

WeXGene: 웹 기반 XML 데이터 생성기

신 선 미* · 정 회 진** · 이 상 호***

요 약

XML 데이터베이스 시스템 성능 평가를 수행하기 위해서는 다양한 형태의 XML 데이터 생성을 필요로 한다. 기존 XML 데이터 생성기들은 특정 성능 평가에 적용 가능한 데이터만을 생성하고, 그 기능 또한 매우 제한적이다. 본 논문에서는 기존 XML 데이터 생성기들의 제한된 기능을 보완하고 동시에 새로운 기능들을 추가한 새로운 XML 데이터 생성기인 "WeXGene"을 제안한다. WeXGene은 열 단위 또는 행 단위의 데이터를 포함하는 사용자 데이터 파일 및 "SDTD(Symbolic DTD)" 또는 입력 파라미터를 명시한 구조 정의 화일을 이용하여 XML 데이터를 생성한다. 또한 사용자 데이터 화일 없이도 임의의 데이터를 포함하는 XML 데이터 생성이 가능하다. 본 논문은 WeXGene의 구체적인 설계 내용, 구조, 데이터 생성과정 등을 기술한다. 또한 타 XML 데이터 생성기와 기능을 비교 분석하였다.

WeXGene: Web-based XML Data Generator

Sun Mi Shin* · Hoe Jin Jeong** · Sang Ho Lee***

ABSTRACT

We need to generate various kinds of XML data to evaluate XML database systems. Existing XML data generators are developed to generate XML data that are suitable for particular evaluation methods, and their functionalities are limited in terms of generating XML data. This paper introduces a new XML data generator, WeXGene, that not only improves the drawbacks of existing data generators but also adds new data generation functionalities. For generating XML data, WeXGene uses the user data files and the structure definition files, which specify SDTD(Symbolic DTD) or input parameters. The user data file is a text data file that has column data or row data. It is also possible that WeXGene generates XML data without accessing the user data file. This paper presents the design details, overall system architecture, and data generation process of WeXGene. An analytic comparison with other XML data generators is also presented.

키워드 : XML 데이터 생성기(XML Data Generator), 데이터베이스 벤치마크(Database Benchmarks), 성능 평가(Performance Evaluation), XML

1. 서 론

XML(eXtensible Markup Language)이 웹에서의 데이터 표현과 교환의 표준으로 부각되면서 데이터베이스 시스템 개발 업체들은 XML 데이터 처리를 위해 기존 데이터베이스 시스템에 관련 기능을 추가하거나 전용 XML 데이터베이스 시스템(native XML Database systems)을 새로 개발하였다. 전용 XML 데이터베이스 시스템은 데이터의 논리적인 저장 단위로 XML 문서를 사용하고, XML 문서 구조 그대로 저장 및 관리되어 XML의 장점을 최대한 활용할 수 있는 데이터베이스 시스템이다. 따라서 새로운 XML 관련 기능이나 전용 XML 데이터베이스 시스템에 대한 성능 검증이 요구되었고, 이를 위해 Xbench 성능 평가[1], XMark

성능 평가[2], Michigan 성능 평가[3] 등과 같은 XML 데이터베이스 시스템 성능 평가들이 개발되었다.

데이터베이스 시스템 성능 평가는 다양한 실험 데이터를 요구한다. 실세계 데이터(real world data)를 사용하여 성능 평가를 수행하는 것이 가장 효과적이다. 하지만 이를 얻기 위해서는 많은 시간과 비용이 소모되기 때문에 일반적으로 실세계 데이터의 종류 및 특성과 유사한 합성 데이터(synthetic data)를 사용한다. 데이터 생성기는 합성 데이터를 생성하고[4], 본 논문에서 제안하는 새로운 XML 데이터 생성기인 "WeXGene(Web-based XML Data Generator)"은 이와 같은 합성 데이터를 포함하는 데이터 화일을 이용하여 XML 데이터 생성이 가능하다. 현재까지 개발된 XML 데이터 생성기는 ToXgene[5], Niagdatagen[6], Xmlgen[2], Mbgen[3] 등이 있으며, 이들은 특정 성능 평가에 한정되어 적용 가능한 XML 데이터를 생성하고 그 기능 또한 제한적이다.

ToXgene은 열(column) 단위 사용자 데이터 화일만을 사용하여 XML 데이터를 생성하기 때문에 행(row) 단위로 저장되는 대부분의 사용자 데이터 화일을 사용할 수 없다. 파

* 본 논문은 2004년도 한국학술진흥재단의 지원에 의하여 연구되었습니다. (KRF-2004-005-D00172)

† 정 회 원 : (주)CNP차앤박화장품 IT팀

** 준 회 원 : 숭실대학교 대학원 컴퓨터학과 박사과정

*** 정 회 원 : 숭실대학교 컴퓨터학부 교수

논문접수 : 2004년 10월 21일, 심사완료 : 2005년 3월 14일

라미터 화일을 이용하여 계층 구조 XML 데이터를 생성하는 Niagdatagen은 XML 데이터의 엘리먼트(element) 내용이나 속성(attribute) 값 정의가 불가능하다. Xmlgen은 사용자가 보유하고 있는 데이터를 활용하여 XML 데이터를 생성할 수 없고, 또한 다수 개의 XML 데이터 화일을 생성하지 못하는 단점을 가진다. 특정 데이터 구조의 XML 데이터만을 생성하는 Mbgen은 XML 데이터의 구조를 사용자가 직접 정의할 수 없는 문제점을 가진다. 이처럼 기존 XML 데이터 생성기들은 사용자의 다양한 XML 데이터 생성 요구에 적극적으로 대처할 수 없다는 문제점을 지니므로 다양한 기능을 효과적으로 지원할 수 있는 새로운 구조의 XML 데이터 생성기 개발이 요구된다.

본 논문에서는 WeXGene을 제안하고 그 내용을 설명한다. WeXGene은 XML 데이터베이스 시스템 성능 평가를 위해 개발된 웹 환경에서 사용 가능한 새로운 XML 데이터 생성기이다. WeXGene은 열 단위 및 행 단위로 구성된 사용자 데이터 화일과 데이터 구조 정의 화일을 이용하여 XML 데이터를 생성하거나 사용자 데이터 화일 없이 임의의 XML 데이터를 생성한다. 데이터 생성을 위해 필요한 데이터 구조 정의 화일로는 제한된 형태의 DTD(Document Type Definition) 화일 또는 본 논문에서 새롭게 정의한 "SDTD(Symbolic DTD)" 화일을 사용한다. 제한된 형태의 DTD 화일은 일반적인 DTD 화일에서 엘리먼트 이름 및 하위 엘리먼트 이름 정의, 엘리먼트의 속성 포함 여부 및 속성 이름 정의, 엘리먼트의 내용 포함 정의만이 표현된 화일을 의미한다. 또한, 엘리먼트 이름 등의 간단한 파라미터 값들을 포함하고 있는 입력 파라미터 화일을 데이터 구조 정의 화일로 사용할 수 있다.

WeXGene은 행 단위 사용자 데이터 화일을 열 단위의 데이터 화일로 분할하는 기능과 열 단위 데이터 화일들을 새로운 행 단위의 데이터 화일로 병합하는 기능을 제공한다. 이를 통해 사용자들은 데이터 화일의 효과적인 재사용이 가능하다. 또한, WeXGene은 XML 데이터의 선택 추출 기능 평가에 필요한 유일한 엘리먼트 이름 작성 기능을 제공한다. WeXGene은 사용자의 다양한 XML 데이터 생성 요구를 유연하게 수용할 수 있는 구조를 가지며, 웹 기반의 사용자 인터페이스를 지원하여 사용 편리성을 도모한다.

본 논문의 구성은 다음과 같다. 2장에서는 XML 데이터베이스 시스템 성능 평가를 위해 개발된 기존 XML 데이터 생성기 관련 연구를 기술하고, 3장에서는 WeXGene 에서 데이터 생성을 위해 사용되는 사용자 데이터 화일과 SDTD 화일, 입력 파라미터 화일에 대해 기술한다. 또한, WeXGene 의 모듈 구성과 사용자 인터페이스를 보이고, 타 XML 데이터 생성기와의 비교 분석 내용을 설명한다. 4장에서는 WeXGene을 이용한 XML 데이터 생성 예를 기술하고, 마지막으로 5장에서 결론을 맺는다.

