

제품 모델과 관련된 웹 표준의 동향

글 - 유상봉 - 인하대학교 컴퓨터공학부 - syoo@inha.ac.kr

본 연구에서는 ISO TC184/SC4에서 개발중인 제품 모델 표준의 웹 활용을 위한 관련 표준 동향을 조사한다. 연구의 대상은 다음과 같다.

- SC4의 IDW(Industrial Data on the Web) 그룹의 활동 현황
- Web Service 표준 개발 현황
- ebXML 표준 개발 현황
- RosettaNet 표준 개발 현황

1. SC4의 IDW 그룹의 활동 현황

제품모델의 국제규격인 STEP을 개발하는 ISO TC184/SC4에서는 2003년 3월 레졸루션을 통하여 IDW(Industrial Data on the Web) 그룹을 만들었다. 이 그룹의 주요 목적은 STEP을 비롯한 SC4 표준들을 인터넷에서 활용하기 위하여 필요한 기존 규격의 수정 또는 새로 필요한 규격을 제안하는 것이다. 현재 추진되고 있는 활동은 다음과 같다.

기존 규격을 Web에서 활용하기 위해

- STEP/SC4 파일들을 MIME 타입에 등록한다
- STEP/SC4 처리기들을 웹 브라우저의 플러그인으로 한다
- STEP/SC4 파일들 내부 또는 외부에서 URI로 참조할수 있도록 한다

다른 XML 표준과의 관계

- PDTnet
- MatML

새로운 웹 기술의 활용

- RDL과 다른 파트 라이브러리의 RDF/OWL 표현
- I5926의 RDF/OWL 버전
- STEP AP들의 RDF/OWL 버전
- Topic Maps

산업 데이터의 스프레드시트 표현

- PLIB 디셔너리 스프레드시트
- I5926 RDL 스프레드시트

현재 STEP 파일들에 의해 사용되고 있는 확장자는 다음과 같다.

*.stp	*.txt	*.203
*.STP	*.tx	*.AP210
*.step	*.htm	*.SEDRES
*.STEP	*.test	*.pdm
*.p21	*.log	*.pdm-schema
*.iai	*.vex	*.part21
*.ifc	*.ap203	*.Pr21

위와 같은 통일되지 않은 확장자들이 STEP 파일에

사용되어 응용프로그램의 연결이 원활하게 되지 않고 있다. 이러한 문제를 해결하기 위하여 step 파일의 표준화된 확장자를 정하여 MIME 테이블에 등록을 준비하고 있다.

2. Web Service 표준 개발 현황

웹서비스는 XML 기술을 기반으로 인터페이스와 바인딩을 정의, 기술, 검색할 수 있고 URI에 의하여 실행할 수 있는 소프트웨어 응용이며 인터넷 기반 프로토콜 위에서 XML 기반의 메시지를 이용하여 다른 소프트웨어 응용과 직접적으로 동작할 수 있는 소프트웨어 응용이다. W3C에서의 웹 서비스에 대한 표준 개발은 주로 구조 도메인 산하 웹 서비스 활동에서 그룹별로 이루어지며 다음과 같은 그룹이 존재한다.

- XML Protocol Working Group (2000년 9월 ~ 현재): SOAP 프로토콜 개발
- Web Services Description Working Group (2002년 1월 ~ 현재): WSDL 명세 개발

- Web Services Choreography Working Group (2003년 1월 ~ 현재): WS-CDL 명세 개발

Web Service 표준은 현재 WSDL, SOAP, UDDI로 구성되어 있으며 이들의 상호 관계는 그림 1과 같다.

우선 Web Service를 제공하는 서버측에서는 자신이 제공하는 인터페이스를 WSDL로 기술하여 UDDI Repository에 등록한다. 또한 웹서비스를 이용하려는 사용자들은 UDDI Operator를 통하여 자신이 원하는 서비스를 검색한다. 원하는 서비스를 찾은 후 해당 WSDL 파일을 통하여 필요한 오퍼레이션과 매개변수를 SOAP 메시지 형식에 맞추어 구성한다. 이렇게 구성된 메시지를 서버측에 전달하면 서버에서 처리하고 필요하면 응답 메시지를 역시 SOAP 형식에 맞추어 반환한다.

WSDL은 웹서비스를 기술하는 언어로서 서비스를 제공하는 서버측과 사용하는 사용자들이 이를 통하여 서비스에 대한 규격을 공유한다. WSDL의 기본 구성 요소는 다음과 같은 7개이다.

