

XML/XSL 기술을 이용한 범용 데이터 폼 편집기 개발 연구 - 범용 데이터 폼 편집기 -

김주한

하버드외대 의료정보학

초 록

XML(Extensible Markup Language)과 XSL(Extensible Stylesheet Language)은 새로 개발 소개된 인터넷 표준이다. 인터넷 기술의 발전에 힘입어 임상 정보 시스템이 빠른 속도로 웹기반 환경으로 옮겨가고 있다. 임상 정보 시스템의 구축과 운영에서 자료 입력 양식의 개발과 유지에는 많은 프로그래밍 자원이 소요된다. 고도의 개별화 및 필요에 따른 수정 갱신이 가능한 범용 자료입력 양식의 개발은 복잡하고 급변하는 임상 정보 시스템 사용자의 요구를 수용하고 의료 정보 시스템 구축의 효율성을 증대시킬 것이다. 템플레이트 등을 이용한 자료 입력 양식의 표준화 연구들이 이러한 요구를 부분적으로 수용하는 것으로 보고되었으나, 여전히 입력 양식 구성 각 성분의 데이터형 및 행동양식을 사전에 정의해야 한다는 제한이 있다. 본 논문은 웹기반 의료 정보 시스템 개발에 XML/XSL 기술을 응용하여, 프로그램 개발 이후에도 데이터형 및 행동 양식의 사후정의를 가능하게 하는 범용 데이터 폼 편집기를 기술한다. XML/XSL의 자료 추상화 능력의 적용은 임상 정보 시스템을 고도로 유연한 컴포넌트화 하는 것을 촉진할 것으로 기대된다. (대한의료정보학회지 6-2, 65~72, 2000)

Keywords : XML/XLS, 정보시스템, 자료입력, 인터넷

I. 서 론

인터넷 기술의 활발한 보급으로, 대규모 정보 시스템들도 웹기반 클라이언트-서버 환경으로 급속히 이전되고 있다. XML(Extensible Markup Language)은 SGML(Standard General Markup Language)이 제공하는 고도의 확장성, 자료 구조화, 자료검사 능력 등

의 기능을 인터넷 환경에서도 사용할 수 있게 하기 위해 새로 개발된 표준 인터넷 브로토폴이다¹⁾. XSL(Extensible Stylesheet Language)은 자료의 구조를 주로 기술하는 XML 언어의 제한점을 보완하기 위해 문서의 스타일쉬트를 기술할 수 있도록 개발된 언어이며, 문서 변환을 기술하기 위한 XSLT(Extensible Stylesheet Language Transformation)와 형식 의미론

주소: Douglas Porter Fellow in Medical Informatics Center for Clinical Computing
Harvard Medical School 21 Autumn Street Boston, MA 02215
전화: 1-617-632-0138 Fax: 1-617-632-0138 E-mail: jhkim@caregroup.harvard.edu

(format semantics)을 기술하기 위한 XML 어휘록으로 구성되어 있다.

효율적이고 다양한 자료 입력 양식의 개발은 대형 정보 시스템의 구축의 성패를 좌우한다. 이러한 자료 입력 양식의 개발은 정보 시스템 사용자 구성이 다양하며, 사용자 요구사항이 복잡하고 변화가 빠를수록 더욱 많은 프로그램 자원과 유지비용을 요한다(2). 대규모 정보 시스템 개발시, 자료 입력 양식의 중요성을 간과하여 그 구성을 단순 획일화하는 경우도 많다. 그러나 시간에 쫓기는 임상 의사가 사용하는 정보시스템에서 사용자 인터페이스의 효율성은 때로 시스템의 성패를 좌우하며, 자료 입력 양식의 경직성은 임상 의사가 정보 시스템 수용을 거부하는 중요한 원인이 된다. 또한, 임상 의료 정보 시스템이 다루는 자료는 그 종류와 범위가 매우 다양하여 복잡한 입력 양식 구성 성분을 필요로 한다. 고도로 개별화가 가능하고 필요에 따라 손쉽게 유연하게 재구성할 수 있는 자료 입력 양식의 개발은 사용자의 변화무쌍한 요구에 적응하는 것을 가능하게 함으로서, 의료 정보 시스템 구축과 관리의 효율성을 크게 향상시킬 것으로 기대된다.

