

유연한 비즈니스를 위한 유연한 IT 환경 - SOA

한국IBM 서경기

1. 서론

최근 비즈니스 환경에 있어 가장 중요시되고 있는 요소 중 하나가 변화에 대한 대응력을 갖추는 것이다. 과거에도 많은 조직들이 변화를 강조해 왔고 이러한 변화에 대응하기 위해서 많은 노력을 해온 것은 사실이다. 그러면 현재 이야기 되고 있는 변화와 과거의 변화는 어떤 차이가 있는가? 한마디로 이야기하면 유연성과 속도라고 말할 수 있다.

과거에는 변화를 미리 예측하고 이에 따라 비즈니스 형태를 변화시켜왔다면 현재에는 이러한 변화를 예측하기가 매우 어렵다. 고객의 행동패턴도 과거와는 아주 다르고 자주 바뀔 뿐만 아니라 경쟁도 과거와는 비교할 수 없을 정도로 치열하다. 또한 정부의 규제도 강화되고 조직간의 인수합병도 활발하기 때문에 과거와 같이 미리 예측하고 이에 대응하는 형태로는 현재의 변화의 속도를 따라갈 수 없다. 또한 지금 같은 경쟁 상황에서 서로 비슷한 상품, 비슷한 서비스를 제공하고 있기 때문에 경쟁업체와의 차별성을 보여주기 위해 매우 힘이 든다.

예측할 수 없는 비즈니스 환경에서 갑작스러운 변화에 잘 대응할 수 있도록 해주는 비즈니스의 유연성과 변화가 생기면 다른 조직보다 빠르게 대응할 수 있는 속도는 이러한 환경에서 다른 조직과의 차별성을 보여줄 수 있는 유일한 수단이라 말할 수 있다.

유연성과 속도는 사실 서로 떨어져서 생각할 수 없다. 유연성이 없으면 속도를 낼 수 없다. 현재의 많은 변수를 미리 예측하여 거기에 맞추어 비즈니스 모델을 변경할 수 없으므로 어떤 변화가 있더라도 수용할 수 있는 비즈니스 체질을 갖추어 놓고 변화가 감지되었을 때 이를 반영하여 비즈니스에 반영할 수 있도록 한다면 그렇지 못한 조직과는 차이가 날수 밖에 없다.

비즈니스의 유연성과 속도를 위해서는 IT 시스템의 유연성이 필수적이다. 그러나 기존 IT 시스템은 이러한 비즈니스의 유연성과 속도의 필요성을 충족시키기

에는 한계가 있다. 빠르게 변화되는 환경변화에 대응하기 위해서는 IT 시스템은 아주 경직된 특성을 가지고 있어 변화하기가 쉽지 않다.

고객의 욕구 변화나 새로운 경쟁자의 등장과 같은 환경 변화가 나타났을 때 기존 IT 시스템의 경우 이러한 변화를 분석하여 어떻게 IT 시스템에 반영할지 설계하고 개발하고 적용하게 된다. 과거에는 이러한 방식을 사용하여도 문제가 없었지만 오늘날과 같이 변화의 속도가 빠를 때 전통적인 방식을 사용하여 IT 시스템을 변경하면 완료되는 시점에서는 이미 또 다른 변화가 있어서 소용이 없을 수 있다. 즉 IT 시스템의 변화가 너무 늦어서 비즈니스 환경 변화를 따라가지 못하게 된다.

따라서 전통적인 IT 시스템과는 다른 패러다임이 필요하고 그것이 서비스 지향 아키텍처(Service Oriented Architecture-SOA)이다. 서비스 지향 아키텍처는 변화에 적시에 적절히 대응할 수 있는 IT 시스템을 구축할 수 있도록 하는 개념이다. 이를 위해 조직 내부 프로세스, 어플리케이션들을 각각 '서비스'라는 기본적인 기능 단위로 나누고 이들 '서비스'를 연결하여 원하는 기능을 하도록 구성하였을 때 이 변화를 반영하여 서비스의 연결 구성을 변화시켜 새로운 기능을 제공하도록 쉽고