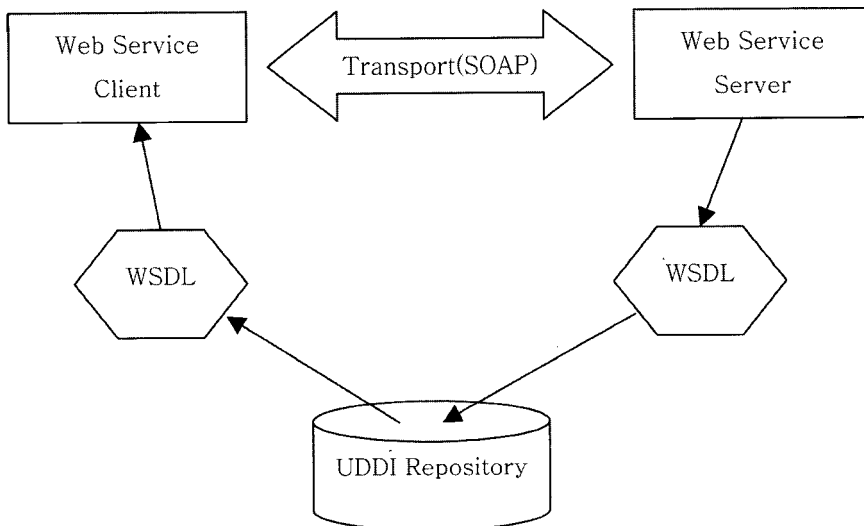


그림 1. Web Service 표준의 관계.

- **Data Type:** 메시지에서 사용될 데이터타입의 정의. XML 스키마를 이용할 수 있다.
- **Message:** 메소드 호출에 사용될 도큐먼트나 매개변수 형태로 정의된 추상적인 데이터 정의.
- **Operation:** 메소드의 이름, 메시지 큐, 또는 비즈니스 절차와 같은 오퍼레이션의 정의. 오퍼레이션들이 메시지를 입력받고 처리한다.
- **Port Type:** 엔드 포인트에 사상되는 오퍼레이션들의 추상적인 집합. 바인딩을 위한 오퍼레이션의 집합을 정의한다.
- **Binding:** 특정 포트 타입을 위하여 정의된 구체적인 프로토콜과 데이터 포맷을 정의한다.
- **Port:** 바인딩과 네트워크 주소의 결합으로서 서비스 통신의 최종 주소를 제공한다.
- **Service:** 하나 이상의 포트들을 포함하는 웹서비스 정의. 바인딩을 포트에 매핑하고 확장된 정의를 포함한다.

WSDL에서 정의되는 이러한 구성요소들 간의 관계는 그림 2와 같다.

SOAP는 분산 네트워크에서 정보를 교환하는 간단한 프로토콜로서 메시지에 있는 사항과 그 내용을 처리하는 방법을 설명하기 위해 프레임워크를 정의한 엔빌로프, 응용 프로그램에서 정의한 데이터 형식의 인스턴스를 나타내는 일련의 인코딩규칙, 원격 프로시저 호출 및 응답을 나타내는 규칙으로 구성된 XML 기반의 프로토콜이다. W3C에서는 웹서비스의 프로토콜로 HTTP, SMTP, FTP 등의 기타 프로토콜과의 바인딩에 대한 명세도 제공하고 있다. SOAP 0.9 스펙은 기존의 COM을 기초로 1999년에 처음 등장하였으며 1999년 11월에 1.0 버전, 2000년 4월에 1.1 버전, 현재는 1.2 버전이 발표되어 W3C의 표준 절차를 밟고 있다. SOAP은 헤더와 보디로 구성되며, 다음 예에서는 PDA, 핸드폰, cmail, 음성메일, 그리고 메시지에 전달될 간단한 메시지를 전달한다.

