른 모든 서비스들도 위와 같은 방법으로 프로세스를 통해서 서비스의 기능을 하는데, 본 방법론에서는 이 프로세스의 형태에 따라 서비스를 분류한다. 이에 대한 자세한 설명은 방법론의 프로세스 부분에서 설명하도록 한다.

- **기반 기술**

  기반 기술은 서비스가 어떤 기술 플랫폼을 기반으로 작성되었는지를 나타낸다. 예를 들면 서비스가 자바언어로 작성 되었다던지, 내부 프로세스들이 메시지를 주고 받을 때 XML을 사용하는지 RMI를 사용하는지에 대한 정보를 담고 있다.

위에서 설명한 서비스 구성요소들은 하나의 서비스를 식별하는데 각각의 중요한 역할을 하게 되는데, 각각의 구성요소의 구성 방법은 후에 설명한 본 방법론의 프로세스에서 설명하도록 한다.

또한 위의 구성요소 중 파라메터와 액터, 그리고 프로세스의 경우는 온톨로지를 구축하여 의미적인

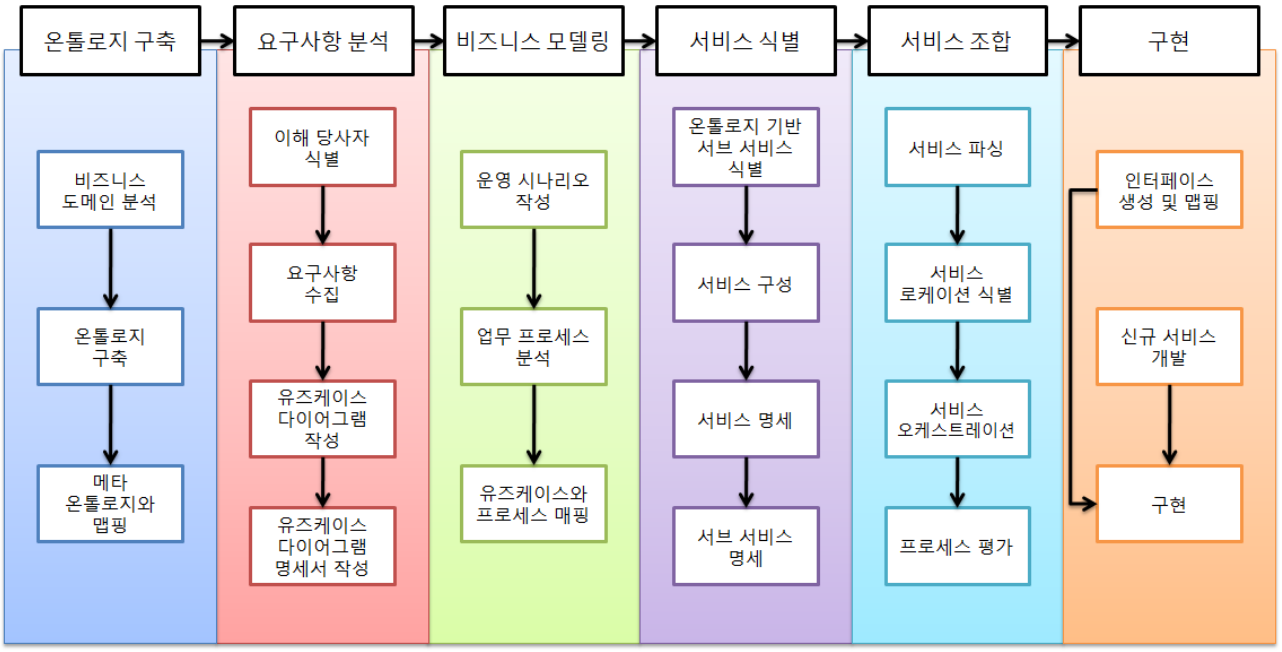

그림 2 제안하는 개발방법론 전체 프로세스

공유를 함으로써 서비스 재사용성을 높일 수 있다.

2.2.2 제안하는 개발 방법론 프로세스

제안하는 개발 방법론 6단계로 그림2와 같다. 프로세스는 총 6단계로 구성되어지며 각 프로세스에 대한 설명을 위해 설명과 함께 쇼핑몰 개발 과정의 일부를 함께 예로 들어 설명할 것이다.