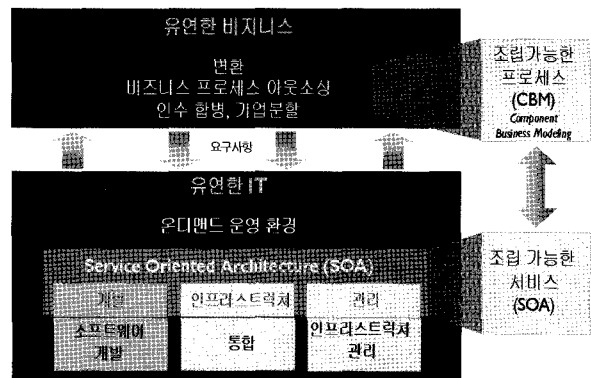


그림 1 유연한 비즈니스와 유연한 IT

빠르게 구성할 수 있도록 하는 것이다. 조직은 SOA를 통해서 유연한 IT 시스템을 구축하고 이를 통해 비즈니스 변화에 신속하고 적절하게 대응함으로써 조직의 경쟁력을 높이고 성장할 수 있는 기반을 마련할 수 있다.

2. 서비스지향 아키텍처(SOA)의 중요 개념

서비스 지향 아키텍처에서 이야기 하고 있는 “서비스”는 “반복사용이 가능한 비즈니스 기능”으로 정의할 수 있다. 서비스는 개별 기능을 수행하는 단위로서 다른 서비스와는 독립적으로 정의된다.

“서비스 지향”이란 비즈니스를 서로 연결된 서비스와 연결로 인해 발생하는 결과물로 통합하는 방식을 말한다. 다시 말하면 내부 비즈니스를 서비스로 정의하고 정의된 서비스를 서로 연결하여 산출물을 내는 방식이다.

따라서 서비스 지향 아키텍처(SOA)는 “서비스 지향” 비즈니스를 지원하는 IT 아키텍처 방식이다. 그리고 이러한 아키텍처 상에서 운영되는 애플리케이션을 “컴포지트(composite) 애플리케이션”이라고 하는데, 컴포지트 애플리케이션은 기존 애플리케이션 개발 방식과는 달리 정의된 서비스를 비즈니스 요구사항에 맞추어 조립(Composite)하여 원하는 기능을 구현한다. 만약 기존 컴포지트 애플리케이션을 변경해야 한다면 컴포지트 애플리케이션을 구성하는 서비스 중 해당 서비스를 빼고 원하는 서비스를 넣어서 새로운 기능을 하도록 변경할 수 있다. 따라서 매우 빠르게 애플리케이션을 만들어 낼 수 있다.

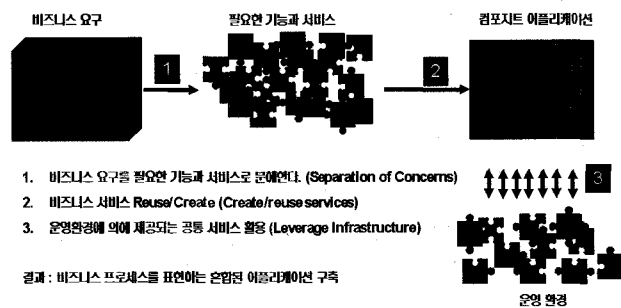


그림 2 컴포지트 애플리케이션

그림 2에서 보면 컴포지트 애플리케이션은 비즈니스 요구를 분해해서 필요한 기능을 조립해서 만들어진다는 것을 볼 수 있다. 이들 서비스는 각각 업무 시스템의 특정 시스템을 통해 구현된다. 이들 업무 시스템은 기존 시스템이 될 수도 있고 새로 만들어진 시스템일 수 있다.

ESB(Enterprise Service Bus)는 정의된 서비스들이 서로 커뮤니케이션 하는 통로이다. ESB는 애플리케이션을 느슨한 결합(loosely coupled) 비즈니스 서비스로서 연결 및 중재(Broker and Mediate) 해주는 표준 메커니즘을 제공한다.

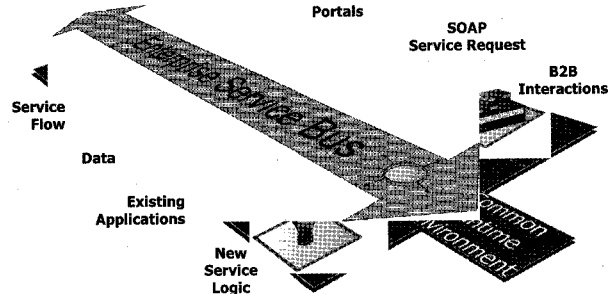


그림 3 ESB(Enterprise Service Bus)

3. 서비스 지향 아키텍처(SOA) 구현

서비스 지향 아키텍처(SOA)는 다음 4단계를 거쳐 구현된다.

