

관계형 데이터 가시화를 위한 XML&XSLT 코드 생성 시스템

* 강원석^o * 최기선

* 한국과학기술원, 첨단정보기술연구센터(AITrc)

* 한국과학기술원, 전문용어언어공학연구센터(KORTERM)

{* wskang^o, * kschoi}@world.kaist.ac.kr

A XML&XSLT code generation system for visualization of a RDBMS data

* Wonseok Kang^o * Keysun Choi

* AITrc, KAIST

* KORTERM, KAIST

요 약

최근 네트워크의 활성화로 인하여 많은 지식 정보들이 생겨나고 있다. 지식 정보들은 구조적인 정보로 표현되고 응용 프로그램들 상에서 대량으로 처리하기 위해 관계형 데이터베이스 시스템을 이용하여 많이 구축된다. 그러나 대부분의 응용 프로그램들 상에서 관계형으로 구성된 지식 정보들을 적당히 표현할 방법이 부족하다. 또한 이들을 가공 처리하여 표현하는 응용 프로그램들을 유지하는 데 많은 비용이 요구된다. 본 논문에서는 웹상이나 응용 프로그램 등에서 XML과 XSLT를 이용하여 관계형화 된 지식 정보들에 대해 효율적으로 가시화시킬 수 있도록 지원하는 시스템을 제시한다. VAS의 효율성을 보이기 위해 다양한 OS 플랫폼 상에서의 응용 프로그램들을 구축하였다.

1. 서론

최근 네트워크의 활성화로 인하여 많은 지식 정보들이 생겨나고 있다. 지식 정보들은 구조적인 정보로 표현되고 응용 프로그램들 상에서 대량으로 처리하기 위해 관계형 데이터베이스 시스템을 이용하여 많이 구축된다. 지식 정보들은 XML과 같은 구조화된 마크 업 언어로 구성되는 추세에 있다. XML을 기초로 한 데이터들은 XSLT를 이용하여 언어학자 및 공학자들 자신이 원하는 형태로 재구성할 수 있는 유용한 변환 스크립트 언어이다. 그러나 대부분의 응용 프로그램들 상에서 관계형으로 구성된 지식 정보들을 언어학자 및 공학자들이 XML&XSLT 같은 마크 업 언어를 이용하여 처리할 수 있는 유용한 방법이 부족하다. 언어학자나 공학자들에 대해 지식 정보를 처리하는 데 도움을 주는 여러 가지 연구 및 시스템들이 개발되어져 있다. 그러나 기존 연구들은 XML 정보를 표현하기 위해서 단순히 처리한 데이터를 저장 및 입력 받는 기능으로 구성되어져 있어 XSLT를 이용하여 처리하는 것은 상당한 어려움이 있다[1]. 또한 지식정보들이 XML 데이터로 구성되고 여기에 맞는 XSLT를 구성하여 언어학자 및 공학자들이 처리하게 쉽도록 구성된 시스템들이 있다. 그러나 이러한 기능을 제공하는 기존 연구들은 여러 가지 XML 데이터들에 대해 한번은 XSLT를 재구성하여야 한다는 문제점이 발생한다[2]. XSLT를 구성하는 것은 언어학자뿐만 아니라 공학자들에게도 많은 코딩 비용이 요구된다. 또한 평면 구조를 가지는 RDBMS 데이터들에 대해 다

른 형태(예로 계층적 구조 표현 등)로 표현하는 것은 XSLT를 구성하는데 있어서 더욱더 복잡한 구조로 코딩을 해야 되어 더 많은 XSLT 코딩 비용이 요구된다.

본 논문에서는 앞에서 기술한 내용의 문제점을 해결하기 위해 XML&XSLT를 이용하여 관계형화 된 지식 정보들에 대해 효율적으로 가시화시킬 수 있도록 지원하는 시스템(VAS : Visualization Assistant System For a RDBMS)을 제시한다. VAS의 효율성을 보이기 위해 다양한 OS 플랫폼 상에서의 응용 프로그램들을 구축하였다. 본 논문에서는 2장에서 XML&XSLT에 관해 간략히 설명하고 3장에서는 VAS에 대해 설명한다. 마지막 4장에서는 결론 및 향후 계획에 대해 설명한다.