인터넷과 그림 사용자 환경의 빠른 보급과 표준화를 통해, '라디오 버튼', '편집 상자', 혹은 '콤보 박스'와 같은 표준화된 자료 입력을 위한 기초 구성 성분들이 널리 받아들여져 사용되고 있다. 그러나 좀더 효율화된 시스템, 예를 들어, 컴퓨터 기반 임상 면접 시스템 등은 이러한 기초 구성 성분들을 필요에 따라 적절하게 조합하여 필요한 복합 구성 성분들을 사전에 정의하여 사용함으로써 프로그램 개발의 표준화와 재사용성을 높인다³⁾. 예를 들어, 다지선다형 문제를 형식화하는 경우에 있어서, 단순히 그때 그때의 필요에 따라 텍스트와 라디오버튼 등의 기초 구성 성분들을 조합하여 프로그래밍할 수도 있지만, 사용자의 입력을 기다리는 '문제'라는 클래스 자체를 추상화하고 그 클래스의 하위 클래스의 하나로서 다지선다형 문제를 형식화 한 후, 해당 객체의 데이터형과 실제 어플리케이션에서 표현되는 다양한 행동 양식을 정의하는 객체지향식 프로그래밍 기법을 사용하면, 프로그램의 재사용성과 모듈화를 의미있

게 증가시킬 수 있다.

더 나아가 입력 화면 전체를 표준화하는 템플레이트를 이용한 고속 프로토타입화 장치들도 이러한 자료 입력 양식의 신속한 개발 및 적용 과정을 상당히 촉진할 수 있는 것으로 보고되었다⁴⁾. 그러나 이러한 방법들은 필수적으로 시스템의 개발 전에 사료 입력 양식 구성 단위들을 사전 정의해야 한다는 한계를 가진다. 즉, 객체지향식 프로그래밍 기법을 적용한 경우에도 일반적으로 각 객체의 정의는 프로그램 코드내에 정의되며, 객체를 데이터화하여 데이터베이스에 저장된 경우에도 각 객체를 기술하는 데이터는 데이터베이스 스키마에 사전 정의되어야 하므로, 시스템이 구축된 이후에 새로운 자료 유형의 사용이 필요해진 경우에는 프로그램의 수정 혹은 데이터 구조의 수정이 불가피하다. 한 층 더 적극적인 접근 방식은 각각의 입력양식 혹은 입력화면 등의 단위를 넘어서 전체 데이터 레포지토리를 단일 클래스와 그 하부구조로 정의하는 완전한 객체지향적 추상화일 것이다. 본 논문에서는 XML/XSL 기술을 이용하여, 데이터 추상화 계층을 한 단계만 더 삽입하여, 객체의 정의 중 데이터화가 가능한 자료 부분은 범용 데이터베이스 스키마로 모델링하고, 데이터화가 난이한 프로시저 부분은 스타일시트 기술 언어인 XSL 문법으로 분리 기술하는 방식으로 객체를 형식화하고, 어플리케이션은 객체 생성을 위한 범용 인터프리터의 역할을 함께 수행하는 객체 정보 처리 과정들을 정의함으로써, 정보 시스템의 구축 이후에라도 새로운 자료 입력 양식 구성 단위를 사후 정의하는 것을 가능하게 하는 방법을 사용하는 실험로서, 범용 자료 입력 폼 편집 시스템을 기술한다.

II. 재료 및 방법

프로그램의 개발은 마이크로소프트사의 윈도우즈 NT 4.0 환경하에서 MS-SQL Server 7.0, Microsoft Active Server Page Extensior, Microsoft Visual Basic 6.0, Microsoft Internet Explorer 5, Microsoft XML 2.5 SDK를 사용하였으며, SDK XML Editor는 Techno2000 USA사의 CLIP! XML Editor, XSL Editor는 Excelon Corporation사의 eXcelon Stylus를 사용하였다. 제품들의 상호 호환성의 문제가 남아 있었

으며, 필요한 부분은 텍스트 편집기로 직접 편집하였다.