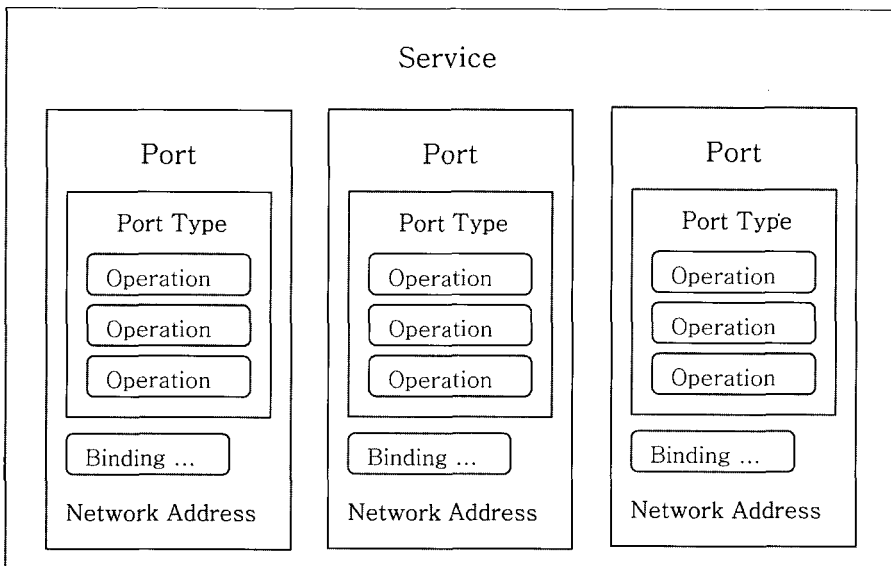


그림 2. WSDL의 구성요소.

```

<env:Envelope xmlns:env="http://www.w3.org/2001/12/soap-envelope">
  <env:Header>
    <n:broadcastService xmlns:n="http://www.xmlbus.com/broadcastService">
      <n:list>PDA, Cell, Email, VoiceMail, IM</n:list>
    </n:broadcastService>
  </env:Header>
  <env:Body>
    <m:Function xmlns:m="http://www.xmlbus.com/broadcastService/send">
      <m:message>
        Eric, you are late for the concall again!
      </m:message>
    </m:Function>
  </env:Body>
</env:Envelope>

```

UDDI는 웹서비스의 등록과 검색을 위한 표준으로 uddi.org를 중심으로 나수의 uddi 오퍼레이터들로 구성되어 있다. 서비스를 등록하기 위해서는 우선 Microsoft, IBM, SAP, Hewlett-Packard 등과 같은 uddi 오퍼레이터들 중 하나를 선택하여 필요한 등록 절차를 마친다. 오퍼레이터에서 제공되는 웹서비스 또는 GUI를 통하여 자신이 제공하는 웹서비스의 이름, 업종, WSDL 파일 등을 등록한다. 이렇게 한곳의 오퍼레이터를 통하여 등록된 정보는 주기적으로 uddi.org에 저장되고 나아가 전체 오퍼레이터들이 공유하게 된다. 등록된 정보의 수정은 처음 등록한 오퍼레이터에서만 가능하다. 등록된 정보의 보안이나 신뢰성을 위하여 제3자 인증기관의 인증을 요청할 수 있다. UDDI의 데이터 모델은 다음과 같다.

- businessEntity: 등록될 비즈니스 또는 다른 개체

- 를 기술하는 상위 구조 다른 구조들을 참조한다.
- businessService: 게시될 서비스의 이름과 설명.
- bindingTemplate: 서비스에 대한 정보. 서비스를 접속하기 위한 시작 주소가 포함된다.
- tModel: 서비스 규격을 유일하게 식별할 수 있는 정보의 모음. 지문과 같은 역할.
- publisherAssertion: 두 개 또는 그 이상의 businessEntity 사이에 존재하는 관계. 예를 들어 자회사 또는 부서.

웹서비스는 궁극적으로 ebXML이나 RosettaNet과 같은 다른 전자거래 표준과 같은 목적을 가지고 있다. 다른 전자거래 표준이 비즈니스 프로세스나 특정 응용분야의 필요에 초점을 맞추어 개발되고 있는 반면, 웹서비스는 bottom-up 접근방법을 취하고 있다. 즉 전자거래에 필요한 기반 기술들을 먼저 정의하고 이들을 기반으로 필요한 기능이나 응용분야의 요구사항등을 정의하는 형태이다. 따라서 웹서비스와 이들 다른 전자거래 표준들은 상호 경쟁적인 입장이라기 보다는 상호 보완하는 협력적인 관계라 할 수 있다.

3. ebXML 표준 개발 현황

UN/CEFACT와 OASIS에 의해 공동으로 제정한 ebXML은 기업간 전자 상거래를 위한 비즈니스 정보 교환시 일관된 방식으로 XML을 적용해 시스템간의 상호 운용성을 확보하기 위한 표준 프레임워크이다. ebXML은 모듈 형태의 명세의 집합으로 인터넷을 통해 모든 규모의 기업이 어느 지역에서나 거래를 하는 것을 가능하게 한다. ebXML을 사용함으로써 거래 메시지의 교환, 거래 관계의 수립, 공통 조건에 의한 데이터 통신, 그리고 비즈니스 프로세스를 정의 및 등록하기 위한 표준 방법을 가지게 된다.