- **단계 1 온톨로지 구축**

온톨로지 구축 단계에서는 구축하고자 하는 시스템의 도메인에 대해서 분석하고, 그 결과를 바탕으로 도메인 온톨로지를 구축하여 메타 온톨로지와 매핑을 통해 통합 온톨로지 구축에 목적이 있다. 온톨로지를 정의하여 메타 온톨로지와 맵핑하는 것은 다른 비즈니스 도메인에 속하는 서비스를 재사용하고자 할 때 현 도메인과 다른 도메인간의 의미적인 결합을 통해 서비스 검색을 하고자 함이다.

예를 들어 쇼핑몰을 구축할 때, 쇼핑몰을 이용하는 사용자와 서비스 제공자, 쇼핑몰에서 파는 물건이나 화폐단위, 할인율 등을 분류하여 온톨로지를 구축하여 메타 온톨로지와 연결함으로써 메타 온톨로지와 연결되어 있는 쇼핑몰이 아닌 다른 비즈니스 도메인상의 서비스를 재사용할 수 있게 하는 것이다.

단계 1-1 비즈니스 도메인 분석

조직의 업무 환경을 비롯하여 시스템 참여자들을 식별하고 사용되는 데이터의 종류를 파악한다. 이 단계에서 분석되는 정보들은 온톨로지 구축 단계에서 온톨로지 구축의 데이터들로 이용된다. 따라서 비즈니스 도메인에 대한 세부적인 사항까지 명확하게 정의할 필요가 있다.

쇼핑몰의 경우 쇼핑몰에서 취급하는 상품, 조직구성, 쇼핑몰 정책 등을 면밀히 분석하여 각각에 대한 정의를 한다.

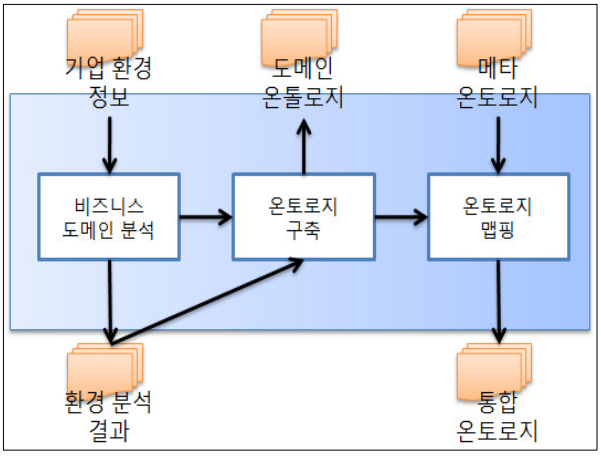

그림 3 온토롤지 구축단계 프로세스

단계 1-2 온톨로지 구축

비즈니스 도메인 분석 단계에서 정의한 개념들을 메타 온톨로지와 같은 형식으로 구축하는 단계이다. 온톨로지 구축은 서비스 구성에서 언급하였듯이 파라메터, 액터 그리고 프로세스에 대해서 구축한다.

파라메터 온톨로지는 비즈니스 프로세스 상에서 이용되는 데이터들을 정의하는 것으로 그림4와 같은 형태를 띤다. 쇼핑몰의 경우는 쇼핑몰에서 취급하는 상품군, 화폐단위, 할인율 등이 이에 해당 된다.

value는 이용되는 파라메터를 가공하는 데이터들로써 예를 들면 세율, 할인율 등이다. 반면 string 범주 속하는 파라메터들은 특별한 가공없이 데이터 자체가 의미를 지니는 파라메터들이다. 즉, 문자로 구성되어진 데이터들이나, 전화번호와 같이 숫자의 형태를 지니지만 그 자체로 의미를 가진 것들을 예로 들 수 있다.

액터 온톨로지는 서비스에 참여하는 주체자들을 정의 한 것으로 그림5와 같다 쇼핑몰에서의 액터는 소비자, 판매자, 상품 공급자, 서버 등이 될 수 있다.

human 범주는 서비스에 참여하는 개인을 의미한다. 앞서 살펴 본 예에서는 개개인의 소비자나 판매자가 될 수 있다. technology 범주는 기술들을 의미하는데, 인증서버 등이 될 수 있다. organ ization 범주는 개인이 모여 하나의 조직으로 분류되는 액터들을 의미하는데 기업에서 상품

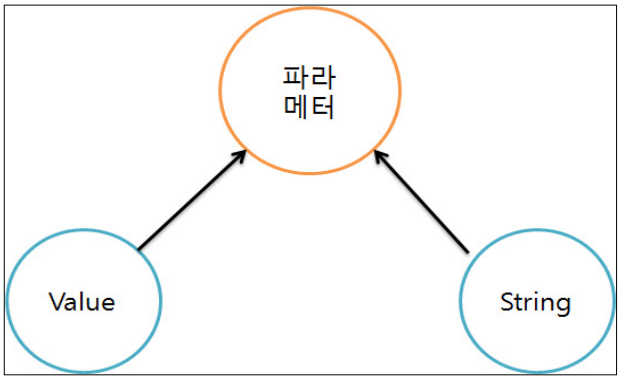

그림 4 파라메터 온톨로지 구성

을 구매하는 경우 기업이나, 공급자가 개개인이 아니라 기업이면 공급자 같은 경우이다.