웹 기반 정보시스템의 사용자 환경은 브라우저 상에서 열람할 수 있는 페이지 형식의 트랜잭션 양식을 제공한다. 페이지는 다시 텍스트와 멀티미디어로 구성된 기술 부분과 다양한 자료 입력 양식 유형들로 구성된 상호작용 부분으로 구성된다. 객체 추상화의 가장 상위 수준인 페이지와 바로 다음 하위수준인 기술 부분과 상호작용 부분을 XML DTD(Data Type Definition)로 정의하였다. 이러한 정의는 "객체 특성-가치"의 객체 지향 모델을 따라서 일관되게 수행했으며, 실제적인 구축은 Johnson이 제시한 방법을 참고로 하여⁽⁶⁾, 범용성 있는 자료 모형을 관계형 데이터 베이스 스키마에 구현하였다(Fig 1). Johnson이 논의한 바와 같이 이러한 접근은 데이터베이스 스키마를 고도로 단순화하여 시스템의 범용성을 보장한다.

그러므로 관계형 데이터베이스의 스키마는 XML DTD의 역할을 하고, 데이터베이스에 저장된 각각의 자료들은 XML 파일에서 태그와 태그 사이에 놓이는 실제 내용 부분 혹은 태그 내의 내용물을 표현하기 위한 애트리뷰트의 값의 역할을 하게 된다. 데이터베이스의 입출력 계층 바로 위에 상주하도록 고안된 XML 생성 모듈은 응용 프로그램의 요청에 의한 데이터베이스의 출력이 있을 때마다 실제 필요한 XML 파일을 동적으로 생성한다.

XML DTD에 의해 정의된 객체는 그 객체를 호출하는 응용 프로그램의 모드에 따라 다양한 유형의 행동 양식을 보일 수 있다. 이 행동의 규칙은 XSL 파일에서 제공하는 프로시저 기술 언어로 작성하고, 이 과정은 공개된 XSL 편집기로 수행할 수 있다. 생성된 XSL 파일은 그림 1의 XSL 입력 양식 구성 성분 레포지토리에 저장된다. 데이터베이스 스키마를 범용성 있게 모델링하여 서로 이질적인 데이터 형들

