

BizTalk에 기반한 ebXML 비즈니스 프로세스 변환기의 개발

황성진, 박경환
동아대학교 컴퓨터공학과

Development of the Translator for an ebXML Business Process based on BizTalk

Sung-Jin Hwang, Kyung-Hwan Park
Dept. of Computer Engineering, Donga-A University

요 약

ebXML 비즈니스 프로세스(Business Process)는 거래의 흐름을 정의하고 거래의 흐름을 주도 하는데 사용되어지며 BPSS(Business Process Specification Schema)에 의해 UML과 XML 두 가지로 표현된다. 그리고 ebXML 비즈니스 프로세스는 메시징 서버와 연동하여 거래 흐름에 따라 실제 데이터를 주고 받게되는데, 현재 ebXML 비즈니스 프로세스를 완전히 지원하는 메시징 서버는 거의 없다. 대개 XML/EDI에 기반을 둔 메시징 서버들이 많기 때문에 이러한 메시징 서버들은 ebXML의 TRP(Transport, Routing & Packaging)는 지원하지만 각 솔루션들의 비즈니스 프로세스 형태가 만드는 회사마다 다르기 때문에 ebXML 비즈니스 프로세스와 연동을 하기 위해서는 변환과정이 필요하게 된다. 본 논문에서는 이러한 문제를 해결할 수 있는 비즈니스 프로세스 변환기를 개발하는 것이며 메시징 서버 중 비즈니스 프로세스가 XML 형태로 생성되는 Microsoft BizTalk Server 2000을 기반으로 할 것이다.

1. 서론

전통적인 전자상거래의 표준은 EDI이며 지금도 많은 곳에서 EDI를 사용하고 있다. EDI는 기업 간 거래 업무 문서를 기계가 직접 읽고 처리할 수 있도록 코드화한 표준 메시지를 거래 당사자간에 합의된 규약에 따라 전용 회선을 통해 상호 전달하는 방식을 말한다. EDI는 거래업무에 필요한 문서처리시간을 단축시키면서 거래업무에서 생겨나는 오류와 비용을 감소시키고 전용 회선을 사용하기 때문에 빠른 문서 전달과 높은 신뢰성을 가지고 있다.

그러나 EDI는 값비싼 서버와 전용선 사용료를 VAN 업체에게 지불해야 하므로 초기설치 비용과 운영비용이 많이 든다. 그리고 EDI 문서 형식이 기계가 읽고 처리할 수 있는 코드로 되어있어서 높은 기술력이 필요하다. 그것뿐만 아니라 사용하는 문서 또는 프로세스가 변경될 경우 기존의 것을 사용할 수가 없어서 처음부터 다시 만들어야 하는 어려움이 있기 때문에 재사용성과 확장성이 저하된다. 또한 전용 회선을

사용함으로써 폐쇄적이기 때문에 타 업무와의 연계가 어렵다. 실제로 무역, 운송, 유통, 금융, 의료, 항만 등에서 EDI를 사용하고 있지만 특정한 작업 없이는 각각 EDI 망끼리 문서를 전달할 수 없다. 이러한 이유로 EDI는 자본이 많고 높은 기술력을 확보하고 있는 대기업이나 국가기관에서 주로 사용하고 있으며 Interactive EDI, Web EDI, Open EDI, 객체지향형 EDI 등과 같은 기존 EDI의 단점을 극복하기 위한 개선 방법이 생겨나게 되었다. 1990년대 인터넷이 보급되면서 기존의 웹 상에서 정보를 표현하는 언어가 HTML에서 XML 기반으로 바뀌게 되었다. 그것은 XML이 정보의 내용과 구조가 분리되어 있어서 정보를 검색하거나 표현하는데 있어서 HTML 보다 뛰어나기 때문이다. 그리고 또한 XML이 HTML보다 뛰어난 확장성과 재사용성을 가지고 있으며 읽기 용이함 때문이기도 했다. 이러한 XML의 장점을 살려 기존 EDI의 단점을 극복하려는 노력이 생겨나게 되었으며 XML/EDI, ebXML, BizTalk, RosettaNet, eCo, cXML 등이 있다[1,2,3].