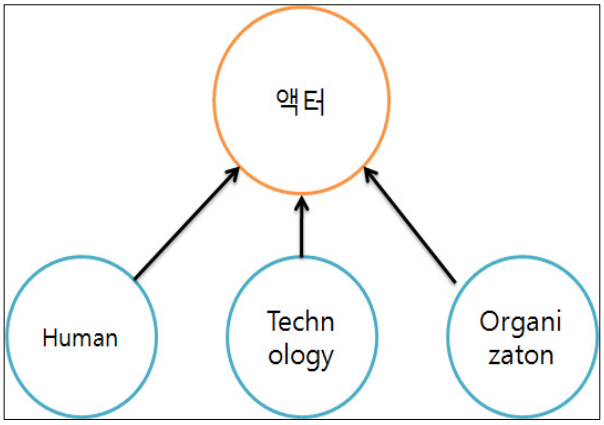

그림 5 액터 온톨로지 구성요소

프로세스 온톨로지는 그림6 과 같이 심플 프로세스와 콤플렉스 프로세스로 구분 할 수 있는데, 콤플렉스 프로세스는 또 다른 콤플렉스 프로세스와 심플 프로세스의 조합으로 표현 할 수 있다. 심플 프로세스와 콤플렉스 프로세스의 구성은 각각 그림7,8 과 같이 구성 된다.

프로세스는 서비스를 구분하는 가장 중요한 요인이므로 각각의 프로세스에 대한 정의가 매우 중요한다. 그 정의는 [8]에서 구분하여 정의하고 있다.

단계1-3 메타 온톨로지와 연결

구축한 도메인 온톨로지를 메타 온톨로지와 연결하는데 이는 앞서 설명하였듯이 도메인에 국한되지 않고 다양한 분야의 도메인의 서비스들을 이용할 수 있게 해주는 장점이 있다. 이 단계의 산출물은 도메인 온톨로지와 메타 온톨로지와의 통합인데, 향후 서비스를 정의하거나 검색할 때 이용하게 된다.