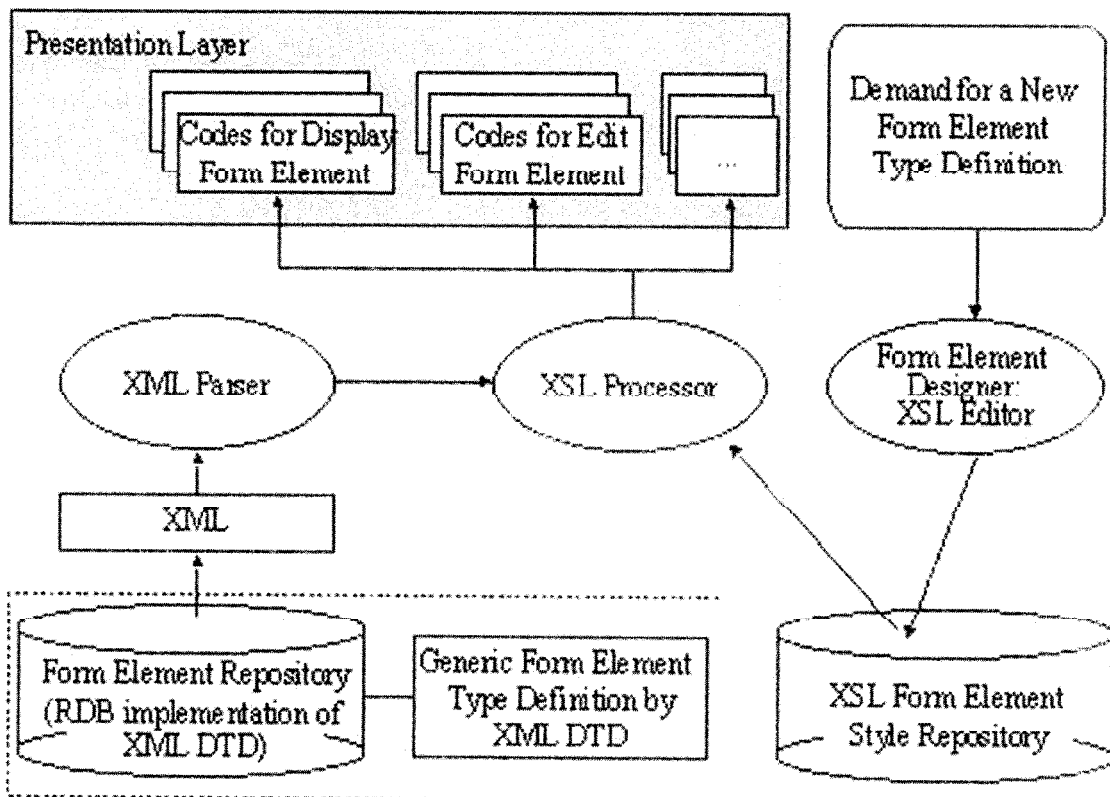


Fig 1. System architecture of generic data form designer. Data part of form object is stored in relational database and its procedural definition in XSL files so as to enable the post-hoc definition of new object types with the help of XML-DTD and XSL Processor.

을 일관된 레포지토리에 구현하고, 응용 프로그램의 모드가 쓰기, 읽기, 편집 등의 다양한 다른 모드일 때 그 XML 데이터 객체가 서로 다른 행동 유형을 보이는 것을 XSL 언어로 기술함으로써, 정보의 데이터화가 용이한 객체의 자료 부분은 관계형 데이터베이스에 저장하고, 정보의 데이터화가 어렵고 프로시저 형식으로 기술하는 것이 적절한 객체의 프로시저 부분은 XSL 스타일 언어로 기술하는 것이 가능하다. 실제로 XML DTD는 최상위 수준의 추상화된 객체 클래스의 형식만을 정의하고, 각 인스턴스 데이터 부분은 관계형 데이터베이스에 저장하며, XSL 스타일 언어는 실제 객체 인스턴스의 프로시저 부분을 정의하게 되는 것이다. 그러므로 응용 프로그램은 존재 가능한 모든 데이터 형과 그 행동 양식을 판독할 수 있는 범용 인터프리터의 역할을 한다.

관계형 데이터베이스와 XML DTD의 상호작용에 의해 동적으로 생성된 XML 파일은 XML 파서에 의해 처리되고, XSL 프로세서는 생성된 XML 파일과 이에 대응되는 XSL 파일을 읽어들이며 필요한 HTML 코드로 렌더링한다. HTML 코드 부분은 객체를 요청한 응용 프로그램의 모드에 따라 다양한 방식으로 동적 생성된 객체 인스턴스에 해당되며, 이 과정은 XSL 파일에 스타일 언어로 기술된 객체 인스턴스의 모드별 다양한 행동양식 정의에 기반한다. 생성된 HTML 코드 부분은 실제 최종적으로 응용 프로그램에 의해 생성되는 HTML 문서의 필요한 각 부분을 구성하는 컴포넌트 성분으로 동작하게 된다.

그러므로 자료 입력 양식의 최상위 클래스는 XML DTD로 정의하고, 각 객체 인스턴스의 생성은 그 객체를 요청한 응용 프로그램의 모드에 따라 동적으로 생성되는 XML 파일과 그에 대응되는 XSL 파일에 의해 생성된 HTML 코드로 이루어진다. 이때, XML 파일을 해당 객체의 데이터화 가능한 내용 부분을 정의하고, XSL 파일은 객체의 모드별 행동 양식을 표현하게 된다. XSL이 제공하는 폭넓은 스타일 기술 언어에 의해 거의 모든 유형의 HTML 코드 생성이 가능하고 이러한 과정들은 공개된 모듈들을 직접 이용할 수 있어 시스템을 범용화 하는 동시에 고도로 컴포넌트화 하는 것이 가능하다.