2. XML&XSLT

XML은 Extensible Markup Language로 한정된 형식으로만 구성된 HTML과는 달리 확장성을 가지는 마크 업 언어이다. 지식 정보를 구성하는 사용자들은 XML을 이용하여 직접적으로 태그를 구성할 수 있다. 이러한 이점을 통하여 사용자들은 자신들이 표현하고자 하는 지식 정보들을 정확하고 구조화된 형태로 표현할 수 있으며 정의된 데이터들에 대해서 의미적 요소를 태그들에게 부여함으로써 한층 더 효과적인 지식 정보를 구축할 수 있다[3].

XSLT는 XPath에 정의된 표현들을 사용하여 XML의 구조화된 정보를 다른 형태의 포맷으로 변환하는데 이용되는

XML의 한 형태이다. XSLT는 XML로 구성된 무수한 데이터들을 언어학자나 공학자들 자신이 원하는 포맷으로 구성하기 위해 아주 유용한 언어이다[3].

3. VAS : 관계형 데이터 가시화 도구

본 시스템에서는 관계형 데이터들을 XML로 구조 변환을 한 후 이 데이터를 언어학자나 공학자들 자신이 원하는 다른 포맷 형태의 데이터 구성을 도와주기 위해 개발된 시스템이다. 본 시스템은 언어학자나 공학자들에게 XSLT의 중간 템플릿 코드를 자동으로 구성하여 줌으로써 복잡한 구조로 변환(의미별 그룹화 및 가지치기)하는데 아주 많은 도움을 줄 수가 있다. 3.1절에서는 본 시스템의 구성도를 설명하고 3.2절에서는 본 시스템을 이용하여 운영체제 독립적으로 다양한 응용 시스템들에 적용한 가시화 구축 방법 및 예들에 대해 기술한다.

3.1 VAS 구성도

VAS는 그림 1과 같이 SQL Generator, XML Code Generator 및 XSLT Code Generator로 구성되어진다. SQL Generator는 그림 4의 2번창으로부터 3번창으로 사용자가 원하는 관계형 데이터의 정보만을 선택하여 옮긴 후 이들을 SQL문으로 생성하여 RDBMS로부터 데이터를 추출한다.

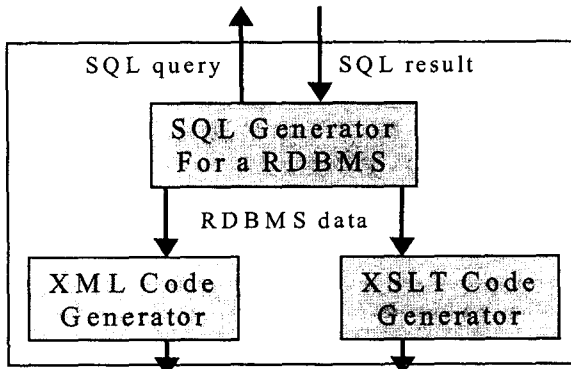


그림 1 VAS 구성도

```

    <DBName>
        <Table2>
            <col1>data</col1>
            <col2>data</col2>
            .....
            <colN>data</colN>
        </Table2>
        <Table2>
            .....
        </Table2>
    </DBName>
    
```

그림 2 Exported XML-Structure I

```

    <DBName>
        <Table2 col1=ID1 col2=ID1ID1>
            <col1>data</col1>
            <col2>data</col2>
            .....
            <colN>data</colN>
        </Table2>
        <Table2 col1=ID1 col2=ID1ID2>
            .....
        </Table2>
    </DBName>
    
```

그림 3 Exported XML-Structure II

XML Code Generator는 SQL Generator로부터 추출된 RDBMS 데이터들을 사용자가 원하는 타입으로 XML 코드를 생성한다. XML Code Generator가 RDBMS를 위해 기본적으로 생성하는 구조는 그림 2와 같다.

여기서 사용자가 의미별 그룹화 및 가지치기를 원할 때에는 그림 4의 3번창에서 의미별 그룹 속성을 설정 하는 기능을 이용한다. 이 작업을 완료하면 그림 3과 같이 각 XML 노드별로 ID가 자동생성 부여된다. 이 ID는 XSLT에서 의미별 그룹화 및 가지치기에 사용되어진다.

