

EDI-SGML 문서 변환 시스템

이 승익, 조 성배
연세대학교 컴퓨터과학과

Document Translation System for EDI-SGML

Seung-Ik Lee and Sung-Bae Cho
Dept. of Computer Science, Yonsei University

요약

국내외적으로 초고속 정보통신망의 구축이 활발히 진행되고 있으며 이에 따라 개인이나 기업이 거래를 행함에 있어서 정보통신망을 이용하여 정보를 교환하고 업무의 효율성을 높이기 위하여 전자적으로 문서를 교환하는데 대한 필요성이 증대되고 있다. 이를 위한 표준으로 UN에서는 전자문서표준을 제정, 보급함으로써 이 신속하고 정확한 업무의 처리를 돕고 있다. 이 논문에서는 전자문서가 UN에서 제정한 전자문서 표준에 따르고 있는지를 검사하는 EDI 파서와 더불어 이 문서를 SGML 규정에 따르는 문서로 변환하는 시스템을 설계하고 구현한다. 문서를 구조적으로 표현하는 SGML문서로의 변환을 통하여 SGML 문서를 사용하는 많은 응용프로그램들의 잇점을 이용할 수 있을 것이다.

1. 서 론

국내외를 막론하고 고속정보통신망의 구축 작업이 활발히 진행되고 있으며, 이에 따라 정보통신망을 이용하여 정보를 교환하고 업무의 효율화를 꾀하려는 노력이 점차 활성화되고 있다. 이러한 노력이 결실을 맺기 위해서는 우선적으로 전자문서의 표준화 작업이 필요하며 각 표준기구들은 전자문서에 대한 표준안을 꾸준히 제시하고 있는데 대표적으로는 EDI(Electronic Data Interchange) [1, 2, 6, 7, 8]와 SGML(Standard Generalized Markup Language) [3, 4, 5]이 있다.

EDI는 조직 사이에 상업과 의사소통을 원활히 하고자 하는 의도에서 70년대 말에 제안되었다. 그 결과 서로 다른 기업간에 약속된 포맷을 사용하여 사업적 또는 행정상의 거래를 컴퓨터와 컴퓨터간에 행함으로써 업무를 신속, 정확, 효율적으로 처리할 수 있게 되었다.

주로 출판계에서 이용되어온 SGML은 국제표준 조직에서 1986년에 채택되었으며(ISO 8879)[4] 미 국방성의 CALS (Continuous Acquisition and Lifecycle Support) 표준이기도 하다. SGML은 간략히 말하면 마크업 언어의 메타 언어라고 할 수 있다. 마크업 언어는 태그 시스템으로 문서를 전자적으로 분해하고 이동한 다음 새로운 위치에서 다시 결합할 수 있도록 문서의 구조를 유지해

준다. 따라서 SGML은 서로 다른 응용 소프트웨어 사이의 문서교환을 가능하게 할뿐만 아니라 문서의 구조적 정보를 지니고 있기 때문에 이를 이용한 검색 시스템 등의 개발이 활발히 진행되고 있다.

이 논문에서는 EDI 문서가 UN/EDIFACT[8] 문법을 잘 준수하고 있는지를 검사하는 EDI 파서와 EDI 문서를 SGML을 지원하는 여러 응용 소프트웨어에서 사용될 수 있도록 하기 위하여 SGML문서로 변환하는 시스템의 구현에 대하여 알아본다.

2. EDI 문법

통상적인 사무용 통신에 있어서 대부분의 메시지는 일반적으로 승인된 규격의 양식과 구문을 따르게 된다. 예를 들어 사무용 편지는 발신자의 주소, 성명이 적힌 편지에 수신자의 주소, 성명을 쓰고 인사말을 쓰는 식으로 작성된다. 이러한 양식과 구문에 대한 규정은 EDI 세계에서도 존재하며 이것을 EDI 표준이라고 부른다. EDI 표준은 여러 가지가 있지만 UN에서 제정한 UN/EDIFACT(EDI for Administration, Commerce, and Transport)은 EDI의 표준적인 문법으로서 자리잡고 있으며 국제적으로도 인정받고 있다.

개략적으로 UN/EDIFACT 표준내의 교환정보(interchange) 구조를 살펴보기로 하자. 교환정보는 전자적으로

문서를 주고받을 때 한번에 주고받는 단위가 된다. 그림 1과 같이 하나의 교환정보 안에는 여러 개의 메시지가 존재하며 하나의 메시지는 여러 개의 세그먼트로 구성된다. 하나의 세그먼트는 다시 여러 개의 데이터 엘리먼트로 구성된다.