2. 관련 연구

사용자가 XML 데이터를 다양한 XML 데이터베이스 시

스템 성능 평가에 적용하기 위해서는 XML 데이터의 구조 및 엘리먼트 내용이나 속성 값을 사용자가 임의로 정의할 수 있어야 한다. 또한 사용자의 다양한 성능 평가 요구에 유연하게 대처할 수 있는 데이터 생성기 구조를 지녀야 한다. 그러나 지금까지 대부분의 XML 데이터 생성기들은 특정 성능 평가에 이용 가능한 XML 데이터만을 생성하며, 그 기능 또한 제한적이었다. 본 장의 관련 연구에서는 XML 데이터베이스 시스템 성능 평가를 위해 개발된 XML 데이터 생성기 중에서 ToXgene, Niagdatagen, Xmlgen, Mbgen 네 개의 데이터 생성기에 대해 간략히 기술한다.

2.1 ToXGene

ToXgene은 XBench 성능 평가에 사용된 템플릿(template) 기반의 XML 데이터 생성기이다. 템플릿 화일은 생성하고자 하는 XML 데이터의 구조를 정의한 화일로서 엘리먼트 이름, 엘리먼트 사이의 포함관계, 사용자 데이터 화일의 이름, 출력 데이터 화일 이름, 출력 데이터 화일의 개수에 대한 정보 등이 기술된다. 템플릿 화일은 TSL(ToXgene Template Specification Language)[7]이라는 언어를 사용하여 작성된다. TSL은 XML 스키마[8]와 유사한 문법을 제공하여 XML 데이터 구조를 정의할 수 있도록 토론토 대학에서 정의한 언어이다. 사용자가 템플릿 화일을 직접 작성하기 위해서는 새로운 언어인 TSL을 별도로 습득해야 하는 불편함이 있다.

ToXgene은 XML 문서의 엘리먼트 내용으로 사용되는 텍스트 데이터를 가지는 사용자 데이터 화일을 이용하여 XML 데이터를 생성하거나, 미리 정의된 일부 데이터 타입을 사용하여 자동으로 데이터를 생성한다. ToXgene에서 사용자 데이터 화일을 이용하여 XML 데이터를 생성하는 경우 열 단위의 데이터 화일만을 사용해야 한다는 단점을 가진다.

2.2 Niagdatagen

Niagdatagen은 위스콘신 대학에서 개발된 XML 데이터 검색 및 질의 처리 시스템의 기능 평가를 위해 패스 트리(path tree)라고 부르는 자체 트리 구조 방식을 사용하여 XML 데이터를 생성한다. 사용자 데이터 화일을 사용하지 않고 입력 파라미터 화일만을 사용하여 XML 데이터 생성을 수행한다. 입력 파라미터 화일은 XML 데이터 출력 화일의 개수와 선택 추출 질의의 원활한 수행을 위해 필요한 화일 및 엘리먼트 이름에 대한 데이터 분포도를 가진다. 또한 XML 데이터의 단계(level) 및 팬아웃(fanout) 정보 등 총 20개의 파라미터를 가진다.

Niagdatagen의 입력 파라미터 화일에 기술되는 파라미터 값은 XML 데이터 출력 화일 이름을 제외한 나머지 값이 수치로 구성되어 사용자가 생성하고자 하는 XML 데이터 화일의 구조 정의나 변경이 쉽다. 하지만, 사용자가 가지고 있는 텍스트 데이터를 활용하여 다양한 XML 데이터를 생성할 수 있는 기능을 지원하지 못하고, 생성하고자 하는

XML 데이터의 크기를 사용자가 직접 제어할 수 있는 방법이 미약하다.

2.3 Xmlgen

Xmlgen은 XMark 성능 평가에 사용된 XML 데이터 생성기이다. Xmlgen은 데이터 생성기가 제공하는 네 가지 크기의 확장 요소(scale factor)에 따라 서로 다른 크기의 XML 데이터를 생성한다. 플랫폼(platform) 독립적인 데이터 생성을 수행하기 위해 Xmlgen은 특정 운영 체제나 하드웨어에서 수행되는 난수 생성 함수(random number generation function)를 사용하지 않고 난수 생성기를 따로 구현하여 XML 데이터 생성에 활용하였다.

Xmlgen은 미리 정의된 하나의 DTD와 7개의 단어 그룹을 이용하여 XML 데이터를 생성한다. 단어 그룹 내 단어들은 조합을 이루어 XML 데이터의 엘리먼트 내용을 구성한다. Xmlgen은 생성하고자 하는 XML 데이터의 구조를 사용자가 직접 정의할 수 없고, 사용자의 데이터 화일을 이용하여 XML 데이터를 생성하지 못하는 단점을 가진다. 또한, 하나의 XML 데이터 화일만을 생성하여 동일한 구조를 가지는 복수 개의 XML 데이터 화일을 이용하여 성능 평가를 수행하고자 하는 사용자의 요구 사항을 만족시키지 못한다.

2.4 Mbgen

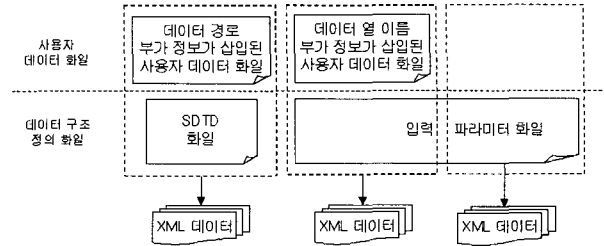
Mbgen은 Michigan 성능 평가 도구를 위해 개발된 XML 데이터 생성기이다. Mbgen은 DTD나 XML 스키마의 형식을 따르는 XML 문서 구조를 모두 지원하지만, 미리 정해진 하나의 데이터 구조 정의 화일만을 이용해야 한다는 단점을 가진다. Mbgen에서 생성되는 XML 문서의 엘리먼트 내용은 Mbgen에서만 사용하는 7가지 속성을 가지며, 속성 값은 일정 형태의 문자열과 숫자의 조합으로 구성된다.

사용자는 전체 엘리먼트의 수와 단계, 개별 엘리먼트 당 하위 엘리먼트 수를 조정하여 생성될 XML 데이터 크기를 결정한다. Mbgen은 Xmlgen과 마찬가지로 단일 XML 데이터를 생성하기 때문에 동일한 구조를 가지는 복수 개의 XML 데이터 화일을 이용하고자 하는 사용자의 요구를 충족시키지 못한다. 또한, 미리 정의된 구조 정의 화일만을 사용하기 때문에 사용자가 임의로 XML 데이터의 구조를 정의할 수 없다는 제약 사항을 가진다.

3. WeXGene

WeXGene은 사용자 데이터 화일 및 데이터 구조 정의 화일의 종류에 따라 세 가지 XML 데이터 생성 방법을 지원한다. 첫 번째 방법은 데이터 경로 부가 정보가 삽입된 사용자 데이터 화일과 데이터 구조 정의 화일로 SDTD 화일을 사용하여 사용자가 직접 정의한 XML 문서 구조에 맞게 데이터를 생성하는 방법이다. 두 번째 방법은 데이터 열 이름 부가 정보가 삽입된 사용자 데이터 화일과 파라미터 값으로 이루어진 간단한 데이터 구조 정의 화일인 입력 파

라미터 화일을 사용하는 경우이다. 마지막 방법은 사용자가 제공하는 데이터 화일 없이 입력 파라미터 화일만을 이용하여 XML 데이터를 생성하는 것이다.



(그림 1) WeXGene의 세 가지 데이터 생성 방법

3.1 사용자 데이터 화일

사용자가 제공하는 데이터 화일을 이용하여 WeXGene은 XML 데이터의 엘리먼트 내용이나 속성 값을 생성한다. 사용자 데이터 화일은 일반적으로 열로 구분되는 레코드 단위의 사용자 데이터를 가지는 텍스트 화일을 의미하며, 열의 구분을 위해 콤마(comma) 등의 구분 기호를 사용하고, 레코드 단위의 구분을 위해 캐리지 리턴(carriage return)을 사용한다. 사용자 데이터 화일은 사용자가 텍스트 데이터 생성기 또는 데이터베이스 시스템을 이용하여 생성하거나 임의로 작성하여 생성한다.