XSLT Code Generator는 SQL Generator로부터 추출된 데이터들과 그림4의 3번창에서 설정한 의미별 그룹 설정 상태에 따라 코드가 생성되어 그림4의 5번창에 그 결과가 출력된다. VAS는 크게 두 가지 XSLT 템플릿 코드를 생성한다. 첫 번째로 RDBMS의 고유한 데이터 형태인 평면구조의 RDBMS 데이터를 변환 할 수 있는 템플릿 코드이다. 두 번째로 RDBMS의 관계형 데이터를 의미별 그룹화 및 가지치기를 할 수 있는 템플릿 코드이다. 이는 RDBMS의 평면 구조에 대해 의미별 그룹 및 해제를 제공하기 때문에 언어학자나 공학자는 이들을 이용하여 유용한 구조로 변환을 할 수 있다. 그림 5는 RDBMS 평면 구조를 의미별 그룹화 및 가지치기된 구조를 표현할 수 있도록 VAS에서 생성된 XSLT의 Pseudo 코드이다.

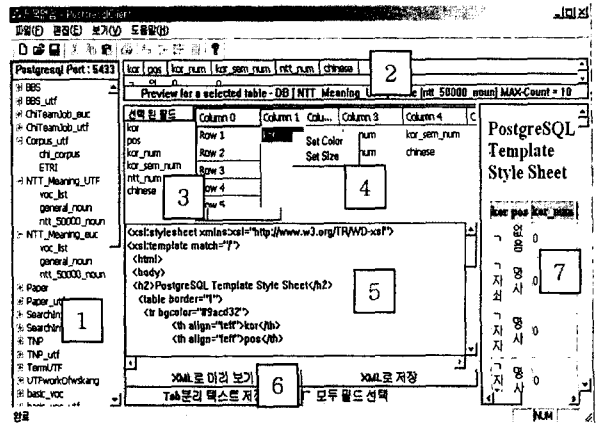


그림 4 VAS의 GUI

```
<xsl:for-each select="//TABLENAME[generate-id() = generate-id(key('Key1ID', @Key1))]">
  <xsl:variable name="lngKey1ID"><xsl:value-of select="@Key1" /></xsl:variable>
  <xsl:variable name="lngStrKey1ID"><xsl:value-of select="Key1" /></xsl:variable>
  <xsl:variable name="lstKey1ID" select="//TABLENAME[@Key1 = SingKey1ID]" />
  <xsl:choose>
    <xsl:when test="$lstKey1ID[generate-id() = generate-id(key('Key2ID', @Key2))]">
      <xsl:for-each select="$lstKey1ID[generate-id() = generate-id(key('Key2ID', @Key2))]">
        <xsl:variable name="lngKey2ID"><xsl:value-of select="@Key2" /></xsl:variable>
        <xsl:variable name="lngStrKey2ID"><xsl:value-of select="Key2" /></xsl:variable>
        <xsl:variable name="lstKey2ID" select="$lstKey1ID[@Key2 = SingKey2ID]" />
        //의 매 Level Depth N을 위한 Key N까지의 Recursive Code//
      </xsl:for-each>
    </xsl:when>
    <xsl:otherwise>
      </xsl:otherwise>
    </xsl:choose>
  </xsl:for-each>
```

그림 5 RDBMS 데이터를 위한 의미별 그룹화 생성 XSLT 코드

3.2 VAS를 이용한 RDBMS 가시화

XML&XSLT가 컴퓨터 플랫폼에 독립적이기 때문에 XML&XSLT 파서만 존재한다면 다양한 플랫폼에 적용이 가능하다. 본 절에서는 이러한 특성을 이용하여 UNIX/Windows 플랫폼에서의 VAS를 이용한 응용 시스템들에 대해서 설명한다.

그림 6은 VAS를 통하여 응용 시스템에 이용할 수 있도록 구축되어진 하나의 예이다. VAS를 이용하여 그림 6과 같은 시스템을 구성함으로써 공학자들은 생성된 XSLT 코드를 편집하여 다양한 시스템들에게 응용 프로그램들을 구축할 수 있다. XSLT Pool에 최종 편집된 XSLT 코드들을 관리함으로써 향후에 재사용성도 증가하며 동적인 응용 소프트웨어 개발을 한층 더 빠른 시간에 구축할 수 있다.