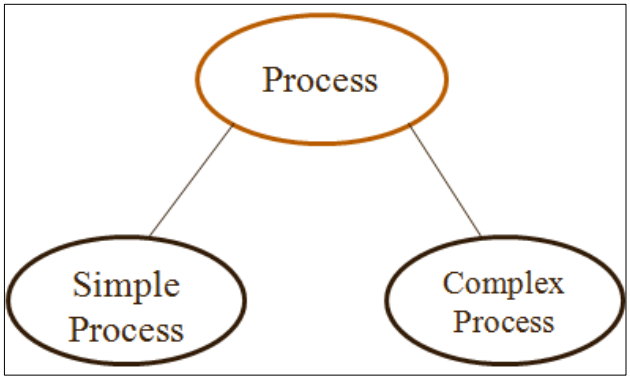

그림 6 프로세스 온톨로지 구성

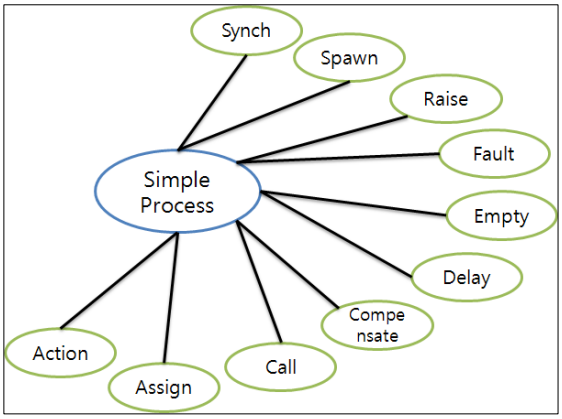

그림 7 심플 프로세스 온톨로지 구성

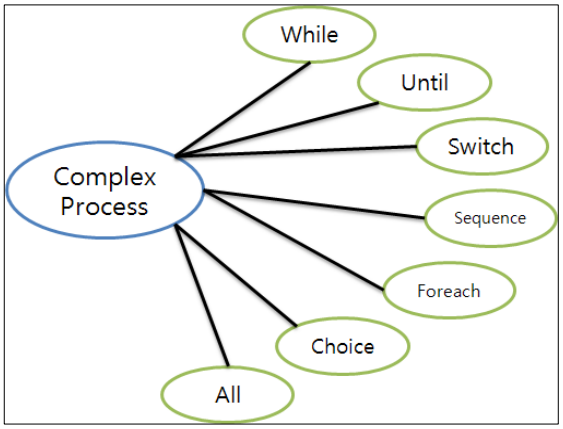

그림 8 콤플렉스 프로세스 온톨로지 구성

단계 2 요구사항분석

이 단계에서는 서비스의 참여자를 식별하여 참여자들의 요구사항을 분석하고 명세화 하는 단계이다. 시스템 개발에서 요구사항은 분석은 시스템의 전체적인 개발 방향과 밀접하게 연관되어 있으므로 매우 중요한 단계이다.

쇼핑몰의 경우 적용해보면 쇼핑몰을 이용하는 사람과 쇼핑몰을 구축하는 사람, 쇼핑몰에서 판매하는 사람 등 쇼핑몰과 연관되어 있는 모든 사람을 식별하고 그 사람들의 요구사항을 빠짐없이 조사하여 기술한 다음 각각을 유즈케이스로 전환하여 명세화 한다.

단계 2-1 이해 당사자 식별

시스템과 연관되어 있는 모든 이해당사자자를 파악한다. 이 때 파악된 이해당사자들의 시스템의 참여도와 중요도에 따라 순위를 부여하는데 이 우선 순위는 요구사항들 간의 상충관계가 발

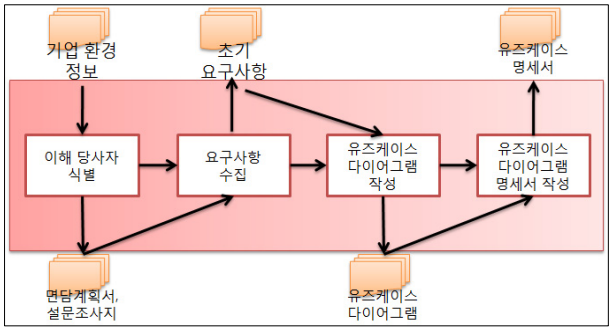

그림 9 요구사항 분석 단계 프로세스

생할 경우 이용하게 된다.

이 단계에서는 식별된 이해당사자들로부터 요구사항을 수집하기 위한 계획 수립도 포함되는데 이해 당사자 별로 면담계획을 세워 면담일정을 정하거나 설문조사를 실시 경우 설문 조사지를 만든다.

예를 들어 쇼핑몰의 경우 판매자, 소비자, 상품공급자를 이해 당사자로 꼽았다면 이 사람들에게서 요구 사항을 어떻게 수집 할 지를 결정해야 한다. 판매자와 상품공급자의 경우 대상이 한정된 경우가 일반적이므로 면담을 통해서 요구사항을 수집할 수 있지만, 소비자는 일반인을 대상으로 하는 경우가 많으므로 모두 면담을 하는 것이 불가능 할 수 있다. 따라서 이들에 대해서는 설문조사를 이용한다던지 일반인을 대표할 수 있는 대표군을 만들어 면담을 하는 방법이 필요한데, 이에 대한 계획을 이 단계에서 모두 결정해야 한다.

단계 2-2 요구사항수집

이전 단계 식별한 이해당사자별 요구사항 수집 계획에 따라 이해 당사자들로부터 요구사항을 수집하는데, 이 단계에서 식별된 요구사항은 시스템의 개발의 방향을 결정하게 됨으로 시스템을 전체적인 관점에서 이해 할 수 있도록 목표 시스템에 관련된 정보를 상세히 기술한다. 또한 수집된 요구사항은 세부적인 사항까지 모두 포함하여 요구사항 명세서로 만든다.