- 1단계: 비즈니스의 컴포넌트화 및 서비스 모델링
- 2단계: 컴포넌트화된 비즈니스 프로세스를 서비스로 구현
- 3단계: ESB(Enterprise Service Bus)를 사용하여 서비스를 연결
- 4단계: 서비스 결합 프레임워크를 사용하여 서비스 통합

첫 번째 단계에서는 애플리케이션에서 수행되는 비즈니스 기능, 프로세스에서 사용되는 데이터, 시스템에서 제공되는 서비스 및 사람에 의해서 수행되는 과제 등의 기존 비즈니스 프로세스의 요소를 확인하고 이들 요소의 사용자를 파악한다.

이 단계에서 필히 비즈니스와 IT를 연계하여 프로세스를 파악하는 모델링이 수행되어야 하고 비즈니스 자체를 컴포넌트화 할 수 있는 방법론, 그리고 컴포넌트화된 프로세스를 서비스로까지 연결시킬 수 있는 방법론이 필요하다.

비즈니스 프로세스를 기업의 비전, 전략 바탕 하에 분석함으로써 해당 기업이 가장 역량을 집중할 수 있는 분야를 파악하고 IT와의 갭을 찾아내는 과정, 최종적으로는 컴포넌트화된 형태로 정의된 비즈니스 체계를 “Component Business Modeling(CBM)”이라는 방법론을 통해 도출할 수 있다. CBM 완료 후 실제 IT화 되어야 할 서비스 리스트를 도출하고 그 서비스들을 구성하는 컴포넌트 스펙까지를 “Service Oriented

Modeling and Architecture(SOMA)”라는 방법론을 통해 도출하게 되면 서비스의 구축을 위한 분석 단계가 끝나게 된다.

두 번째 단계에서는 이들 요소를 실행 가능한 서비스로 만든다. 서비스 사용자를 위한 클라이언트를 정의하고 웹 서비스의 WSDL(Web Services Description Language)을 사용하여 각 서비스의 인터페이스를 생성한다. 또한 각 서비스에 의해 사용되는 데이터 구조를 XML로 정의한다.

세 번째 단계에서는 ESB(Enterprise Service Bus)를 사용하여 위치, 전송형태 및 조직 범위에 상관 없이 서비스와 서비스 사용자를 연결한다.

마지막으로 이러한 기반 위에 각각의 서비스로 구현된 업무들을 통합한다.

4. SOA 라이프 사이클

구현된 서비스 지향 아키텍처(SOA)는 라이프 사이클을 가진다.

SOA 라이프 사이클은 모델(Model)→조립(Assemble)→운영(Deploy)→관리(Manage)의 절차를 거치게 되고 이 모든 단계를 관리/통제하는 거버넌스(Governance & Processes) 위에서 진행된다.

모델(Model) 단계에서는 비즈니스를 잘 아는 현업 담당자가 비즈니스 요건을 분석하여 어떤 서비스를 어떻게 조립할 것이고 그 흐름이 어떻게 되는지 정의하여 비즈니스 모델을 만든다. 기존 애플리케이션 개발은 운영을 해 보아야 애플리케이션의 문제를 파악할 수 있는 반면에 SOA 라이프 사이클에서는 애플리케이션을 IT 전문가가 개발하기 전에 현업 담당자가 정의한 비즈니스 모델을 미리 시뮬레이션 해서 개발 전에 미리 비즈니스 모델의 문제나 개선점을 파악하여 비즈니스 모델에 반영함으로써 문제 발생 가능성을 최소화 한다.

앞서 언급했던 CBM, SOMA가 비즈니스 프로세스 분석을 위한 모델링 단계라고 한다면, 실제 서비스 체제의 구현을 위한 모델링도 본 단계에 포함되어 수행된다.

조립(Assemble)단계에서는 모델 단계에서 만들어진 모델을 가지고 이를 실제로 운영될 수 있는 컴포지트 애플리케이션을 조립한다. 모델 단계에서 정의된 서비스와 프로세스를 기존의 시스템으로 연결하거나 신규로 작성하여 운영 가능하게 만든다.