WeXGene은 사용자 데이터 화일에 삽입된 부가 정보를 참고하여 XML 데이터를 생성한다. 사용자 데이터 화일에 삽입되는 부가 정보는 두 가지 종류이다. 한 가지는 ‘데이터 경로’이며, 사용자 데이터 화일의 레코드 수와 사용자 데이터 화일의 내용이 XML 데이터의 엘리먼트 내용이나 속성 값으로 처음 삽입될 위치에 대한 정보, 그리고 사용자 데이터 화일의 구분 기호 등 세 가지 정보를 가진다. 데이터 경로는 ‘<레코드 수/엘리먼트 포함관계>구분기호’의 예와 같이 기술된다. 엘리먼트 포함관계는 ‘school/member/student’와 같이 나타내고 데이터가 삽입될 바로 상위 엘리먼트 이름까지만 정의한다. 사용자 데이터 화일은 열의 개수가 서로 다른 데이터를 가지며, 이를 각각 ‘데이터 집합’이라 한다. 즉, 하나의 사용자 데이터 화일에 다수 개의 데이터 집합이 존재할 수 있다. 사용자 데이터 화일에 존재하는 다수 개의 데이터 집합은 각 데이터 집합 당 한 개의 데이터 경로를 가진다. 사용자 데이터 화일의 끝에는 더 이상 데이터가 없음을 의미하는 ‘<0/>’와 같은 데이터 경로 정보를 가진다.

다른 한 가지 부가 정보는 ‘열 이름’이다. 열 이름은 사용자 데이터 화일이 입력 파라미터 화일과 함께 사용되어 XML 데이터를 생성할 때 사용되는 부가 정보이다. 입력 파라미터 화일에는 XML 데이터의 엘리먼트 이름이 기술되지 않기 때문에 각 열 이름을 XML 데이터의 엘리먼트 이름으로 사용하기 위함이다.

3.2 SDTD 화일

SDTD 화일은 XML 데이터의 구조 정의를 위해 일반적

으로 사용되는 DTD 파일에 비해 간략한 기호로 기술되는 화일이다. XML 데이터의 구조와 사용자 데이터 사용 여부를 정의하기 위해 사용된다. SDTD 화일은 다양한 단계를 갖는 엘리먼트 구조 및 다수 개의 속성을 정의함으로써 XML 데이터의 복잡한 구조 정의가 가능하다. SDTD 화일은 사용자가 보유한 DTD 화일을 WeXGene의 관련 모듈을 통해 자동으로 변환하거나 사용자의 직접 기술(description)을 통해 생성 가능하다.

사용자가 보유한 DTD 화일은 일반적으로 XML 데이터의 구조뿐만 아니라, 자식 엘리먼트의 다양한 선택 연산자 정의, 속성 값 유무 정의, 속성 값을 선언하는 다양한 유형의 정의 등 많은 내용들을 포함하는 복잡한 문법을 가지고 있다. 하지만, WeXGene은 엘리먼트 내용이나 속성 값으로 미리 준비된 사용자 데이터 화일을 이용하기 때문에, 부모 엘리먼트와 자식 엘리먼트간의 관계, 엘리먼트의 내용 값의 유무, 속성 값의 유무에 관련된 구조 정의만을 정의하면 된다. WeXGene을 이용하여 XML 데이터를 생성하기에 구조 정의 화일으로써 사용자가 가지는 복잡한 DTD의 문법을 모두 적용하여 생성하기에는 어려움이 있으며, 굳이 DTD에 정의된 내용이 XML 데이터 생성에 필요한 부분이 아니기 때문에, 간단히 정의할 수 있는 구조 화일로 SDTD를 제안한다.

<표 1>은 SDTD 화일에서 사용되는 각 기호를 설명한다. SDTD 화일에서 사용되는 기호는 DTD 화일이 정의하고 있는 기호를 참고하여 일부 사용하였지만, 그 의미가 동일하지는 않고, 그 작성법은 매우 간단하다.

<표 1> SDTD 화일의 기호

기호	설 명
!	속성을 포함하지 않는 엘리먼트를 의미한다.
?	속성을 포함하는 엘리먼트를 의미한다.
*()	하위 엘리먼트를 의미하며, 여러 개가 존재하는 경우 콤마로 이들을 구분한다.
)	하위 엘리먼트가 더 이상 존재하지 않는다.
#	엘리먼트가 내용을 포함하고 있다.
[]	속성을 의미하며 여러 개가 존재 하는 경우 콤마로 이들을 구분한다.
]	속성이 더 이상 존재하지 않는다.
;	하나의 정의에 대한 끝을 의미한다.

(그림 2)는 XML 데이터의 구조를 정의하고 있는 SDTD 화일을 보인다. (그림 2)에 기술된 (a) ~ (d)는 SDTD에서 기술될 수 있는 주요 구조 정의 형태를 나타내며, 음영으로 표시하였다. (c)의 하단에 있는 기호로 표시되지 않은 음영 부분인 'text'는 (d)의 설명을 위해 존재한다.

(그림 2)의 (a)는 루트(root) 엘리먼트가 다수개의 하위 엘리먼트를 가지는 경우이다. 루트 엘리먼트인 'site'가 하위 엘리먼트로 'regions', 'categories', 'catgraph'를 포함하고 있

음을 의미한다. (b)는 다수의 속성을 포함하는 엘리먼트의 형태를 나타내며, 'item' 엘리먼트가 'id'와 'no' 라는 두 개의 속성을 포함하면서 하위 엘리먼트로 'location', 'name', 'description', 'mailbox'를 가진다. (c)는 내용을 포함하는 엘리먼트의 표현 형태로서 (b)에서 선언된 'location' 엘리먼트가 내용을 포함하고 있음을 의미한다. 엘리먼트 선언 이후에 엘리먼트의 내용이 포함되어 있음을 별도로 정의하지 않은 엘리먼트는 내용을 가지지 않는다. (d)는 이전에 정의된 엘리먼트 내용의 재사용을 나타내는 예이며, 'mail' 엘리먼트의 하위 엘리먼트로 선언된 'text'가 'description' 엘리먼트의 하위 엘리먼트로 이전에 정의된 'text' 엘리먼트와 같은 내용을 가진다는 의미이다. 즉, SDTD 화일에서 같은 이름의 엘리먼트는 같은 내용을 가진다. 다른 내용을 가지는 엘리먼트를 정의하기 위해서는 엘리먼트 이름을 다르게 기술하여야 한다.

```

!site*(regions,categories,catgraph); ---- (a)
!regions*(africa,asia);
!africa*(item);
!asia*(item);
?item[id,no]*(location,name,description,mailbox); ---- (b)
!location#; ---- (c)
!name#;
!description*(text);
!text#;
!mailbox*(mail);
!mail*(from,to,date,text); ---- (d)
!from#;
!to#;
!date#;
!categories*(category);
?category[id]*(name,description);
!catgraph*(edge);
?edge[from,to]*;
    
```

(그림 2) SDTD 화일

3.3 입력 파라미터 화일

입력 파라미터 화일은 XML 데이터 생성을 위해 필요한 기본 정보를 정의하는 화일이다. 사용자가 XML 데이터의 구조를 단계별 엘리먼트 이름이나 속성까지 상세히 기술하지 않고 자신이 가진 텍스트 데이터를 활용하여 XML 데이터를 생성하고자 하거나, 사용자 데이터는 보유하지 않았지만 XML 데이터를 생성하고자 하는 경우 입력 파라미터 화일을 사용한다. 입력 파라미터 화일에 기술되는 파라미터의 종류는 사용자 데이터 화일과 함께 사용되는지 여부에 따라 다소 차이가 있다.