단계 2-3 유즈케이스 다이어그램 작성

수집된 요구사항을 기능별로 유즈케이스 다이어그램을 작성한다. 유즈케이스 다이어그램을 통

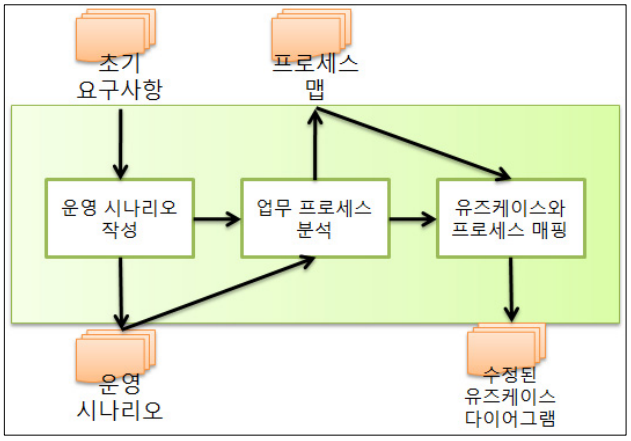

그림 10 비즈니스 모델링 프로세스

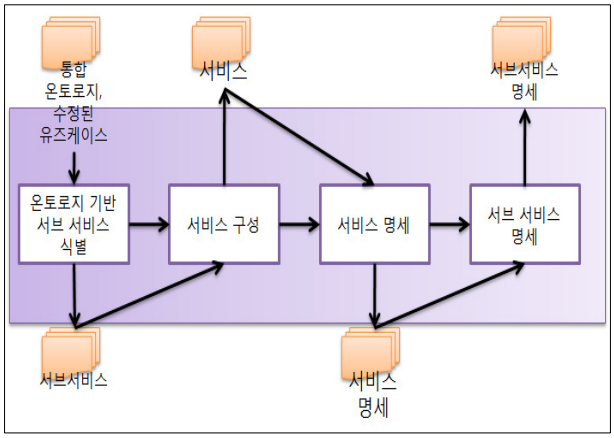

그림 11 서비스 식별 프로세스

하여 시스템에 참여하는 이행당사자들과 이들 요구한 사항들을 기술함으로써, 시스템 요구사항을 보다 명확히 정의할 수 있다.

단계 2-4 유즈케이스 다이어그램 명세서 작성

작성된 유즈케이스 별로 유즈케이스 명세서를 작성하게 되는데 기능을 구현하기 위해 진행되는 절차들을 세부적으로 기술하고 예외 상황에 대해 자세히 기술한다.

또한 앞서 수집한 요구사항들을 정략적으로 변환하여 유즈케이스 다이어어그램 명세서에 기입한다. 이는 서비스 명세를 만들 때, 이를 기준으로 서비스 구성을 하기 때문에 반드시 정량화 될 수 있는 것들은 정량화시켜 기입해야 한다.

예를 들어 쇼핑몰에서 소비자가 "상품을 검색했을 때 빠르게 반응 했으면 좋겠다" 라고 했다면 "빠르게"라는 단어에 대해서 정량적으로 0.5초안에 응답해야한다 라고 기입해야 한다.

단계 3 비즈니스 모델링

지금까지 수집된 요구사항과 운영상의 요구사항들을 파악하여 개발하게 될 시스템에 대한 운영 시나리오를 작성한다. 이때 수집된 요구사항을 정제 하게 된다. 즉, 지금까지 수집한 요구사항과 개발하고자 하는 시스템의 운영상의 요구사항을 파악하여 개발하고자 하는 시스템의 프로세스의 흐름을 정의하는 단계이다.

단계 3-1 운영시나리오 작성

수집된 요구사항들을 분석하여 기술, 운영, 재무적인 관점에서 실현가능 여부를 파악하고 중복되는 요구사항들을 정제하며 요구사항간의 상충관계가 존재하는지를 파악하여 상충 관계 존재한다면 상충되는 요구사항을 조정한다. 요구사항 분석과 정제를 모두 완료하여 구현이 결정된 요구사항들을 포함하여 시스템에 대한 운영시나리오를 작성한다. 이때 요구사항 수집 당시 이해당사들의 언어로 기술 된 것들은 앞서 정의한 온톨로지 정의한 개념들로 바꿔 표현한다.

운영 시나리오는 발생가능한 모든 상황에 대해 작성하여 추후 시스템이 운영 될 때 예외 사항의 발생 가능성을 줄일 수 있도록 한다.

또한 이 단계에서 운영시나리오를 작성함으로써 요구사항 중에서 1차적으로 실행 불가능한 요구사항에 대해서 식별하게 된다.

단계 3-2 업무프로세스 분석

작성된 운영시나리오를 바탕으로 시스템의 업무 프로세스를 분석하여 상세한 시스템 프로세스를 파악한다. 파악된 프로세스는 IDEF 3 등을 이용하여 표현한다.

단계 3-3 유즈케이스와 프로세스 매핑

식별된 업무프로세스를 유즈케이스와 매핑 한다. 이는 요구사항이 프로세스상의 위치를 식별함으로써 기술적으로는 실현 가능할지라도 운영상의 문제점으로 인해 구현하지 못하는 요구사항을 식별 할 수 있게 한다. 또한 유즈케이스와 프로세스 매핑 결과는 서비스를 식별하고 조합할 때 이용하게 된다.