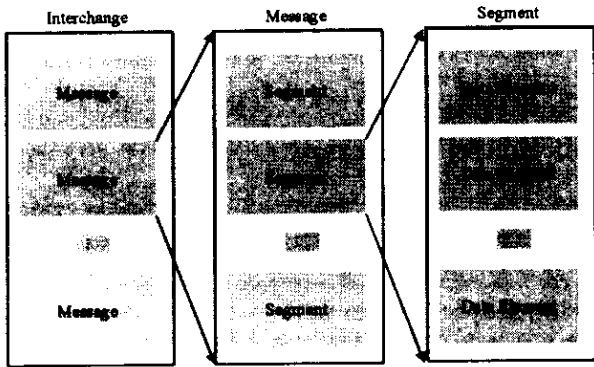


그림 1 Interchange 구조

3. EDI-SGML 변환 시스템

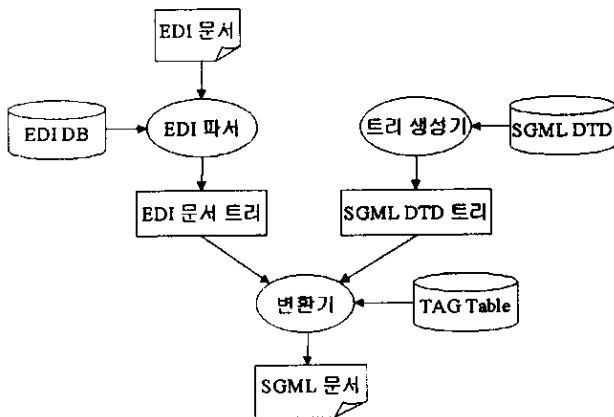


그림 2 시스템 구성도

그림 2는 EDI-SGML 변환 시스템의 전체 구조를 보여준다. EDIFACT 문법 데이터베이스와 SGML DTD 데이터베이스를 참조하여 EDIFACT 문서를 SGML문서로 변환시킨다. 시스템에 EDI 문서가 입력되면 EDI 문법 데이터베이스를 참조하여 파서가 EDI 문법 트리를 생성하고 트리 생성기는 SGML DTD 데이터베이스를 참조하여 DTD 트리를 생성한다. 최종적으로 생성된 두 트리를 이용하여 변환기가 EDI 문서를 SGML로 변환한다.

3.1 EDI 파서

EDI 파서는 EDI 문서를 입력으로 하고 데이터베이스에서 EDIFACT 규정을 읽어 들여 EDI 문서가 EDIFACT 규정을 준수하고 있는지를 검사한다.

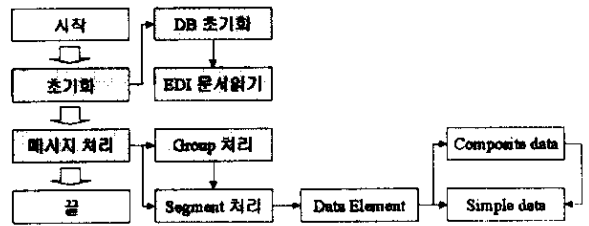


그림 3 파싱 과정

파서가 동작하는 전체적인 과정은 그림 3과 같다. 우선 데이터베이스 파일에 접근하여 EDIFACT 규정을 읽어 들인 후 이를 이용하여 문법 트리를 구성한다. EDI 문법은 하나의 메시지가 여러 개의 세그먼트 또는 세그먼트 그룹으로 이루어져 있으며 하나의 세그먼트는 여러 개의 단순 데이터 엘리먼트나 복합 데이터 엘리먼트로 이루어져 있는데 구성된 문법 트리가 이를 표현하고 있다. 문법 트리의 노드는 여러 가지 종류가 있는데 노드의 종류에 따라서 표 1과 같은 정보를 지니고 있다.

표 1 노드의 정보

노드의 종류	주요 정보
메시지 노드	- 이름 - 세그먼트 리스트
세그먼트 노드	- 이름 - 필수/조건 - 최대 반복 회수 - 데이터 엘리먼트 리스트
복합 데이터 노드	- 이름 - 필수/조건 - 단순 데이터 엘리먼트 리스트
단순 데이터 노드	- 이름 - 필수/조건 - 데이터 타입 - 데이터 최소 길이 - 데이터 최대 길이 - 코딩 값

메시지 노드는 메시지의 이름과 자신을 구성하는 세그먼트에 대한 리스트를 지니고 있으며, 세그먼트 노드는 세그먼트의 이름, 해당 세그먼트가 현재 메시지 안에서 필수적인지 조건적인지를 표시하는 필수/조건, 메시지 안에서 해당 세그먼트가 얼마나 반복적으로 나타날 수 있는지를 표시하는 최대 반복 회수, 자신을 구성하는 데이터 엘리먼트에 대한 리스트를 지니고 있다. 복합 데이터 노드는 자신의 이름과 자신을 구성하는 단순 데이터 엘리먼트들의 리스트를 지니고 있다. 단순 데이터 노드는 이름, 필수/조건, 그리

고 데이터의 타입(숫자, 알파벳, 숫자와 알파벳의 혼합), 그리고 데이터의 최소 길이와 최대 길이에 관한 정보를 지니고 있다. 이와 같은 정보를 지니고 있는 노드들로 구성된 문법 트리가 구성되면(그림 4) 트리의 각 노드를 방문하면서 파싱이 이루어진다.