웹서비스의 SOAP, WSDL, UDDI가 웹상에서 XML 문서의 교환을 위하여 RPC 메커니즘을 구현하기 위한 인터넷 사용자들의 표준이라면, ebXML은 EDI 사

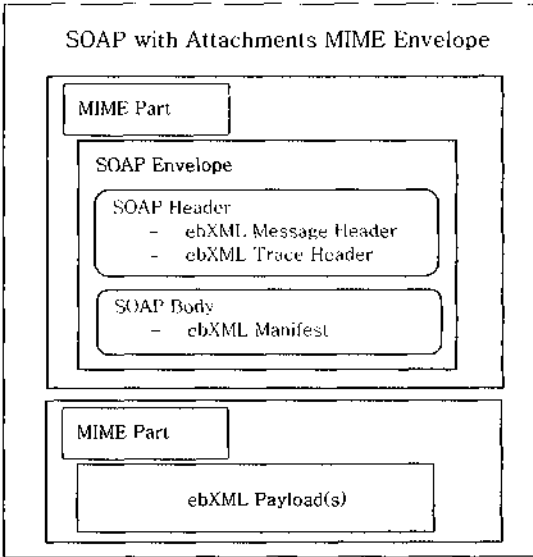


그림 3. SOAP을 사용한 ebXML의 구조

용자들이 고비용의 사설 네트워크보다 저렴한 인터넷 상에서 XML을 이용하여 거래문서를 교환하기 위하여 개발한 것이다. SOAP에 문서 중심의 교환 스타일이 추가되어 ebXML에서도 SOAP을 메시지 전달 규격으로 사용하고 있다. ebXML과 SOAP/WSDL/UDDI 간에는 많은 유사점이 존재한다. 오늘날 ebXML은 W3C에서 SOAP와 다른 규격을 개발하는 데 있어서 요구사항의 역할을 한다. SOAP을 사용한 ebXML의 메시지 구조는 그림 3과 같다.

ebXML의 구조는 다음을 포함한다.

- 비즈니스 프로세스와 그들에 관련된 메시지와 내용
- 비즈니스 프로세스 순서와 관련된 메시지 교환을 게시하기 위한 레지스트리와 검색 메카니즘
- 회사 프로파일
- 교역 파트너 합의서
- 통일된 메시지 전달 레이어

이러한 ebXML 아키텍처에 적합하기 위한 요구사항이 각각의 규격에 정의되어있다. 주요 ebXML의 규격은 다음과 같다.

- 기술적 아키텍처: 완전한 아키텍처에 대한 개요와 자세한 내용을 기술적으로 정의한다.
- 비즈니스 프로세스와 정보 모델링: BPSS의 정의와 BPSS를 생성하기 위한 모델링 방법론.
- 협업 프로토콜 프로파일과 합의서 규격: 교역 파트너 정보의 정의와 XML 문서로 기술하는 방법.
- 레지스트리와 레포지터리 서비스: 비즈니스 프로세스 모델과 정의, 교역 파트너 식별, 그리고 기술적 요구사항들의 저장과 인출.
- 메세징 서비스: ebXML-호환하는 문서들이 네트워크 상에서 전달되는 방법을 정의.
- 핵심 컴포넌트와 핵심 라이브러리: 비즈니스 나 큐먼트 개요와 같이 산업 전반에 걸쳐 공통된 컴포넌트들.

웹서비스 표준인 WSDL, SOAP, UDDI는 메시지 교환에 필요한 기본적인 규격만 정의하고 있다. 현재 W3C에서는 ebXML, 로제타넷과 같은 전자거래 표준을 참고로 다음과 같은 기능을 추가로 개발하고 있다.

- 보안
- 프로세스 플로우
- 트랜잭션 처리
- 메시징

4. RosettaNet 표준 개발 현황

로제타넷은 개방형 전자상거래 표준 프로세스를 제정, 구현, 보급하기 위해 전세계 전자부품, 정보기술, 그리고 반도체제조에 종사하는 500개 이상의 기업들의 컨소시엄이다. 로제타넷은 이 컨소시엄이 정의하는 전자상거래 표준 프레임워크를 지칭하는 명칭이기도 하다. 이같은 표준 프레임워크에는 특히 표준 비즈니스

스 프로세스들이 포함되어 있어 전 세계 공통의 전자 상거래 언어를 제공한다.