<표 2>는 입력 파라미터 화일의 파라미터 종류를 보인다. 사용자 데이터 화일을 사용하여 XML 데이터를 생성하는 경우에는 <표 2>에서 적용 구분이 '공통'인 파라미터와 '사용자 데이터 화일을 이용한 데이터 생성'인 파라미터들이 사용된다. 그리고 사용자 데이터 화일 없이 임의의 XML 데이터를 생성하고자 하는 경우에는 적용 구분이 '공통'인 파라미터와 '임의 데이터 생성'인 파라미터들이 사용된다.

〈표 2〉 입력 파라미터 화일의 파라미터 종류

적용 구분	파라미터 이름	파라미터 설명
공통	xmlFileName	XML 데이터 화일 이름
	xmlFileNum	XML 데이터 화일의 수
	rootEleName	XML 데이터 화일의 루트 엘리먼트 이름
사용자 데이터 화일을 이용한 데이터 생성	csvFileName	XML 데이터를 생성을 위해 이용한 사용자 데이터 화일 이름
	rowNum	XML 데이터 화일 하나에 생성할 행의 수
	fileSize	XML 데이터 화일의 크기
	rowName	XML 데이터 화일의 행을 구성하는 엘리먼트 이름
	colDelimiter	사용자 데이터 화일의 열 구분기호
입의 데이터 생성	eleLevel	엘리먼트 단계
	eleFanout	하나의 엘리먼트가 가지는 하위 엘리먼트의 개수

‘xmlFileNum’ 파라미터를 이용하여 다수 개의 XML 데이터 화일 생성이 가능하나, ‘xmlFileName’ 파라미터에 화일 이름은 한 개만 기술 가능하다. 따라서 생성되는 XML 데이터 화일의 이름은 ‘xmlFileName’ 파라미터에 기술된 화일 이름에 ‘1’부터 시작되는 일련번호가 덧붙여져 자동으로 생성된다. 다수 개의 XML 데이터 화일에 사용되는 루트 엘리먼트 이름으로는 사용자가 입력한 루트 엘리먼트 이름이 동일하게 사용된다.

사용자 데이터 화일을 이용하여 XML 데이터가 생성되는 경우, 하나의 사용자 데이터 화일의 이름이 ‘csvFileName’ 파라미터에 기술된다. 사용자 데이터 화일은 WeXGene이 지정한 특정 디렉터리에 위치하여야 한다. XML 데이터의 크기는 ‘rowNum’ 파라미터나 ‘fileSize’ 파라미터를 이용하여 결정한다. 단, XML 데이터의 크기를 결정하기 위해 두 가지 파라미터를 동시에 사용할 수 없다. 사용되지 않는 나머지 파라미터는 ‘*’ 문자를 값으로 가지도록 한다.

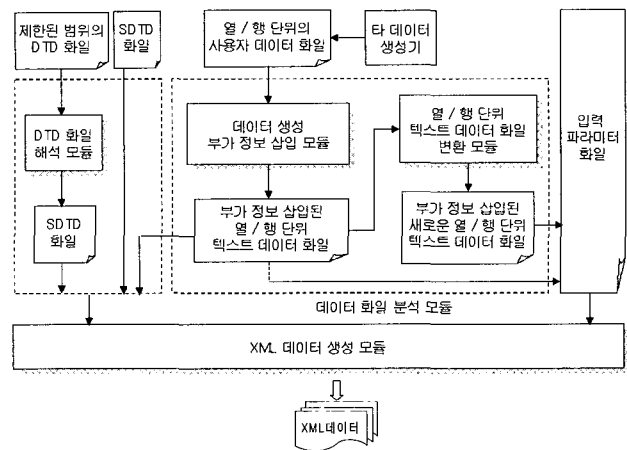
사용자가 ‘rowNum’ 파라미터를 지정하면 사용자 데이터 화일에서 지정된 행의 수만큼의 데이터를 이용하여 XML 데이터가 생성된다. ‘rowNum’으로 기술된 행의 수가 사용자 데이터 화일의 행의 수보다 많을 경우 사용자 데이터 화일의 처음부터 반복되어 적용되고, 적거나 같은 경우에는 해당되는 행의 수만큼의 사용자 데이터만 사용된다. ‘fileSize’ 파라미터는 생성하고자 하는 XML 데이터의 최소 크기를 의미한다. 즉, 생성된 XML 데이터 화일의 크기는 사용자가 파라미터에 기술한 화일의 크기와 같거나 크다. 루트 엘리먼트의 하위 엘리먼트에 사용되는 이름은 ‘rowName’ 파라미터에 지정된 엘리먼트 이름과 ‘1’부터 시작되는 일련번호의 조합으로 구성된다.

입의 XML 데이터를 생성하는 경우, XML 데이터는 WeXGene이 제공하는 입의 엘리먼트 이름 규칙과 엘리먼트 내용 규칙에 따라 생성된다. 엘리먼트 이름은 중복되지 않은 유일한 값을 가지도록 ‘+’ 기호와 영문자 및 숫자를 조합하여 구성한다. 엘리먼트 이름 구조는 ‘상위 엘리먼트 이름의 영문자’와 ‘새로운 영문자’ 그리고 ‘레벨을 의미하는 숫

자’의 조합으로 정의하고, 엘리먼트 내용은 ‘sm_입의 영숫자(alphanumeric)’와 같이 정의한다.

3.4 모듈 구성

WeXGene은 SDTD 화일 생성을 위한 DTD 화일 해석 모듈과 부가 정보가 삽입된 사용자 데이터 화일의 이용을 위한 데이터 화일 분석 모듈 그리고 XML 데이터 생성을 위한 XML 데이터 생성 모듈로 구성된다. (그림 3)은 각 모듈들의 세부 구성 및 모듈을 수행하기 위해 필요한 입력 화일과 모듈 수행 후의 출력 화일을 보인다.



(그림 3) WeXGene 모듈 구성

WeXGene은 XML 데이터의 구조 정의를 위해 SDTD 화일을 기본적으로 이용하지만, XML 데이터의 데이터 타입을 정의하는 표준 형태인 DTD 화일을 일부 지원한다. ‘DTD 화일 해석 모듈’은 사용자가 제한된 범위의 DTD 화일을 사용한 경우 이를 SDTD 화일로 자동 변환하는 작업을 수행한다. 사용자가 SDTD 화일을 직접 작성하여 사용한 경우에는 ‘DTD 화일 해석 모듈’은 사용되지 않는다.

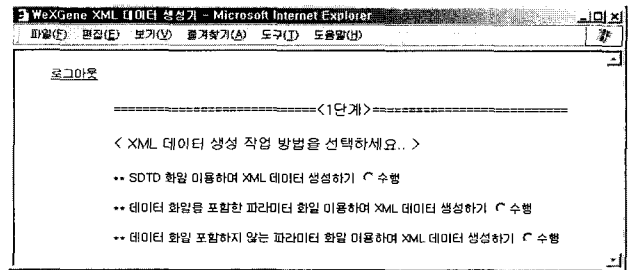
‘데이터 파일 분석 모듈’은 타 데이터 생성기로부터 생성되거나 사용자가 보유하고 있는 열/행 단위의 사용자 데이터 파일을 처리하는 모듈로서, ‘데이터 생성 부가 정보 삽입 모듈’과 ‘열/행 단위 텍스트 데이터 변환 모듈’로 구분된다. ‘데이터 생성 부가 정보 삽입 모듈’은 사용자가 사용하고자 하는 XML 데이터 생성 방법에 따라 앞서 3.1절에서 설명한 두 가지 종류의 부가 정보를 선택하여 데이터 파일에 삽입하는 작업을 수행하는 모듈이다. ‘데이터 경로’ 부가 정보는 엘리먼트의 포함관계 정의를 사용자에게 입력받고 사용자 데이터 파일의 레코드 수의 정의와 구분 기호, 그리고 파일의 끝을 의미하는 ‘<0/>’의 삽입 작업을 ‘데이터 생성 부가 정보 삽입 모듈’에서 자동 수행한다. ‘열 단위 및 행 단위 텍스트 데이터 파일 변환 모듈’은 부가 정보가 삽입된 하나의 행 단위 데이터 파일을 각 열 단위 데이터 파일로 분할하거나 부가 정보가 삽입된 여러 개의 열 단위 데이터 파일들을 새로운 하나의 행 단위 데이터 파일로 생성하는 작업을 수행하는 모듈이며, ‘열 이름’ 부가 정보가 삽입된 사용자 데이터 파일에 대해서만 처리 가능하다. ‘XML 데이터 생성 모듈’은 사용자가 선택한 XML 데이터 생성 방법에 적합한 파일들을 적용하여 XML 데이터를 생성한다.