단계 4 서비스 식별

유즈케이스별로 온톨로지를 이용하여 서비스를 식별하여 서비스를 명세하고 서비스 명세의 결과를 이용하여 서브서비스를 명세한다. 이 단계가 끝나고 나면 1차적으로 서비스의 일부가 구성되며, 서비스의 식별이 가능해 진다.

단계 4-1 온톨로지 기반 서브서비스 식별

유즈케이스의 프로세스들을 온톨로지 기반으로 서브서비스를 식별한다. 앞서 예를 들었던 로그인의 서비스의 경우 사용자가 데이터를 입력하면 입력된 데이터와 서버에 저장된 사용자의 데이터를 비교하여 로그인을 승인하거나 거절하는 세 단계 서브서비스들로 구성되는데, 사용자가 데이터를 입력하는 부분, 입력된 데이터와 서버에 저장된 사용자의 데이터를 비교하는 부분, 로그인을 승인하거나 거절하는 부분으로 구분하여 프로세스 온톨로지에서 정의한 요소들로 표현한다. 로그인 서비스의 경우 데이터를 입력받는 assign 프로세스, 데이터를 비교하는 action, 로그인을 거절하거나 승인하는 choice의 프로세스로 구분할 수 있으며 각각 하나의 서브 서비스가 된다.

단계 4-2 서비스 구성

식별된 서브서비스들로 모든 유즈케이스에 대한 서비스를 구성한다. 구성된 서비스를 요구사항별로 정렬함으로써 앞에서 정의하였던 서비스 구성요소들 중에서 파라메터. 액터. 프로세스와 서비스 수준에 대해서 정의하게 된다.

이 단계가 끝나면 본 논문에서 제안하였던 서비스 구성 4가지 요소가 정의되며 서비스 식별이 가능하다.

단계 4-3 서비스 명세

서비스 구성단계에서 정의된 요소들에 대해서 자세히 기술하여 서비스에 대한 명세를 만드는 단계이다. 특히 이 단계에서 정량적인 서비스 수준에 대한 명확한 정의는 다음 프로세스인 서브서비스 명세의 기준으로 활용된다.

예를 들어 상품의 검색 결과는 "사용자가 검색을 클릭한 후부터 상품의 정보가 모두 표현되기까지의 응답시간은 3초 이내로 응답된다." 의 형

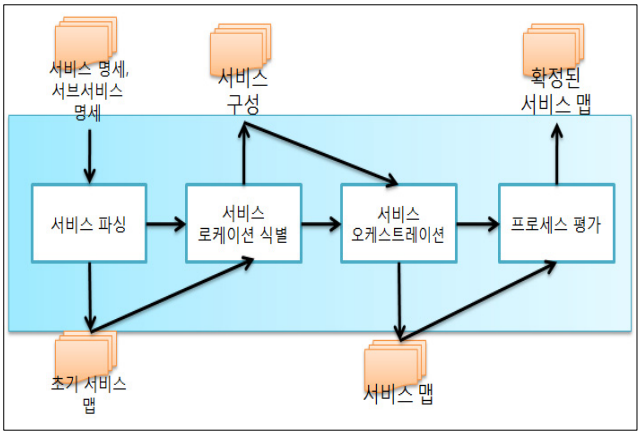

그림 12 서비스 조합 프로세스

태로 정량적인 수치와 그 범위까지 모두 기술해야 한다.

단계 4-4 서브서비스 명세

서비스 명세단계에서 정의된 서비스 수준을 기준으로 서브서비스 명세한다. 유즈케이스 별로 서브서비스에 대한 서비스 수준을 결정하는 단계이다.

단계 5 서비스 조합

통합 온톨로지에서 정의한 언어를 이용하여 유즈케이스 별로 정의한 서비스 구성의 서브서비스를 검색한다. 검색된 결과 중 재사용 여부는 앞서 명세한 서비스 명세와 서브서비스 명세 서비스 수준에 의해서 결정된다. 즉, 서비스를 검색하여 서브서비스 명세의 기준과 일치하는 서비스들을 조합하여 프로세스를 테스트한 뒤, 요구사항을 만족하는 검증을 하는 단계이다.

예를 들면 쇼핑몰에서 식별된 모든 서비스에 대해서 검색을 하고 쇼핑몰에 구성되는 서비스들의 서브서비스의 명세 기준에 적합한 서비스를 찾아 조합하였을 때, 실제 프로세스를 테스트 했을 때 1초 이내 응답하는지 검증하고 테스트 하는 것이다.