운영(Deploy) 단계에서 작성된 컴포지트 애플리케이션을 운영한다.

관리(Manage) 단계에서는 운영되는 컴포지트 애플리케이션의 운영 상황을 실시간으로 모니터링하여 정

의된 서비스와 프로세스가 문제가 없는지 개선점이 없는지 판단한다. 만일 개선해야 할 것이 있다면 다시 모델(Model) 단계로 가서 작성된 모델을 수정한 후 다시 조립단계, 운영 단계를 거쳐 다시 모니터링 하는 단계를 순환하게 된다.

이 과정에서 전체 단계에 있어서의 거버넌스(Governance & Process)를 통해 관리/통제를 한다. 전체 프로세스를 제어하고 각각의 서비스에 대한 비용관리뿐만 아니라 서비스의 복잡한 서비스 사이의 관계를 정의하고 이들이 원활하게 운영될 수 있도록 통제한다.

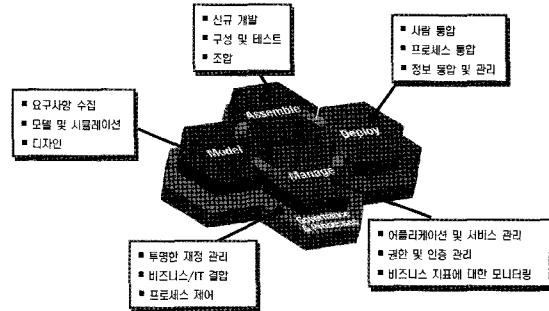


그림 4 SOA 라이프 사이클

5. 서비스 지향 아키텍처(SOA) 적용

서비스 지향 아키텍처를 적용하는 것은 각 조직마다 다르다. 어떤 조직은 전사적으로 적용하기도 하고 아니면 기존 시스템 중 일부를 SOA로 구현한 후 확장하는 방식을 취할 수 있다. 전체적으로 보면 각 조직에서 SOA를 적용하는 방식을 4가지로 나눌 수 있다.

첫 번째는 초기 적용 단계로서 특정 애플리케이션을 연결하는 것이다. 특징의 기존 혹은 신규 애플리케이션을 웹 서비스 기술을 이용해 연계한다. 두 번째는 여러 애플리케이션을 SOA를 통해 통합하는 단계이다. 비즈니스 프로세스상의 요구 충족을 위해 한정된 업무에 있어서의 복수의 애플리케이션 프로세스를 연결하고 제어한다. EAI를 구축 시 SOA를 기반으로 한다면 이 단계라고 할 수 있다. 세 번째는 전사적 단계로서 ESB 등에 의한 전사적인 SOA 인프라를 구축하여서 전사적으로 부서 경계를 넘은 서비스의 공유, 연계를 가능하게 한다.

마지막 단계로서 CBM등에 의해 도출된 새로운 비즈니스 모델을 수행하기 위한 조직을 넘은 가치 사슬 전체의 비즈니스 최적화를 목표로 한다.

6. 서비스 지향 아키텍처(SOA)의 가치

SOA는 유연한 비즈니스를 가능하게 하는 유연한 IT를 구현할 수 있게 한다. SOA의 특징을 보면 다음

과 같다.

- **유연성**: 서비스 컴포넌트 단위의 조립, 재조합이 간단
- **재사용성**: 공통으로 이용할 수 있는 서비스 컴포넌트를 만들고 기존 시스템을 서비스 컴포넌트화
- **확장성, 통합성**: 조직 및 회사 전체에 걸쳐 시스템 어플리케이션 간의 데이터 연계가 용이하고 업무 프로세스간의 제어를 자동화
- **감시, 모니터링**: 프로세스 이벤트의 모니터링 및 이벤트 취득을 위한 표준 기반

이러한 특징을 가지는 SOA는 현업 측면에서 보면 시장 변화에 즉시 대응할 수 있도록 하고 비용을 절감하면 조직의 핵심역량에 집중할 수 있게 한다. 또한 IT 측면에서 보면 개발기간을 단축하고 운영을 단순화하여 비용을 절감할 수 있게 한다. 또한 기존 시스템을 활용할 수 있는 방안을 제시한다.