3.5 사용자 인터페이스

WeXGene은 웹 기반의 사용자 인터페이스를 제공한다. 사용자에게 제공되는 웹 페이지는 총 13개이며, XML 데이터 생성 방법 선택 페이지, 3개의 XML 데이터 생성 실행 관련 페이지, 2개의 파라미터 파일의 생성 및 수정 페이지, 2개의 데이터 파일 부가 정보 입력 페이지, 행 단위의 데이터 파일의 열 단위 분할 페이지, 열 단위의 데이터 파일의 행 단위 병합 페이지, 3개의 생성 결과 데이터 확인 및 다운로드 페이지로 구성된다. (그림 4)는 WeXGene이 가지는 사

용자 인터페이스 흐름 및 구성을 간략히 보인다. 그림에서 화살표를 통해 웹 페이지간의 이동 가능 여부를 알 수 있으며, 다른 웹 페이지로의 화살표가 없는 웹 페이지는 해당 작업 수행 후 사용자의 선택에 따라 수행 작업을 반복하거나 웹 페이지를 종료함을 의미한다.

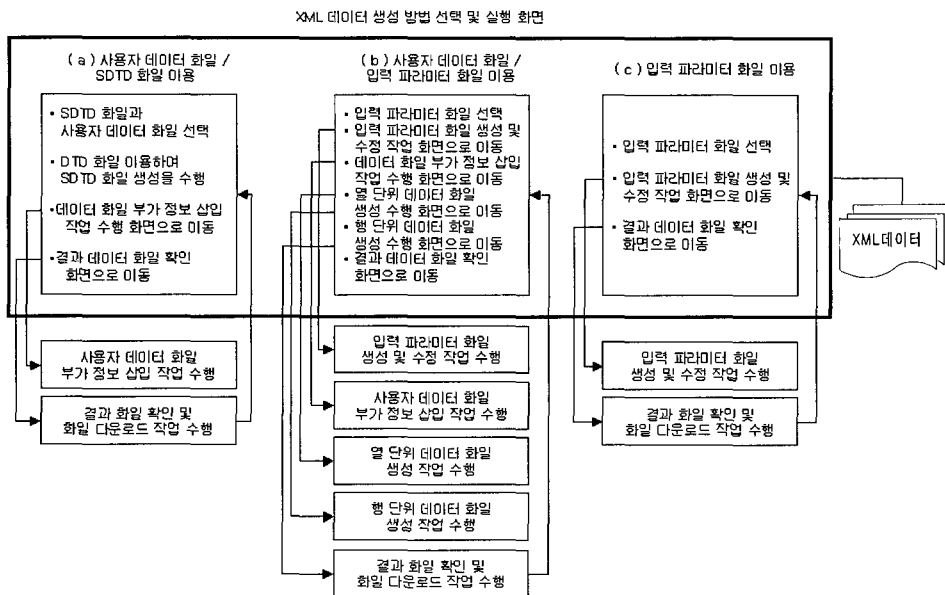
(그림 5)는 WeXGene의 사용자 로그인 후 수행 초기 화면이다. 사용자는 정해진 아이디와 패스워드를 사용하여 로그인 작업을 수행해야 하며, 성공적인 로그인 후 수행 초기 화면에서 XML 데이터 생성 방법을 선택한다. 그림에서 로그아웃을 사용자가 선택한 경우 로그인된 사용자 계정에서 로그아웃 작업을 수행하고 웹 페이지는 로그인 화면으로 변경된다.



(그림 5) WeXGene 실행 초기화면

3.6 타 XML 데이터 생성기와의 비교 분석

지금까지 발표된 XML 데이터 생성기 중 ToXgene, Niagdatagen, Xmlgen, Mbgen과 WeXGene의 기능을 비교하면 <표 3>과 같다. ‘O’는 해당 기능을 지원함을 의미하고, ‘X’는 해당 기능이 지원되지 않음, ‘△’는 해당 기능을 지원하되 제한된 형태로 지원함을 의미한다. <표 3>에 나열된 XML 데이터 생성기의 ‘기능’은 각 XML 데이터 생성기의 특징되는 기능들이다.



(그림 4) WeXGene의 사용자 인터페이스 흐름도

〈표 3〉 XML 데이터 생성기들의 기능 비교

기능	ToXgene	Niagdatagen	Xmlgen	Mbgen	WeXGene
사용자 데이터 화일 이용	△	X	X	X	O
XML 데이터 구조 제어	O	△	X	X	O
XML 데이터 크기 제어	X	X	△	△	O
결과 데이터 저장	O	O	O	O	O
임의의 XML 데이터 생성	X	O	X	X	O
복수 개의 데이터 화일 생성	O	O	X	X	O

‘사용자 데이터 화일 이용’ 항목은 사용자가 보유한 데이터를 이용하여 XML 데이터 생성이 가능한지 여부에 관한 항목이다. 비교 대상이 된 XML 데이터 생성기 중 대부분은 사용자 데이터를 이용해서 XML 데이터를 생성할 수 없고, 생성기가 보유한 한정된 소량의 데이터를 이용하거나, 임의의 XML 데이터 생성만이 가능하다. ToXgene은 열 단위의 사용자 데이터 화일만을 이용한다는 제한이 있다. WeXGene은 열 단위 사용자 데이터 화일은 물론 행 단위 사용자 데이터 화일을 이용한 XML 데이터 생성이 가능하다.

‘XML 데이터 구조 제어’ 항목은 XML 데이터 구조를 사용자가 직접 정의하여 사용자가 원하는 형태의 XML 데이터를 생성할 수 있는지 여부를 결정하는 항목이다. ToXgene은 템플릿 화일을 이용하여 XML 데이터의 구조 정의가 가능하지만, 사용자가 직접 화일을 작성하기에는 복잡하고 어렵다. Niagdatagen은 파라미터 화일을 통해 생성할 XML 데이터의 깊이, 하위 노드의 수 등을 정의할 수 있지만, 엘리먼트 이름이나 속성 값 정의 등은 불가능하다. Xmlgen과 Mbgen은 사용자가 XML 데이터 구조를 정의할 수 없다. WeXGene은 SDTD 화일을 사용한 다양한 구조의 XML 데이터 구조 정의 및 파라미터 화일을 사용한 간략한 XML 데이터 구조 정의가 가능하다.

‘XML 데이터 크기 제어’ 항목은 사용자가 원하는 크기의 XML 데이터 생성 가능 여부이다. XML 데이터베이스 시스템 성능 평가를 수행함에 있어 성능 평가 대상 시스템의 성능에 따라 필요한 XML 데이터의 크기가 다양해야 하기 때문에 ‘XML 데이터 크기 제어’ 항목은 XML 데이터 생성기의 기능 비교에 있어 중요한 항목이다. Xmlgen과 Mbgen은 정해진 몇 가지 확장 요소를 사용하여 XML 데이터 화일의 크기를 조절한다. ToXgene과 Niagdatagen은 XML 데이터 크기 제어가 불가능하다. 하지만 WeXGene은 사용자가 지정한 데이터 크기에 따라 XML 데이터를 생성할 수 있어 다양한 크기의 XML 데이터 생성이 가능하다.

‘결과 데이터 저장’ 항목은 사용자가 생성한 XML 데이터의 저장과 관련된 항목이다. 모든 생성기들이 결과 데이터를 저장할 수 있도록 기능을 지원한다. ToXGene은 미리 정해진 출력 디렉터리에 결과 데이터를 저장한다. Niagdatagen, xmlgen, Mbgen은 실행 화일 수행 디렉터리에 결과 화일을 저장한다. WeXGene도 미리 정해진 출력 디렉터리에 결과 데이터를 생성한다. 하지만, 생성된 날짜 및 시간의 조합으

로 구성된 디렉터리 이름의 하위 디렉터리에 생성하여 사용자에게 결과 화일 관리 편리성을 제공한다.