메시지 노드에서는 주어진 EDI 문서의 종류가 메시지 노드에 기록된 이름과 일치하는지를 검사한다. 이름이 일치하면 이 메시지를 구성하는 여러 세그먼트에 대한 파싱에 들어간다(그림 4에서 루트 노드의 자식 노드들).

세그먼트 노드에서는 EDI 문서에서 입력된 세그먼트의 이름이 현재 세그먼트 노드의 이름과 일치여부를 조사하고 이름이 일치하지 않는 경우 해당 세그먼트가 최대 반복 회수를 초과하여 반복적으로 EDI 문서에 나타나고 있는지를 조사한 후 이 세그먼트를 구성하는 데이터 엘리먼트들에 대한 파싱 루틴을 호출한다(그림 5).

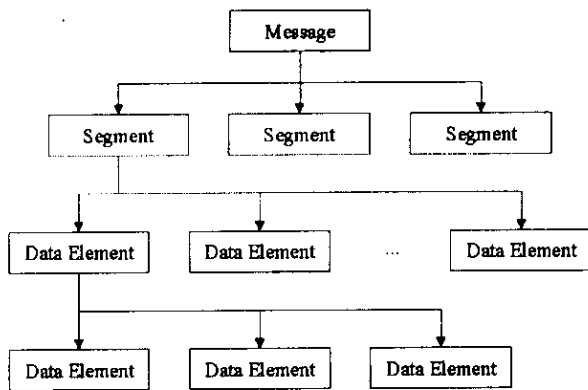


그림 4 문법 트리

복합 데이터 노드에서의 파싱은 세그먼트노드에서의 파싱과 마찬가지로 자신을 구성하는 단순 데이터 노드들에 대한 파싱 루틴의 호출로 실행된다.

단순 데이터 노드에서는 현재의 데이터 엘리먼트가 생략 가능한지를 검사한다. EDI 문서에서 현재의 데이터 엘리먼트가 생략된 경우 단순 데이터 노드의 '필수/조건' 필드의 값이 '조건' 일 경우 파싱은 성공한다. 생략되어 있지 않을 경우는, 입력된 데이터 엘리먼트의 데이터 타입, 최소 길이, 최대 길이 등을 검사하고 코딩된 값일 경우 그 값이 등록된 값인지 조사한다. 등록되어 있지 않은 값이 입력된 경우 파싱은 실패한다.

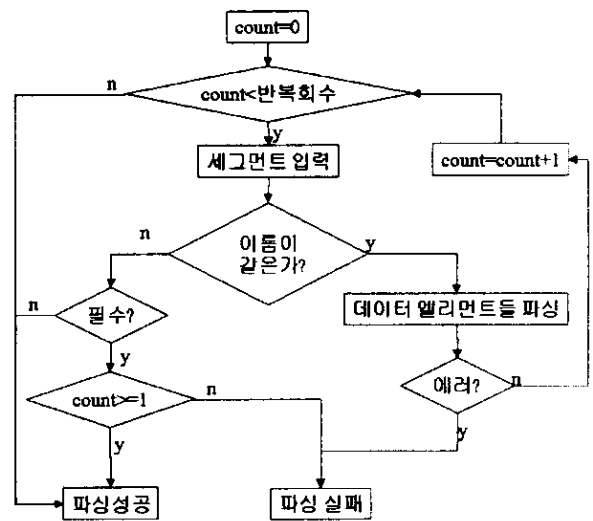


그림 5 세그먼트 파싱 과정

3.2 EDI 문서의 SGML 문서로의 변환

EDI 문서를 SGML 문서로 변환하기 위해서 EDI 트리 와 SGML의 DTD 트리를 구성한다. EDI 트리는 그림 4와 동일하며 단 EDI 문서에서 생략된 세그먼트 및 데이터 엘리먼트는 트리노드로서 표현되지 않으며 노드의 내용은 실제 EDI 문서의 내용이다.