XML/EDI 그룹에서 제안한 XML/EDI는 인터넷 기반의 EDI를 확장한 형태이며 메시지 교환형식으로 XML을 사용한다. XML/EDI는 EDI를 XML DTD나 스키마로 정의하여 기존의 EDI와 완전하게 호환이 되며 이것을 통해 기존 EDI의 구조적인 문제를 해결한다. 그리고 인터넷을 기반으로 하기 때문에 저렴한 운영비용이 드는 장점을 가지고 있다.

ebXML은 UN/CEFACT과 OASIS에서 주축이 되어 제정하고 있는 차세대 인터넷 전자상거래 표준 프레임워크이다. ebXML은 XML 기반으로 하며 "Creating A Single Global Electronic Market" 이라는 목적을 가지고 있으며 기존의 EDI의 폐쇄성을 극복하고 다른 산업들 간의 정보를 교환할 수 있도록 한다. ebXML도 기존의 EDI와 호환되며 지금 현재까지도 표준화가 부분적으로 진행되고 있다.

BizTalk 프레임워크는 1999년에 Microsoft에서 제안하였으며 XML 기반 문서뿐만 아니라 타 기반 문서를 처리할 수 있으며 이것을 위해 MIME (Mutipurpose Internet Mail Extensions) 기술을 사용하고 있다. 하지만 아직 완전히 W3C의 XML 스키마를 지원하고 있지 않다. 2001년에 BizTalk 프레임워크를 기반으로 한 Microsoft BizTalk Server 2000이 출시되었고 현재 2002 버전이 출시된 상태이다.

RosettaNet은 400여개 이상의 전자, 반도체, IT 산업의 업체들이 컨소시엄을 구성하여 1998년에 설립되었다. RosettaNet의 목적은 공통 비즈니스 인터페이스 만들어 개방적인 e-비즈니스 표준을 창조, 이행, 증진하는데 있다.

CommerceNet이 개발한 eCo 프레임워크는 XML 기반 애플리케이션 표준과 주요 전자상거래 환경 사이의 상호 운용을 위한 공통 프레임워크이다. 이를 위해 CommerceNet은 휴렛 팩커드, IBM, 선 마이크로 시스템, RosettaNet 등이 참여하는 eCo 프레임워크 프로젝트와 워킹 그룹을 구성하였다.

cXML(Commerce XML)은 아리바(Ariba), Sterling Commerce 등 B2B 전자상거래의 40여개 선도 업체들에 의해 개발된 표준으로써 1999년 5월에 처음 발표되었다. cXML은 XML 기반한 DTD와 각종 거래 정보의 교환을 위한 요구 및 응답을 정의하고 있다[4].

이들 프레임워크들 중 XML/EDI는 글로벌한 거래에 적용하기보다는 특정 기업들간의 거래에 적합하고 RosettaNet, BizTalk, CommerceNet, cXML 등은 특정 산업이나 상거래의 일부분만을 지원하므로 서로간의 연동성이 떨어지는 한계를 가지고 있다. 이에 반해

ebXML은 글로벌한 거래를 지원하므로 앞으로 국제적인 표준으로 자리잡을 것이다. 그리고 ebXML이 각 산업의 특성을 고려하여 만들어진 것이 아니라 기본적으로 공통적인 것으로 만들어 졌기 때문에 각 산업들의 고유한 특성을 살리기 힘들 수도 있다. 그러한 이유로 인해 ebXML은 다른 프레임워크와 연동하면서 발전해 나가고 있다[5].