‘임의의 XML 데이터 생성’ 항목은 사용자가 XML 데이터 생성에 이용할 기본 데이터를 가지고 있지 않아도 XML 데이터 생성이 가능한지 여부이다. Niagdatagen과 WeXGene을 제외한 기능 비교 대상 XML 데이터 생성기는 모두 해당 기능을 지원하지 않는다. ‘복수 개의 데이터 화일 생성’ 항목은 사용자가 다수 개의 XML 데이터 화일을 생성할 수 있는지에 관한 항목이다. ToXgene, Niagdatagen, WeXGene은 사용자가 정의한 개수의 XML 데이터 화일 생성이 가능하다. Xmlgen과 Mbgen은 오직 하나의 XML 데이터 화일만을 생성한다.

4. WeXGene을 이용한 XML 데이터 생성

C 언어로 WeXGene을 구현하였으며, 사용자 인터페이스는 PHP 언어를 이용하여 작성하였다. XML 데이터 생성을 위해 사용된 PC는 Pentium III 600MHz CPU와 128MB 메모리를 가진다. 운영 체제로는 와우 리눅스 7.3 파란 릴리즈 2가 사용되었고, 웹 서버로는 아파치 웹 서버가 사용되었다. 본 장에서는 WeXGene의 세 가지 XML 데이터 생성 방법을 적용하여 XML 데이터를 생성하는 과정을 데이터 생성에 사용된 관련 화일 내용과 함께 살펴보고 생성 결과를 확인한다. XML 데이터 생성 방법에 따른 생성 내용과 그 결과를 보임으로써 사용자가 WeXGene의 데이터 생성 과정을 좀 더 쉽게 이해할 수 있도록 한다.

4.1 SDTD 화일과 사용자 데이터 화일을 이용한 XML 데이터 생성

(그림 6)의 (a)는 DTD 화일을 보이고 있고, (그림 6)의 (b)는 (a)에 보인 DTD 화일을 WeXGene의 DTD 화일 해석 모듈의 입력 화일로 사용하여 생성한 SDTD 화일을 보인다. (그림 7)은 사용자가 데이터 화일 부가 정보 입력 웹 페이지에 기술한 부가 정보를 WeXGene이 사용자 데이터 화일에 삽입한 형태를 보인다. (그림 7)의 부가 정보 삽입시 (그림 6)의 (b)에서 음영된 부분을 보면 처음으로 ‘item’ 엘리먼트가 ‘id’ 속성을 가지며 속성 값 설정을 위한 데이터를 필요로 하므로 부가 정보로 삽입되는 데이터 경로로는 ‘item’ 엘리먼트의 바로 상위 엘리먼트인 ‘africa’ 까지 사용자가 기

<pre> <!ELEMENT site (regions)> <!ELEMENT regions (africa)> <!ELEMENT africa (item)> <!ELEMENT item (location, quantity, name, payment, description, shipping, incategory, mailbox)> <!ATTLIST item id ID #REQUIRED> <!ELEMENT location (#PCDATA)> <!ELEMENT quantity (#PCDATA)> <!ELEMENT name (#PCDATA)> <!ELEMENT payment (#PCDATA)> <!ELEMENT description (text)> <!ELEMENT text (#PCDATA)> <!ELEMENT shipping (#PCDATA)> <!ELEMENT incategory EMPTY> <!ATTLIST incategory category ID #REQUIRED> <!ELEMENT mailbox (mail)> <!ELEMENT mail (from, to, date)> <!ELEMENT from (#PCDATA)> <!ELEMENT to (#PCDATA)> <!ELEMENT date (#PCDATA)> </pre>	<pre> !site*(regions); !regions*(africa); !africa*(item); ?item[id]*(location,quantity,name,payment, description,shipping,incategory,mailbox); !location#; !quantity#; !name#; !payment#; !description*(text); !text#; !shipping#; ?incategory[category]*; !mailbox*(mail); !mail*(from,to,date); !from#; !to#; !date#; </pre>
---	--

(a) cities.dtd 파일 내용

(b) cities.sDTD 파일

(그림 6) DTD 파일과 SDTD 파일

```

</site/regions/africa>,
item0,United States,1,duteous nine eighteen,Creditcard,page rous lady idle,Will ship
internationally,category540,Libero Rive mailto:Rive@hitachi.com,Benedikte Glew
mailto:Glew@sds.no,08/05/1999
item1,Moldova Republic Of,1,condemn,Creditcard,gold promotions despair,Will ship only within
country,category966,Javam Suwanda mailto:Suwanda@gmu.edu,Glew
mailto:Glew@cohera.com,11/28/1999
</>
                
```

(그림 7) cities.csv 파일 내용

술하여야 한다.

(그림 6)의 (b)에 기술된 SDTD 파일과 (그림 7)의 사용자 데이터 파일에 기술된 2개 행의 사용자 데이터를 이용하여 WeXGene은 XML 데이터 생성을 수행한다. XML 데이터는 'site'를 루트 엘리먼트로 가지고, (그림 6)의 (b)에서 음영 부분에 기술된 여러 하위 엘리먼트를 포함하는 2개의 'item' 엘리먼트가 생성된다. 생성된 XML 데이터는 (그림 8)과 같고, 동일한 구조를 갖는 'item' 엘리먼트는 태그 정보의 일부만을 보이고 그 내용은 생략한다.

(그림 7)의 내용 중 각 열 데이터들은 (그림 6)의 (b)에 기술된 SDTD 내용 중 데이터 값을 필요로 하는 엘리먼트 내용이나 속성 값에 매핑되어 순서에 맞게 생성된다. 이러한 생성 방법은 사용자가 데이터 파일만 가지고 있다면, 간단한 구조 파일을 통해 다양한 XML 데이터의 생성을 가능하게 한다. 사용자가 원하는 다양한 XML 데이터의 생성은 제한된 범위에서 XML 데이터를 생성했던 기존 생성기들에 비해 WeXGene이 가지는 특징이다. 예를 들어, 사용자가 (그림 8)과 같은 XML 데이터를 필요로 하는 경우, ToXgene은 복잡한 템플릿 작성 언어를 통해 템플릿 구조 파일을 생성한 후, 엘리먼트 내용 값으로 사용할 데이터들을 각 열 데이터 파일로부터 가져와야 한다. 생성된 결과 데이터는

(그림 8)과 동일한 구조는 가지지만, ToXgene이 데이터 파일로부터 랜덤하게 데이터를 가져와서 XML 데이터 생성에 사용하기 때문에, 엘리먼트 내용 값은 순서에 맞게 정확히 생성되지는 않는다. Niagdatagen은 내부적으로 데이터를 자동 생성하기 때문에 사용자가 원하는 데이터 파일을 이용하지 못하므로, (그림 8)과 같은 XML 데이터를 생성하지 못한다. Xmlgen과 Mbgen은 각 생성기에서 사용할 구조 정의 파일을 자체적으로 가지고 있고, 생성기 내부에 정의된 데이터만을 사용하기 때문에 사용자가 임의로 정의한 XML 데이터 결과 파일을 생성하지는 못한다. 이처럼 사용자가 원하는 구조의 결과 데이터 파일을 생성하는 WeXGene은 각종 XML DBMS 성능 평가에 적용하는 것이 다른 데이터 생성기에 비해 손쉽다.