로제타넷은 마케팅, 수요예측, 설계, 구매, 제조, 물류, 전자지불의 7개 분야에서 구현사업을 수행해왔다. 구매 분야에서 대표적인 성공사례는 일본의 OMJ로 소니등 200여 업체가 참여했다. 이 사례에서 업무 프로세스만도 12개에 이르고 있다. 이 사업의 핵심 리더인 소니는 그동안 사용해온 EDI를 중단하고 로제타넷을 사용하겠다고 밝히기도 하였다. 로제타넷을 구성하는 표준들은 크게 덕서너리, RNIF, PIP, 그리고 제품과 파트너 코드로 이루어진다.

로제타넷 덕서너리들은 교역 네트워크의 범위 안에서 비즈니스를 운영하는데 필요한 공통의 토대를 제공함으로써 개개의 회사들에 의해 중복되는 노력을 없애며, 각 회사 별로 독자적으로 정의된 용어들로 인해 야기되는 구매 과정상의 혼란을 감소시킨다. 로제타넷의 비즈니스 사전은 당사자들 사이에서 실행되는 비즈니스 거래를 정의하기 위한 요소들을 지정하고, 로제타넷 기술사전은 제품과 용역을 정의하는데 필요한 요소들을 제공해준다.

로제타넷 구현 프레임워크(RNIF: RosettaNet Implementation Framework)의 핵심 사양은 로제타넷 표준의 신속하고 효율적 채택 및 보안을 위해 통상적인 교환 규약을 제공한다. RNIF는 운반, 운송경로 지정 및 포장, 보안, 신호체계, 거래 당사자간의 합의 사항과 같은 정보가 HTML/XML이 사용되는 통신 서버를 통하여 교역당사자들 간에 교환되는 방식을 특별한 사양으로 표시한다. 로제타넷 비즈니스 메시지, 비즈니스 페이로드를 한데 모아 놓은 전송 프로토콜 독립 컨테이너와 연결되어진 헤더 부분, 그리고 로제타넷으로 상호 작용하는 당사자 사이에 교환되어야만 하는 전자서명 같은 여타의 속성들이 포함되어 있다.

로제타넷의 PIP(Partner Interface Process)는 교역 당사자들간의 비즈니스 프로세스를 정의한다. PIP는 교

역 당사자의 비즈니스 프로세스를 정의하는 시스템과 시스템간의 전문화된 대화로서 XML을 기본 문서양식으로 한다. PIP에는 프로세스 어휘를 정의하는 비즈니스 다큐먼트나 파트너 사이의 대화메시지의 구성을 정의하는 비즈니스 프로세스가 들어있다. PIP는 행정 사무, 파트너, 제품 및 서비스 점검, 제품 정보, 계약 관리, 재고 관리, 마케팅 정보, 서비스 및 지원, 생산과 같은 핵심 프로세스에 적용된다.

로제타넷에서는 제품과 파트너 코드를 사용하는데 이는 교역 당사자간의 비즈니스 절차의 조정을 원활하게한다. 다음과 같은 코드가 사용된다.

- Global Company Identifier: PIP에서 글로벌 회사 식별자로 사용하기 위하여 DUNS(Data Universal Numbering System)을 정의하였다.
- Global Product Identifier: PIP에서 글로벌 제품 식별자로 사용하기 위하여 GTIN(Global Trade Item Number)을 정의하였다. GTIN은 14자리 숫자로서 전세계적으로 제품과 서비스를 유일하게 식별한다.
- Global Class Identifier: PIP에서 글로벌 클래스 식별자로 사용하기 위하여 UN/SPSC(United Nations/Standard Product and Services Code)를 정의하였다. 전체 분류 시스템은 5개 레벨로 구성되었고 각 레벨에 2자리 숫자가 할당된다.

로제타넷의 강점은 전자 및 통신업계의 오프라인기업들이 앞장서서 개발하고 사용하고 있다는 점이다. 현장의 필요에 의해 자생적으로 탄생한 표준인 것이다. 그리고 ebXML과 웹서비스의 UDDI 등과 같은 관련 기술들을 채택하여 계속 발전하고 있다. 또한 로제타넷은 물류 및 유통 전문 국제기구인 GSI에 통합됐으며 이 분야에 적용하기 위해 표준 개발작업을 진행하고 있다. 따라서 웹서비스 ebXML, 로제타넷등의 전자거래 표준들은 상호 보완의 입장에서 계속 발전할 것이다.