ebXML 비즈니스 프로세스(Business Process)는 거래의 흐름을 주도하는데 사용되어지며 BPSS (Business Process Specification Schema)에 의해 UML과 XML 두 가지로 표현되어진다. 그리고 ebXML 비즈니스 프로세스는 메시징 서버와 연동하여 거래 흐름에 따라 실제 데이터를 주고 받게된다. 하지만 실제 ebXML 전자상거래 시스템을 구축하기 위해서 필요한 메시징 서버 중 완전히 ebXML 비즈니스 프로세스를 따르는 것이 거의 없는 실정이다. 그러므로 ebXML 비즈니스 프로세스를 지원하는 메시징 서버를 사용하지 않을 경우 ebXML 비즈니스 프로세스를 메시징 서버에 맞게 변환하는 과정이 필요하게 된다. 본 논문의 목적은 이러한 문제를 해결할 수 있는 비즈니스 프로세스 변환기를 개발하는 것이며 메시징 서버 중 비즈니스 프로세스가 XML 형태로 생성되어 ebXML 비즈니스 프로세스의 XML 버전으로 변환하기 쉬운 Microsoft BizTalk Server 2000을 기반으로 할 것이다.

2. ebXML 비즈니스 프로세스 명세 스키마

ebXML 비즈니스 프로세스는 BPSS(Business Process Specification Schema)로 정의되어 표현되어 지는데 BPSS는 비즈니스 시스템(비즈니스 거래로 구성된 비즈니스 협업을 수행 보조하는 것)에 바탕을 둔 표준 프레임워크를 제공한다. 기본적으로 UN/CEFACT 모델링 방법론(UMM)의 메타모델에 기초한다. BPSS의 목표는 e-비즈니스 프로세스 모델링과 e-비즈니스 소프트웨어 컴포넌트 명세간의 가교를 제공하는 것이다. e-비즈니스 프로세스 모델링은 UMM을 바탕으로 비즈니스 프로세스와 정보를 모델링하는 것을 가리킨다. 그리고 e-비즈니스 소프트웨어 컴포넌트 명세는 실제 e-비즈니스에서 거래를 실행할 소프트웨어 컴포넌트 명세를 가리킨다. BPSS는 e-비즈니스 프로세스 모델링을 실제 e-비즈니스 소프트웨어 컴포넌트 명세에서 활용할 수 있도록 변환하는 역할을 한다.

BPSS는 UML버전과 XML버전 등 두가지 독립적 표현을 가진다. UML버전은 단순히 UML 클래스 다이어그램이며, 이것은 단지 ebXML을 따르는 비즈니스 프로세스 명세를 생성하도록 하기 위해 필요한 모든 명세 엘리먼트와 관계를 표현한다. XML버전은 ebXML 비즈니스 프로세스 명세의 XML 기반 인스턴스를 제공하고, 다른 표현의 생성 규칙으로 제공된다. DTD와 W3C 스키마 모두가 제공된다. 또한 이 두 버전은 서로간에 명확하게 매핑된다[6, 7].

BPSS는 비즈니스 협업을 정의하는데 필요한 의미론, 엘리먼트, 그리고 특성을 제공한다. 비즈니스 협업은 비즈니스 문서를 교환함으로써 코레오그래피화된(순서화된) 거래 집합으로 구성된다.[5]

둘 혹은 그 이상의 비즈니스 파트너는 각각의 역할을 통해 비즈니스 협업에 참여하며 역할은 비즈니스 거래를 통해 상호작용을 한다. 비즈니스 거래는 코레오그래피에서 상호관계에 따라 순서화되어 있고, 각각의 비즈니스 거래는 하나 또는 둘의 미리 정의된 비즈니스 문서 흐름으로 구성되어 있다.

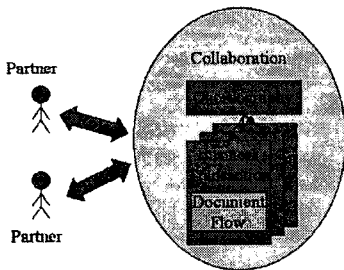


그림 1 비즈니스 협업의 의미론

3. BizTalk 오케스트레이션

Microsoft BizTalk server 2000에서는 비즈니스 프로세스를 오케스트레이션 디자이너를 사용하여 생성한다. 오케스트레이션 디자이너는 XLANG 스케줄 도면을 디자인하고 컴파일하여 구동시키는 것이다.