서로 다른 열의 개수를 가지는 데이터를 하나의 사용자 데이터 파일에 포함하기 위해 WeXGene에 사용되는 사용자 데이터 파일은 여러 개의 데이터 집합을 포함한다. (그림 9)는 3개의 데이터 집합을 가지는 사용자 데이터 파일에 부가 정보인 데이터 경로를 각각 삽입한 내용을 보인다. 이를 이용한 결과 XML 데이터 파일은 (그림 10)에 보이고 있다. (그림 10)의 XML 데이터 생성 결과를 보면 'site'의 하위 엘리먼트로 'regions', 'categories', 'categori' 엘리먼트가 각각 위치한다.


```

<site>
  <regions>
    <africa>
      <item item="item0">
        <location>United States</location>
        <quantity>1</quantity>
        <name>duteous nine eighteen</name>
        <payment>Creditcard</payment>
        <description>
          <text>page rous lady idle</text>
        </description>
        <shipping>Will ship internationally</shipping>
        <incategory incategory="category540"></incategory>
        <mailbox>
          <mail>
            <from>Libero Rive mailto:Rive@hitachi.com</from>
            <to>Benedikte Glew mailto:Glew@sds.no</to>
            <date>08/05/1999</date>
          </mail>
        </mailbox>
      </item>
      <item item="item1"> ..... </item>
    </africa>
  </regions>
</site>

```

(그림 8) cities.xml 파일 내용

```

<2/site/regions/africa>,
item0,United States,1,duteous nine eighteen,Creditcard,page rous lady idle,Will ship
internationally,category540,Libero Rive mailto:Rive@hitachi.com,Benedikte Glew
mailto:Glew@sds.no,08/05/1999,virgin preventions half logotype
item1,Moldova Republic Of,1,condemn,Creditcard,gold promotions despair,Will ship only within
country,category966,Javam Suwanda mailto:Suwanda@gmu.edu,Glew
mailto:Glew@cohera.com,11/28/1999,back greg flay across sickness peter
<2/site/categories>,
category0,arm solemn weak finds,loyalty merit wicked plagued strangely devise doom
category1,coward sexton marrows,accesses praise measur hymen
<3/site/categraph>:
category476:category414
category36:category524
category586:category782
<0/>

```

(그림 9) compound.csv 파일 내용

```

<site>
  <regions>
    <africa>
      <item id="item0">
        <location> United States </location>
        .....
        <mailbox>
          <mail>
            <from>Libero Rive mailto:Rive@hitachi.com </from>
            .....
          </mail>
        </mailbox>
      </item>
      .....
    </africa>
    .....
  </regions>
  <categories>
    <category id="category0">
      <name>arm solemn weak finds</name>
      <description>
        <text>loyalty merit wicked plagued strangely devise doom</text>
      </description>
    </category>
    .....
  </categories>
  <catgraph>
    <edge from="category476" to="category414"></edge>
    .....
  </catgraph>
</site>

```

(그림 10) compound.xml 파일 내용

4.2 입력 파라미터 화일과 사용자 데이터 화일을 이용한 XML 데이터 생성

(그림 11)의 (a)는 사용자 데이터 화일을 사용하여 XML 데이터를 생성하기 위해 사용자가 파라미터 값을 설정하는 웹 페이지를 보인다. (a)에 설정된 값에 의해 생성되는 입력 파라미터 화일은 (그림 11)의 (b)와 같은 형태를 가진다. (그림 11)의 (c)는 사용자가 데이터 화일 부가 정보 입력 웹 페이지에 기술한 열 이름 부가 정보를 WeXGene이 사용자 데이터 화일에 삽입한 형태를 보인다.

입력 파라미터 화일 설정

- 입력 파라미터 화일 이름
- XML 데이터 화일 이름
- 부가 정보가 삽입된 데이터 화일 선택
- 사용자 데이터 화일의 열 구분 기호

- XML 데이터 화일의 수
- XML 데이터 화일 하나에 생성할 행의 수
- XML 데이터 화일의 크기 (KByte)

- XML 데이터 화일의 루트 엘리먼트 이름
- XML 데이터 화일의 행을 구성하는 엘리먼트 이름

(각 항목에 해당하는 값을 지정하지 않을경우에는 * 를 입력하세요)

(a) 입력 파라미터 화일 설정 웹 페이지

```
"ssm"
"dblab"
"2"
"*"
"1"
"dblab"
"member"
"
```

(b) dataparm.cfg 화일 내용

```
name,addr,semester,major
홍석후,경기도,석사4학기,웹로봇
강호연,인천,석사4학기,데이터마이닝
유승근,서울,석사4학기,데이터마이닝
```

(c) dblab.csv 화일 내용

(그림 11) 입력 파라미터 화일 설정 화면 및 XML 데이터 생성을 위한 화일

(그림 11)의 입력 파라미터 화일과 부가 정보가 삽입된 사용자 데이터 화일을 이용하여 WeXGene이 생성한 XML 데이터 화일의 내용은 (그림 12)와 같다. 즉, (그림 11)의 (a)에 기술된 'XML 데이터 화일 이름' 항목의 값인 'ssm' 문자열에 일련번호를 덧붙여 (그림 12)의 (a), (b)와 같이 두 개의 XML 데이터 화일이 생성된다. (그림 11)의 (c)에서 부가 정보로 삽입되어 있는 각 열 이름이 XML 데이터의 엘

리먼트 이름으로 생성된 형태를 볼 수 있다.

생성된 두 개의 XML 데이터 화일은 각각 루트 엘리먼트의 이름으로 'dblab' 문자열을 가지고, 행을 구성하는 엘리먼트 이름으로 'member' 문자열에 일련번호를 덧붙여 구성된 형태를 갖는 동일한 구조를 갖는다. 다만 두 XML 데이터 화일의 엘리먼트 내용만이 다르게 기술된다.

```
<dblab>
  <member1>
    <name>홍석후</name>
    <addr>경기도</addr>
    <semester>석사4학기</semester>
    <major>웹로봇</major>
  </member1>
  <member2>
    <name>강호연</name>
    <addr>인천</addr>
    <semester>석사4학기</semester>
    <major>데이터마이닝</major>
  </member2>
  <member3>
    <name>유승근</name>
    <addr>서울</addr>
    <semester>석사4학기</semester>
    <major>데이터마이닝</major>
  </member3>
  .....
  <member8>
    <name>강호연</name>
    .....
  </member8>
</dblab>
```

(a) ssm1.xml 화일 내용

```
<dblab>
  <member1>
    <name>유승근</name>
    <addr>서울</addr>
    <semester>석사4학기</semester>
    <major>데이터마이닝</major>
  </member1>
  <member2>
    <name>홍석후</name>
    <addr>경기도</addr>
    <semester>석사4학기</semester>
    <major>웹로봇</major>
  </member2>
  <member3>
    <name>강호연</name>
    <addr>인천</addr>
    <semester>석사4학기</semester>
    <major>데이터마이닝</major>
  </member3>
  <member4>
    <name>유승근</name>
    .....
  </member4>
  .....
</dblab>
```

(b) ssm2.xml 화일 내용

(그림 12) 수행 결과 XML 데이터

입력 파라미터 화일 설정

- * 입력 파라미터 화일 이름
- * XML 데이터 화일 이름
- * XML 데이터 화일의 수

- * XML 데이터의 엘리먼트 단계

* 마지막 단계의 엘리먼트는 하위 엘리먼트 수를 갖지 않는다.

- * 단계 1 엘리먼트의 하위 엘리먼트의 개수
- * 단계 2 엘리먼트의 하위 엘리먼트의 개수
- * 단계 3 엘리먼트의 하위 엘리먼트의 개수
- * 단계 4 엘리먼트의 하위 엘리먼트의 개수

- ** 생성 할 XML 데이터의 루트 엘리먼트 이름

(a) 입력 파라미터 설정 화면

```

"exam"
"10"
"5"
"1"
"2"
"3"
"4"
"root"
    
```

(b) param.cfg 화일 내용

<pre> <root> <a1> sm_1868-6 <aa2> sm_hold <aaa3> sm_1806-10-6 <a a a a 4 > sm_P6Y8M19D T18H54M36S </aaaa4> <a a a a b 4 > sm_wonder </aaab4> </aa2> <ab2> <abb3> sm_could <abba4> </pre>	<pre> <abb4> sm_P8Y5M2DT 15H58M57S </abb4> <abbc4> sm_1149-8 </abbc4> <abbd4> sm_million </abbd4> </abb3> <abc3> sm_1089-3 <abcd4> sm_compare </abcd4> </abc3> </ab2> </a1> </root> </pre>
--	--

(c) exam.xml 화일 내용

(그림 13) 파라미터 설정 화면과 생성된 입력 파라미터 화일 및 XML 데이터 화일

4.3 입력 파라미터 화일만을 이용하여 XML 데이터 생성

(그림 13)의 (a)는 임의 데이터를 WeXGene에서 생성하여 XML 데이터를 생성하기 위해 사용자가 파라미터 값을

설정하는 웹 페이지이다. 사용자가 (a)에 입력한 파라미터 값을 바탕으로 생성되는 입력 파라미터 화일의 내용은 (그림 13)의 (b)와 같고, 입력 파라미터 화일에 기반을 두어 WeXGene이 생성한 XML 데이터 화일은 (그림 13)의 (c)와 같은 내용을 가진다.