7. IBM과 서비스 지향 아키텍처(SOA)

단순히 SOA 라이프 사이클 상의 각 단계에서 필요한 솔루션을 가진다고 해서 SOA를 구현할 수 있는 것은 아니다. SOA를 구현하기 위해서는 서비스 정의단계, 작성 단계, 운영 및 관리 단계에 있어서 적절한 가이드와 지침이 필요하다.

IBM은 SOA를 구현하는데 필요한 SOA 라이프 사이클 상의 모든 단계에서 필요한 솔루션을 제공할 뿐만 아니라 "SOA 인더스트리 엑셀러레이터"를 통해서 산업별로 특화된 프로세스참조 모델과 컴포넌트 모델과 프로세스 플로우에 대한 참조 모델을 제공하여, SOA의 계획에서 구현과정에서 발생 가능한 오류를 최소화하고, 프로젝트의 성공확률을 높인다.

또한 CBM, SOMA와 같은 SOA 방법론을 통해 비즈니스 컴포넌트에 기반하여 서비스를 정의하여서 조직이 시행착오를 최소화 하도록 하면서 SOA를 구현할 수 있도록 지원한다.

그리고 글로벌 ISV 파트너가 IBM의 지원을 통해 솔루션을 SOA 기반으로 전환할 수 있도록 하는 SOA 파트너 커뮤니티 정책을 가지고 있다.

IBM은 SOA와 관련된 솔루션, 서비스 및 파트너를 통해서 조직이 성공적으로 SOA를 적용할 수 있도록 돕고 있다.

8. 전자정부와 서비스 지향 아키텍처(SOA)

대한민국 정부는 2001년 5월 "전자정부 구현전략"을 수립하고 2003년 8월에는 "참여정부의 전자정부 로드

맵"을 수립하여 추진해 오고 있다. 이를 통해 문서처리 전 과정 전자화, 전자지방정부 구현, 전자국회 구현, 행정정보 공유 확대 등의 31대 중점과제를 추진 중에 있다.

궁극적으로는 전자정부의 수준을 부처간/기관간 경계가 없는 온라인 서비스 제공을 목표로 설정하고 있으며 지난 20여년 간 전자정부 구현을 위해 지속적으로 노력해 온 결과 한국의 전자정부 수준은 세계 선두권으로 평가받고 있다.

이러한 전자정부 수립과정에서 SOA와 관련된 주제를 찾아 볼 수 있다. 즉, 동일기능 시스템의 중복 개발을 배제하고 이를 위한 최적의 설계 방법을 적용함은 물론 필수 기능들은 반드시 공통 시스템의 개발로 활용도를 높여 유사 서비스의 공동개발/활용을 가능토록 하려는 노력이 그것이다. 따라서 서비스 재활용에 대한 인식을 제고하여 시스템 개발/운영이 업무요건의 변화에 능동적으로 대처할 수 있도록 하며 이를 사용자 만족도를 높이기 위한 프로세스의 단순화 노력, 시스템 간의 유기적인 연계를 통한 전체 프로세스의 재정립 차원으로 향후 전자정부사업이 펼쳐질 것으로 예상된다.

전자정부 웹서비스 도입 전략의 로드맵에 의하면 2006년 이후 이러한 문제점을 SOA에 기반한 신규 공공정보화 사업의 추진으로 해결하고자 하는 노력을 펼칠 예정이다. 즉 단순히 차세대 기반의 응용기술을 채택하는 것만으로 전자정부의 수준을 높이는 것 뿐만 아니라 개발방법론, 도입절차, 정부조직 및 프로세스, 활용체계에 대한 보다 심도 있는 고민을 통해, SOA 기반의 비즈니스 서비스 구축, 검증, 지원 프로세스를 수립해야 함을 인지했다고 할 수 있다. 이를 위해 향후 전자정부 추진 과정에는 SOA 아키텍처 하의 서비스 재사용성을 높이기 위한 서비스 모니터링 및 측정용 통해 정부차원의 서비스 관리 프로세스를 정립할 것으로 예상된다.

서 경 기



1992 서강대학교 경제학과(학사)
 1992 두산그룹 근무
 1996~현재 한국IBM 재직 중
 관심분야 : EA, BPM, SOA, OOAD,
 Application Development
 E-mail : kkseo@kr.ibm.com