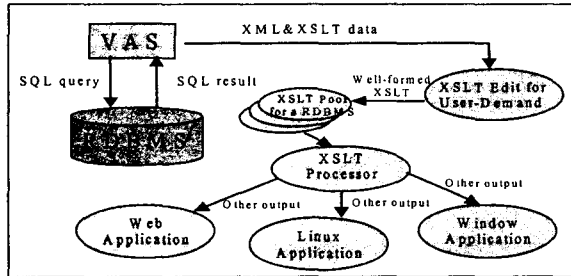


그림 6 VAS를 이용한 RDBMS 가시화 구축 예
그림 7,8,9는 웹 인터페이스 및 윈도우 응용 프로그램에서 가시화한 것으로 그림 7은 중한개념사전 명사에 대해서 관계형 데이터를 재구성하여 나타낸 사전 모습이다. 그림 8은 그림 7의 데이터를 다른 형태로 재구성한 모습이다.

第三人称(单数)삼인칭 단수	(11111131, 22)
人<状态>사람	人(사람)
	(11111231, 67)
佛人위인	人(佛)
	(111116111, 183)
贤시현자	人(사람 인자)
	(111116131, 189)

그림 7 중한개념사전 검색1

人	佛	(佛 人<状态>삼인칭 단수)	(佛 人(사람))
1) [N1] 佛(佛)<O>	2) 佛(佛)	(佛 佛(佛))	(佛 佛(佛))
3) [N1] 佛(佛)<O>	4) 佛(佛)	(佛 佛(佛))	(佛 佛(佛))
5) [N1] 佛(佛)<O>	6) 佛(佛)	(佛 佛(佛))	(佛 佛(佛))
7) [N1] 佛(佛)<O>	8) 佛(佛)	(佛 佛(佛))	(佛 佛(佛))
9) [N1] 佛(佛)<O>	10) 佛(佛)	(佛 佛(佛))	(佛 佛(佛))
11) [N1] 佛(佛)<O>	12) 佛(佛)	(佛 佛(佛))	(佛 佛(佛))
13) [N1] 佛(佛)<O>	14) 佛(佛)	(佛 佛(佛))	(佛 佛(佛))
15) [N1] 佛(佛)<O>	16) 佛(佛)	(佛 佛(佛))	(佛 佛(佛))
17) [N1] 佛(佛)<O>	18) 佛(佛)	(佛 佛(佛))	(佛 佛(佛))
19) [N1] 佛(佛)<O>	20) 佛(佛)	(佛 佛(佛))	(佛 佛(佛))

그림 8 중한개념사전 검색2

1) [N1] 佛(佛)<O>	[V] 佛 [Aux] 3 [N3] 佛(佛)<O>
2) [N1] 佛(佛)<O>	[V] 佛 [Aux] 7 [N3] 佛(佛)<O>
3) [N1] 佛(佛)<O>	[V] 佛 [Aux] 7 [N3] 佛(佛)<O>
4) [N1] 佛(佛)<O>	[V] 佛 [Aux] 7 [N3] 佛(佛)<O>
5) [N1] 佛(佛)<O>	[V] 佛 [Aux] 7 [N3] 佛(佛)<O>

그림 9 중한개념사전 검색3-격틀

그림 9는 중한개념 사전 동사에 대한 격틀 구조 관계형 데이터를 재구성하여 나타낸 사전 모습이다. 그림 7,8,9는 중국어 워드를 검색으로 하여 나타내어진 화면이다.

4. 결론 및 향후 계획

다양한 지식 정보의 표현이 나날이 변경되는 상황에서 지식 정보의 처리 시간 또한 빠르게 반응하여야 한다. 지식 정보의 표현이 최근 들어 XML 기반의 마크 업으로 다양하게 표현되고 있다. 이러한 XML 데이터들을 효율적으로 재구성할 수 있게 한 것이 XSLT이다.

VAS는 XML 데이터들을 효율적으로 재구성할 수 있는 XSLT를 언어학자 및 공학자들에게 지원해 줌으로써 사용자 각자들이 빠른 시간에 다양한 방법으로 지식 정보들을 처리할 수 있다. VAS는 XSLT의 모든 기능을 제공하지는 않는다. 앞으로 관계형 데이터를 위한 가시화 지원 XSLT 코드를 좀더 세부적으로 추가하여야 할 것이다.

-참고문헌-

[1] Hamish Cunningham et al., Developing Language Processing Components with GATE(a User Guide), The University of Sheffield, 2001-2002
 [2] Ulrich Sachafer, WHAT: An XSLT-based Infrastructure for the Intergration of Natural Language Processing Components, HLT-NAA-CL 2003 Workshop, pp. 9-16, 2003
 [3] 류광, Professional XML APPLICATIONS, 정보문화사, 1999