(그림 13)의 (c)를 보면 사용자가 루트 엘리먼트 이름으로 정의한 'root' 문자열을 루트 엘리먼트 이름으로 가지며, 단계 1 엘리먼트의 하위 엘리먼트 개수로 기술된 값에 따라 'root' 엘리먼트가 하위 엘리먼트로 'a1' 하나만을 포함하고 있음을 볼 수 있다. 또한, 단계 2 엘리먼트의 하위 엘리먼트 개수로 기술한 값에 따라 단계 1 엘리먼트인 'a1'은 하위 엘리먼트로 'aa'와 'ab'를 가지고 있음을 알 수 있다. 각 엘리먼트의 이름 명명과 엘리먼트 내용은 앞서 기술한 방법에 따라 생성되었다.

5. 결 론

본 논문에서는 XML 데이터베이스 시스템 성능 평가를 위해 웹 사용자 인터페이스를 제공하는 새로운 XML 데이터 생성기인 WeXGene의 모듈 구성 및 사용자 인터페이스와 XML 데이터 생성 과정 및 결과를 보였다. WeXGene은 열 단위 및 행 단위로 구성된 사용자 데이터 화일과 데이터 구조 정의 화일을 이용하여 XML 데이터를 생성하거나 사용자 데이터 화일 없이 임의의 XML 데이터를 생성하는데 사용한다. 또한, 행 단위 사용자 데이터 화일 이용 등 다른 XML 데이터 생성기의 문제점을 해결하여 사용자가 쉽게 XML 데이터를 생성할 수 있도록 다양한 기능을 제공한다.

또한 본 논문에서는 WeXGene에서의 XML 데이터 구조 정의를 위해 SDTD를 기술하였다. SDTD는 DTD의 일부를 기호화함으로써 일반 사용자들이 쉽게 XML 데이터의 구조를 정의할 수 있도록 하였다. WeXGene의 SDTD 화일 변환 기능을 통해 SDTD에 대한 숙지 여부에 관계없이 사용자가 XML 데이터 구조에 맞는 데이터를 생성할 수 있으며, 또한 사용자가 직접 SDTD 화일을 작성하여 복잡한 구조를 갖는 XML 데이터를 쉽게 생성한다.

WeXGene 사용자 인터페이스 개발 시 초기에는 사용자가 입력해야 할 모든 내용을 하나의 웹 페이지에서 입력할 수 있도록 구성하여 여러 XML 데이터 생성 방법 중 사용자가 자신이 원하는 방법을 선택하여 해당 부분의 내용을 입력하도록 하였다. 이러한 인터페이스는 한 화면에서 전반적인 XML 데이터 생성 관련 입력 내용을 관리할 수 있다는 장점이 있었지만, 사용자는 자신이 원하는 생성 방법에 따라 입력해야 할 내용에 비해 많은 항목들로 인한 혼란을 겪어 단계별 웹 페이지 입력 구성 방법으로 최종 구현하였다. 단계별 웹 페이지 입력 구성 방법은 사용자가 원하는 XML 데이터 생성 방법에 따라 해당되는 입력 항목만을 사용자에게 단계적으로 제시하여 사용자가 많은 입력 항목들로 인해 혼란을 겪는 것을 방지하고자 하는 방법이다.

향후 연구에서는 XML 언어 자체와는 상이한 문법 및 구

조를 이용하여 정의된 DTD에 비해 XML 언어를 사용하여 구조를 정의한 DTD보다 유연한 특성을 지니고 있는 XML Schema의 지원이 가능하도록 할 것이다. 또한 대용량 XML 데이터 생성 시 데이터 생성 시간은 사용자가 XML 데이터 생성기를 평가하는 중요한 측정 요소가 되므로 WeXGene의 각 데이터 생성 방법에 따른 데이터 생성 알고리즘을 최적화하여 데이터 생성 시간이 감소되도록 개선하고, 이를 기반으로 하여 다른 XML 데이터 생성기와의 성능 비교 평가를 수행할 것이다.

참 고 문 헌

[1] B.B. Yao, M.T. Ozsu, and J. Keenleyside, "XBench-A Family of Benchmarks for XML DBMSs," Proceedings of the Efficiency and Effectiveness of XML Tools and Techniques and Data Integration over the Web-Revised Papers, pp.162-164, 2003.

[2] A. Schmidt, F. Waas, M. Kersten, M.J. Carey, L. Manolescu, and R. Busse, "XMark: A Benchmark for XML Data Management," Proceedings of the 28th International Conference on Very Large Databases, pp.974-985, 2002.

[3] K. Runapongsa, J.M. Patel, H.V. Jagadish, Y. Chen, and S. Al-Khalifa, "The Michigan Benchmark: Towards XML Query Performance Diagnostics," Technical Report, University of Michigan, 2002.

[4] 옥은택, 정회진, 이상호, "데이터베이스 벤치마크를 위한 데이터 생성기와 성능 평가", 정보처리학회논문지D, 제10-D권 6호, pp.907-916, 2003.

[5] D. Barbosa, A. Mendelson, J. Keenleyside, and K. Lyons, "ToXgene: An Extensible Template-based Data Generator for XML," Proceedings of the 5th International Workshop on the Web and Database, pp.49-54, 2002.

[6] A. Aboulnaga, J.F. Naughton, and C. Zhang, "Generating Synthetic Complex-structured XML Data," Proceedings of the 4th International Workshop on the Web and Databases, pp.79-84, 2001.

[7] D. Barbosa, "ToXgene Template Specification Language," Department of Computer Science, University of Toronto, Version 2.1, 2003.

[8] D.C. Fallside. "XML Schema part 0: Primer - W3C candidate recommendation," <http://www.w3.org/TR/xmlschema-0/>, 2000.

[9] A.R. Schmidt, F. Waas, M.L. Kersten, D. Florescu, M.J. Carey, I. Manolescu, and R. Busse, "Why and How to Benchmark XML Databases," ACM SIGMOD Record, Vol.3, No.30, pp.27-32, 2001.

[10] A. Aboulnaga, A.R. Alameldeen, and J.F. Naughton, "Estimating the Selectivity of XML Path Expressions for Internet Scale Applications," Proceedings of the 27th International Conference on Very Large Databases, pp. 591-600, 2001.



신 선 미

e-mail : myfreshness@hotmail.com
 2000년 조선대학교 컴퓨터공학과 (학사)
 2005년 숭실대학교 대학원 컴퓨터학과 (석사)
 1999년~2000년 (주)SMIT 개발팀
 2000년~2002년 (주)ELEART 개발팀
 2005년~현재 (주)CNP차앤박화장품 IT팀
 관심분야 : 벤치마크, XML 데이터베이스



정 회 진

e-mail : sinclear@dreamwiz.com
 1993년 우석대학교 전산학과(학사)
 1995년 숭실대학교 대학원 컴퓨터학과 (석사)
 1995년~2000년 (주)핸디소프트 기술연구소 선임연구원
 2002년~2003년 숭실대학교 정보미디어 연구소, 전임연구원
 2000년~현재 숭실대학교 대학원 컴퓨터학과 박사과정
 관심분야 : 데이터베이스 시스템 성능 평가 및 튜닝, XML 데이터베이스



이 상 호

e-mail : shlee@comp.ssu.ac.kr
 1984년 서울대학교 전산공학과 (학사)
 1986년 미국 노스웨스턴대 전산학과 (석사)
 1989년 미국 노스웨스턴대 전산학과 (박사)
 1990년~1992년 한국전자통신연구원 선임연구원
 1999년~2000년 미국 조지메이슨대, 소프트웨어정보공학과, 교환 교수
 1992년~현재 숭실대학교 컴퓨터학부 교수
 관심분야 : 인터넷 데이터베이스, 데이터베이스 시스템 성능 평가 및 튜닝