BizTalk 오케스트레이션은 다음과 같은 주요 개념에 기반을 두고 있다.[8, 9, 10]

(1) XLANG 언어

비즈니스 프로세스와 논리적인 순서를 기술하는

XML 기반의 언어이다.

(2) XLANG 스케줄 도면

비즈니스 프로세스를 서로간의 관계와 순서를 보여주는 하나의 그림이다. 도면은 Flowchart, Port, Implementation으로 이뤄져 있으며 완성된 도면은 .skv라는 확장자로 저장된다.

(3) XLANG 스케줄

XLANG 언어로 표현되는 특정 비즈니스 프로세스를 말한다. .skx라는 확장자의 파일로 저장된다. 이것이 XML 형태로 생성되는 비즈니스 프로세스라 하겠다. 본 논문에서는 XLANG 스케줄을 활용할 것이다.

(4) XLANG 스케줄 인스턴스

XLANG 스케줄의 특정 발생이다. XLANG 스케줄러 엔진은 한 개 이상의 XLANG 스케줄 인스턴스를 구동시킬 수 있다. 같은 XLANG 스케줄의 서로 다른 인스턴스들은 다른 메시지를 포함하고 있지만 같은 비즈니스 프로세스 규칙을 따른다.

(5) XLANG 스케줄 엔진

XLANG 스케줄 인스턴스를 실행하고, 조절하는 서비스이다.

4. 비즈니스 프로세스 변환기의 개발

본 논문에서는 ebXML 비즈니스 프로세스를 BizTalk 비즈니스 프로세스(XLANG 스케줄)로 변환시키는 변환기의 개발을 목적으로 하고 있으므로 두 개의 비즈니스 프로세스를 비교하여 변환에 필요한 부분을 추출하는 것이 중요하다. 그래서 각각의 비즈니스 프로세스를 특정 부분들로 세분화시켜 각 부분들의 관계를 알아보도록 하겠다.

우선 본 논문에서는 XLANG 스케줄을 3개의 부분으로 나누었다. 그것은 비즈니스 흐름을 보여주는 Flowchart 부분, 메시지의 입·출력을 나타내는 Implementation 부분, 그리고 이 두 부분을 연결하는 Port 부분이다. 이 부분들을 ebXML 비즈니스 프로세스와 비교하여 보면 Flowchart 부분은 ebXML 비즈니스 프로세스의 코레오그래피와 연결되고 Port 부분은 ebXML 비즈니스 프로세스의 문서 흐름과 일부분이 연결된다고 할 수 있다. 그러나 Implementation 부분은 ebXML 비즈니스 프로세스 상에서 구체적인 입·출력 방법을 정의해 놓지 않았으므로 본 논문에서는 BizTalk의 입력을 Message Queuing으로 하고, 출력을 Script Component으로 한정을 시켜서 변환기에서 자동으로 생성·설정이 되도록 한다.

위의 설명에서도 알 수 있듯이 XLANG 스케줄로

변환하는데 ebXML 비즈니스 프로세스의 모든 데이터가 필요하지 않다. 그것은 ebXML 비즈니스 프로세스가 BizTalk 비즈니스 프로세스보다 하이레벨이기 때문에 그만큼 표현의 범위가 넓기 때문이다.

(1) Flowchart 부분

BizTalk XLANG 스케줄에서 Flowchart에 해당하는 부분은 sequence 엘리먼트이다. Flowchart 부분은 비즈니스의 흐름을 정의한 부분이므로 ebXML의 코레오그래피, 비즈니스 트랜잭션과 그 역할이 흡사하다고 할 수 있다. 다음은 Flowchart 부분과 코레오그래피, 비즈니스 트랜잭션의 엘리먼트들을 비교해 놓은 것이다.