단계 5-1 서비스 파싱

검색된 서브서비스들을 서비스 구성에 따라 조합하는 단계이다. 식별된 서비스들의 서브서비스 명세에 따라 재사용할 서브서비스들을 검색하여 서브서비스 명세에 따라 재사용을 결정하게 된다.

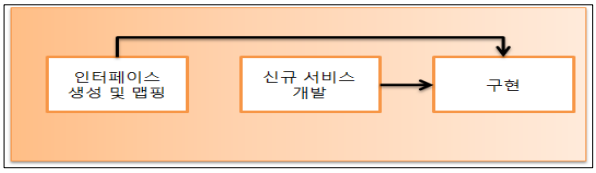

그림 13 구현 프로세스

단계 5-2 서비스 로케이션 식별

서비스 파싱 단계에서 재사용이 결정된 서비스들에 대하여 서비스의 위치 정보와 기반 기술들을 구성한다. 이 단계가 끝나고 나면 서비스 구성요소에 대한 정보를 모두 식별할 수 있게 되며 이때 비로써 완벽한 서비스 식별이 된다.

단계 5-3 서비스 오케스트레이션

업무 프로세스 분석 단계에서 작성한 프로세스 맵을 서비스 맵으로 전환하는 단계이다. 프로세스 맵에서 식별된 서비스들로 치환하고 재사용하는 서비스에 대한 위치 정보를 표시한다.

또한 재사용할 서비스를 찾지 못한 서비스를 식별하여 개발 서비스로 분류한다. 이 때 개발할 서비스에 대한 서비스 명세를 표시하여 개발할 때 참고하도록 한다.

단계 5-4 프로세스 평가

구성된 서비스 맵에 대해서 성능을 평가하여 초기에 정의한 사용자 요구사항과 일치하는지를 파악한다. 이 때 서비스 실행시간, 가용성, 품질 등 모든 부분에 대해서 성능 평가를 하고, 요구사항을 만족시키지 못하는 기능에 대해서는 서비스 파싱 단계로 리턴하여 새로운 서브서비스들로 구성한다.

단계 6 구현

지금까지 식별된 서비스들과 새롭게 개발 서비스들을 구현하는 단계이다. 이때 확정된 서비스 맵을 기준으로 구현 하게 된다. 최종 구현 단계까지 끝나게 되면 시스템을 테스트하고 운영 단계로 넘어가게 된다.

단계 6-1 인터페이스 생성 및 맵핑

재사용 결정된 서비스에 대해서 인터페이스를 생성하고 시스템의 파라메터, 액터와 맵핑한다.

단계 6-2 신규서비스 개발

새롭게 개발될 서비스에 대하여 서비스 명세와 서브서비스의 명세 등의 정보를 이용하여 구현한다. 이때, 프로세스는 온톨로지에서 정의한 개념들을 이용하여 정의하고 개발한 뒤 다른 서비스에서 이용할 수 있도록 한다.

단계 6-3 구현

새롭게 개발된 서비스와 재사용하는 서비스들을 모두 조합하여 최종 시스템을 구현한다.

2.3 제안하는 방법론의 평가

제안하는 방법론의 효용성에 대해서 평가를 위해 현재 S기업에서 사용하고 있는 CBD 방법론과 비교하여 평가하였다. 평가단은 S기업 근무자 중 10년 이상의 개발 경력을 가지고 있는 전문가들 5명으로 구성하였으며, 평가항목은 [9]에서 제시한 개발기간, 비용, 고객만족, 재사용성 4가지 항목에 대해 7점 척도로 평가 하였다. 또한 개발방법론을 대형, 중형, 소형 시스템 개발로 구분하여 각각을 평가 하였다.

평가 결과는 전문가 채점 결과의 평균을 비교 한 것과 [9]에서 제시한 AHP 가중치를 부여하여 계산 한 것, 두 가지 모두 실시하였다. 대형, 중형, 소형 시스템에 대한 평가의 결과는 차례로 표2, 3, 4와 같다.