DTD 트리를 구성하기 위해서는 EDI 문서의 종류에 맞는 SGML DTD를 먼저 정의한 후 이를 이용하여 트리를 구성한다. SGML DTD는 EDI의 각 메시지마다 미리 정의한다. DTD 트리를 구성하는 각 노드는 main element, sub element, attribute 노드의 세 가지 타입이 있다(그림 6, 표 6).

id	iamstart	attribute ptr	content ptr
----	----------	---------------	-------------

(a)

id	own ptr	next ptr
----	---------	----------

(b)

id	next ptr
----	----------

(c)

그림 6 DTD 트리 노드형

EDI 문서의 SGML 문서로의 변환을 위하여 EDI 트리의 루트에서 마지막 리프까지 preorder search 하면 SGML 형식의 문서를 얻게 된다. 변환 과정에서 트리에 있는 데이터 엘리먼트의 이름을 사용자의 편의대로 바꾸어 SGML 문서의 태그 이름으로 표현하기 위해 테이블을 참

조한다. 표 3은 EDI 데이터 엘리먼트의 이름을 SGML 태그 이름으로 변환하기 위한 테이블의 일부를 보여준 것이다.

표 2 노드 필드 별 의미

태그	의미
id	element의 이름
iamstart	1 이면 루트
attribute ptr	attribute를 가리키는 포인터
content ptr	element의 내용을 가리키는 포인터
id	sub element의 이름
own ptr	자신이 루트인 트리를 가리키는 포인터
next ptr	다음 sub element를 가리키는 포인터
id	attribute의 이름
next ptr	다음 attribute를 가리키는 포인터

표 3 태그 테이블

0062	MREF
S009	MID
0065	MTYPE
.	.
.	.

4. 실행 예

EDI 문서 전체를 대상으로 한 결과는 다 보여주기 어려우므로 여기서는 간단한 하나의 세그먼트가 어떻게 변환되는지를 보이도록 하겠다.

UNH+ME000001+INVOIC:90:1:UN:EAN004'

그림 7은 앞의 세그먼트에 대한 EDI 트리의 구성결과를 보여준다. 이 트리와 함께 DTD 트리 및 표 3의 태그를 이용하면 그림 8과 같은 결과를 얻을 수 있다.

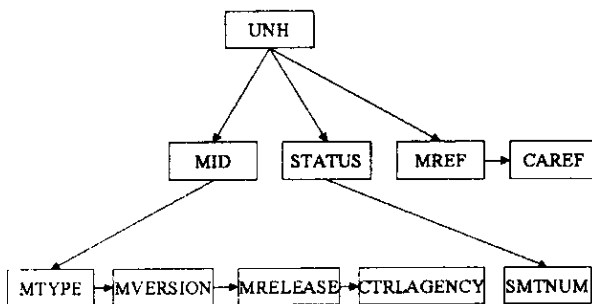


그림 7 구성된 EDI 트리

```

<INVOIC>
<UNH MREF="ME000001">
  <MID MTYPE="INVOIC" MVERSION="90"
    MRELEASE="1" CTRLAGENCY="UN"
    ASCODE="EAN004">
  </MID>
</UNH>
<BGM>
    
```

그림 8 SGML 문서

5. 결론

지금까지 EDI문서의 파싱과 이 문서의 SGML 문서로의 변환에 대하여 살펴보았다. 많은 기업이 EDI 시스템의 도입을 검토하고 있고 실제로 수출입 업무 등에서는 EDI가 실용되고 있는 상황에서 국제적인 EDI 규약으로 제시되는 UN/EDIFACT를 따르는 EDI 파서를 가지는 것은 바람직한 일이라 할 수 있다. 또한 EDI문서를 SGML 문서로 변환함으로써 현재 SGML 기반 응용 프로그램을 충분히 활용할 수 있을 뿐만 아니라 앞으로 많은 응용프로그램들이 SGML을 지원하도록 개선될 것에 견주어 본다면 실질적으로 많은 잇점이 있다고 할 수 있다.

참고 문헌

- [1] 오호근, *EDI란 무엇인가*, 크라운 출판사, 1993.
- [2] 임승택, *EDI 혁명으로 가는 길*, 대청정보시스템, 1997.
- [3] Bryan, M., *SGML : An Author's Guide to the Standard Generalized Markup Language*, Addison-Wesley, 1991.
- [4] Goldfarb, C.F. and Rubinsky, Y., *The SGML Handbook*, Clarendon Press, 1990.
- [5] Herwijnen, E., *Practical SGML*, 2nd Ed., Kluwer Academic Pub., 1994.
- [6] <http://www.premenos.com>
- [7] <http://www.ezedi.com>
- [8] <http://www.unece.org/trade/untdid/>