표 1 Flowchart 부분과 코레오그래피의 비교

BizTalk 비즈니스 프로세스 Flowchart 부분	ebXML 비즈니스 프로세스 코레오그래피
sink	RequestingBusinessActivity
source	RespondingBusinessActivity
없음(자동생성)	Start
진행	Success
zero	Failure
connect의 하위 sequence 엘리먼트에서 source	Join
connect의 하위 sequence 엘리먼트에서 sink	Fork

참고로 BinaryCollaboration 엘리먼트 하위에 코레오그래피 엘리먼트가 있고 BusinessTransaction 엘리먼트 하위에 RequestingBusinessActivity, RespondingBusinessActivity 엘리먼트들이 있다.

(2) Port 부분

BizTalk XLANG 스케줄에서 Port 부분은 header 엘리먼트의 하위 엘리먼트들과 binding 엘리먼트의

하위 엘리먼트 중 schemaList, messageDeclList 이다. Port 부분은 XLANG 스케줄 도면에서 Flowchart 도형과 Implementation 도형을 이어주는 부분인 Port를 가리킨다고 할 수 있다. 그렇기 때문에 Port 부분 일부가 Flowchart 부분이나 Implementation 부분에 포함된다. 또한 Port 부분은 들어오고 나가는 메시지의 대한 정보를 담고 있다. 그것은 메시지의 이름, 타입과 같은 정보들을 말한다. 그래서 Port 부분에서 필요한 정보들은 ebXML 비즈니스 프로세스의 BusinessDocument 엘리먼트에서 얻을 수 있다. BusinessDocument에는 요청·응답 메시지의 이름과 스키마 경로에 대한 정보를 가지고 있기 때문이다.

(3) Implementation 부분

Implementation 부분은 BizTalk XLANG 스케줄에서 portBindingList 엘리먼트의 하위 엘리먼트들이다. Implementation 부분은 입·출력 형식과 경로 정보를 가지고 있으며 포트와 메시지간의 연결 정보를 가지고 있다. 하지만 Implementation 부분에서 필요한 정보들은 대부분 ebXML 비즈니스 프로세스에서 얻을 수 없다. 그러므로 입·출력 형식과 경로 정보는 ebXML 비즈니스 프로세스에서 가지고 있지 않기 때문에 이 부분은 자동으로 생성할 것이다.

XLANG 스케줄의 portBinding 엘리먼트는 입·출력 방법과 입·출력 메시지에 대한 정보를 가지고 있다. 입·출력 방법에서 입력은 Message Queue를 사용하며 출력은 Script Component를 사용한다.

Message Queue의 이름과 Script Component의 이름은 포트의 이름을 따르며 이것은 XLANG 스케줄의 header 엘리먼트에서 가져오면 된다. 그리고 XLANG 스케줄의 portTranslation 엘리먼트를 통해서 실제 MSMQ와 Script Component 파일과의 연결한다.

그리고 또한 입·출력되는 메시지 이름도 XLANG 스케줄의 header 엘리먼트에서 가져오면 된다. 이렇게 가져온 것을 XLANG 스케줄의 messageTranslation 엘리먼트를 통해서 연결시켜주면 XLANG 스케줄의 portBinding 엘리먼트가 생성되면서 입·출력이 설정된다.

(4) 인터페이스

비즈니스 프로세스 변환기의 인터페이스 메뉴와 두 개의 창으로 이루어져 있다. File 메뉴에서 ebXML 비즈니스 프로세스를 불러오면 왼쪽 창에 ebXML 비즈니스 프로세스가 열린다.

Translation 메뉴에서 Microsoft BizTalk Server 2000을 클릭하면 BizTalk 비즈니스 프로세스(XLANG

스케줄)로 변환되면서 오른쪽 창에 나타나게 된다.

그림 2는 BizTalk 비즈니스 프로세스로 변환시킨 모습을 보여준다.