	평균		가중 평균		
	CBD	제안하는 방법	가중치	CBD	제안하는 방법
개발기간	5.8	6.4	0.383	2.22	2.45
비용	5.2	5.4	0.206	1.07	1.07
고객만족	5	6	0.346	1.73	1.11
재사용성	5.2	6.2	0.066	0.34	0,41
합계	5.3	6		5.37	6.05

표 2 대형 시스템에서의 평가 결과

	평균		가중 평균		
	CBD	제안하는 방법	가중치	CBD	제안하는 방법
개발기간	4.4	5.6	0.383	1.69	2.14
비용	4	5	0.206	0.82	1.03
고객만족	4.2	5.8	0.346	1.45	2.01
재사용성	5.2	5.8	0.066	0.34	0.38
합계	4.45	5.55		4.31	5.56

표 3 중형 시스템에서의 평가 결과

	평균		가중 평균		
	CBD	제안하는 방법	가중치	CBD	제안하는 방법
개발기간	3.4	4	0.383	1.30	1.153
비용	3	3.4	0.206	0.62	0.70
고객만족	4.4	5.2	0.346	1.52	1.80
재사용성	4.4	4.6	0.066	0.29	0.30
합계	3.8	4.3		3.73	4.34

표 4 소형 시스템에서의 평가 결과

평가의 결과 본 방법론이 기존 CBD방법론 보다 대체로 더 좋은 것으로 나타났다. 또한 단순 산술 평균과 가중 평균의 차이의 결과가 거의 일치하게 나타났다.

기존 CBD 대비 제안하는 방법론의 효용성의 절대적인 수치는 대형 시스템 개발 시 가장 높은 것으로 나타났으며 중형, 소형시스템 순이었다. 상대적 효용성을 평가하기 위하여 수식 1과 같이 가중 평균에 계산을 하였을 때, 가중 평균으로 계산 하였을 경우 대형, 중형, 소형 시스템 순으로 12.7%, 29.2%, 16.1%으로 중형, 소형, 대형 시스템 순으로 상대적인 효용성이 높았다.

$$\frac{\text{제안하는방법론} - CBD}{CBD} \qquad \text{(수식 1)}$$

위 결과와 같이 대형 시스템에서의 절대적 가치가 더 큰 것은 최근 시스템이 복잡 다양해짐에 따라 이기종간의 연동의 필요성이 급증하고, 비즈니스 환경이 급변함에 따라 이기종간의 연동과 재사용성의 중요성이 커지고 있기 때문이다. 따라서 전통적인 구조적 방법론 보다는 CBD, SOA와 같은 재사용성을 추구하는 개발방법론을 대형 시스템에 적용하여 개발하는 것이 적합함을 보여주는 것으로 사료 된다.

3. 결론

SOA는 급변하는 비즈니스 환경에서 기업의 민첩하게 대응할 수 있게 해줄 수 있는 새로운 IT 전략의 대안으로, 기업의 기존 자산의 활용이나, 비용 절감 등의 효과를 줄 것으로 기대를 모으고 있다.

그러나 서비스 기반의 개발을 위한 표준 방법론이 존재하지 않았다. 이에 본 논문에서는 온톨로지를 이용하여 서비스를 식별하고, 서비스를 기반으로 시스템을 개발하는 방법론을 제안하였다.

제안하는 방법론은 서비스의 6가지 구성요소 정의하고 3가지 요소에 대한 온톨로지를 구축하여 그 온톨로지를 기반으로 서비스를 식별하고 서브 서비스들로 구성함으로써 서비스를 만들었다. 제안하는 방법론을 기존의 CBD방법론과 효율성 측면에서 비교하였다.

본 방법론에서는 서비스 식별과 검색에 온톨로지를 적용함으로써 서비스의 유연성을 높여 재사용성을 향상 시킬 수 있는 방법을 제시하였는데, 이는 서비스에 대한 의미론적 접근에 대한 방향성을 제공 할 수 있을 것으로 기대된다.

[참고문헌]

[1] 한국 developerWorks 운영팀, SOA 기술자료 특집

[2] 신수미, SOA 기반의 정보시스템 설계 및 구현, 한국콘텐츠학회 2007 춘계 종합학술대회 논문집

[3] 조병욱, What is SOA? How to SOA? 아키텍처 모델 구현과 수행 방법론, 한국마이크로소프트, 2003년 3월호

[4]W Schulte Yefim, V. Natis, service oriented architecture part1

[5] 엄창현, 조직모델과 프로세스의 온톨로지 정의

[6] Korotkiy, M, Top, J., From Ontology-enabled SOA to Service-enabled Ontologies, Telecomunications, 2006

[7] SOA(Service-Oriented Architecture) 관련 기술 동향과 전망, 정보통신연구진흥원 학술정보

[8] Arkin, A.; *Business Process Modeling Language, BPMI.org, 2002*

[9] 송정환, 조재훈, 김채수, AHP를 이용한 SOA 평가방법론, 한국산업경영시스템학회, 2006