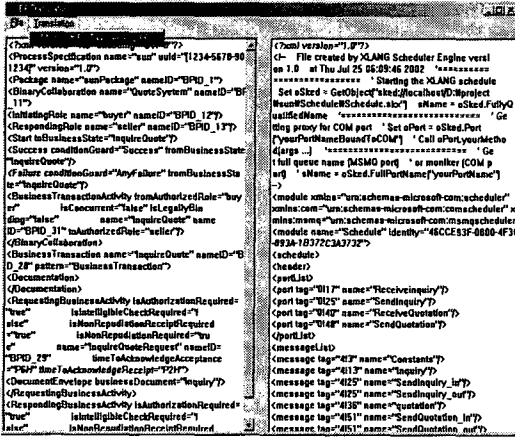


그림 2 비즈니스 프로세스 변환기에서 ebXML 비즈니스 프로세스를 BizTalk 비즈니스 프로세스로 변환한 화면

5. 결론

본 논문에서는 ebXML 비즈니스 프로세스를 BizTalk 비즈니스 프로세스(XLANG 스케줄)로 변환하는 변환기를 개발하였다.

이러한 비즈니스 프로세스 변환기를 개발함으로써 얻어지는 효과는 ebXML 기반의 전자상거래 시스템에서 기존의 XML 기반 메시징 서버를 그대로 활용할 수 있기 때문에 비용이 절감되고 비즈니스 프로세스를 잘 모르더라도 자동으로 변환되기 때문에 높은 기술이 필요 없고 시간이 절약되는 효과를 가져온다.

하지만 본 논문에서 개발한 비즈니스 프로세스 변환기는 에디팅 기능이 없으며 Microsoft BizTalk Server 2000으로만 변환되므로 다른 메시징 서버에는 적용시킬 수 없다. 이러한 문제는 향후 에디팅 기능을 넣고, 다른 메시징 서버의 비즈니스 프로세스로 변환할 수 있게 비즈니스 프로세스 변환기를 발전시켜 해결할 수 있다.

앞으로 ebXML이 완전히 전자상거래의 표준이 된다면 본 논문에서 개발한 비즈니스 프로세스 변환기의 유용성이 증대될 것이며, 이와 같은 목적을 가진 변환기들이 많이 개발될 때, 본 논문이 앞으로 개발될 비즈니스 프로세스 변환기의 개발을 위한 기반기술이

될 것으로 기대된다.

[참고문헌]

- [1] 김미영, 웹에 기반한 향만물류 XML/EDI 브라우저의 설계 및 구현, 동아대학교 컴퓨터공학과 일반대학원, 2000
- [2] 성명건, 황성진, 이석희, 박경환, ebXML에 기반한 향만물류 비즈니스프로세스 오케스트레이션, 한국멀티미디어학회 추계학술발표대회, 2002
- [3] 황성진, 박경환, BizTalk에 기반한 향만물류 XML EDI 시스템의 개발, 한국멀티미디어학회 추계학술발표대회, 2001.1
- [4] 정부연, 신일순, "XML을 통한 B2B 비즈니스 모델의 변화 및 시사점", http://www.foodmerce.com/jsp/fileupload/common_data/20011129173918411/view4.htm, 2001
- [5] KIEC, ebXML white paper, KIEC, 2002, pp. 14-76
- [6] ebXML Business Process Team, "ebXML Business Process Specification Schema v1.01," Technical report, 2001
- [7] David A. Chappell, Vivek Chopro, Jean-Jacques Dubray, Colleen Evans, Betty Harvey, Tim McGrath, Duane Nickull, Marcel Noordzij, Bruce Peat, Pim van der Eijk, Jan Vegt, Professional ebXML Foundations, WROX, 2002, pp.157-228
- [8] BizTalk Server 2000 개발팀, Microsoft BizTalk Server 2000 Documented, Microsoft Press, 2002, pp.406-576.
- [9] Robert Shimonski, Chris Farmer, Scott Roberts, Henk-Evert Sonder, Milton Todd, BizTalk Server 2000 Developer's Guide for .NET, Syngress, 2002
- [10] Brian E. Travis, XML and SOAP Programing for BizTalk Servers, Micosoft Press, 2000
- [11] Jon Duckett, Professional XML Schemas, Wrox, 2002