III. 결 과

XML/XSL 기술에 기반한 범용 자료 입력 폼 편집기를 구현하였다. XML의 자료 추상화 기능과 XML 파서, XSL 프로세서 등의 공개된 모듈을 이용한 시스템 구조는 객체의 데이터 특성이나 행동 특성의 수정이 필요한 경우 프로그램 코드 자체를 수정하지 않고도 각 객체에 해당하는 XSL 파일을 수정하는 것만으로 필요한 업무를 수행할 수 있게 해준다. 뿐만 아니라, 프로그램 개발시에는 전혀 고려하지 못했던 새로운 자료 유형의 신규 모델링이 필요해진 경우에는 더욱더 유용하다. 예를 들어, 입력 폼 구성 성분으로서 주관적 판단의 강도를 눈금자 형식으로 표현하는 라이커트 스케일(Likert Scale)이나 그 판단의 강도를 색상이나 명암과 같은 시각적인 아날로그 지표로 표현하는 비주얼 아날로그 스케일과 같은 새로운 복합 자료 입력 구성 성분의 추가가 절실한 상황이 된 경우, 기존의 시스템 구성에서는 프로그램의 모듈화 수준에 무관히 프로그램 코드 혹은 원 데이터베이스 구조를 직접 수정해야 한다는 한계가 있다. 그러나 Fig 1과 같은 유연한 시스템 구성에서는 단지 새로운 자료 유형의 데이터 특성과 행동 특성을 정의하는 XSL 파일을 하나 더 작성하여 XSL 입력 양식 구성 성분 레포지토리에 저장하기만 하면 되는 것이다. 응용 프로그램이 신규 자료 입력 구성 성분을 호출하는 것만으로 새로운 객체의 인스턴스가 생성된다.

XSL 파일의 작성은 현재 개발된 공개된 XSL 에디터를 사용하며 다양한 모듈이 공개되어 있다. XSL 입력 양식 구성 성분 레포지토리는 단순한 파일 시스템으로 구성하거나 혹은 관계형 데이터베이스에 구조화시켜 저장할 수 있다. 동적 XML 파일 생성기는 각각의 XML DTD에 따라 데이터베이스 스키마와 함께 모듈을 작성하였다. 이론적으로 XML DTD의 정의만으로 데이터베이스 스키마와 XML 파일의 동적 생성 전 과정을 자동화 할 수 있다. 본 연구에서는 다양한 문답 양식을 가진 웹기반 컴퓨터 환자 임상 인터뷰 프로그램에 상기 기술한 개념을 실제 적용하였다. 그 형식화를 위한 XML DTD를 Fig 2에 표기하였다. XML 파서와 XSL 프로세서도

```

<?xml version="1.0"?>
<?xml:stylesheet type="text/xsl" href="WebCONVERSE,xsl"?>
<!DOCTYPE conversation [
<!-- XML DTD definitions for WebCONVERSE communication system: a database implementation
of the generic type for computer-based communications -->
<!ELEMENT dialogue (header, author_information, body)>
<!ELEMENT header (dialogue_id, dialogue_name, type, interview_name, interview_version,
interview_page, interview_no, heading?, intention)>
  <!ELEMENT dialogue_id (#PCDATA)>
  <!ELEMENT dialogue_name (#PCDATA)>
  <!ELEMENT type (#PCDATA)>
  <!ELEMENT interview_name (#PCDATA)>
  <!ELEMENT interview_version (#PCDATA)>
  <!ELEMENT interview_page (#PCDATA)>
  <!ELEMENT interview_no (#PCDATA)>
  <!ELEMENT heading (#PCDATA)>
  <!ELEMENT intention (#PCDATA)>
  <!ELEMENT required (#PCDATA)>
  <!ELEMENT required_message (#PCDATA)>
<!ELEMENT author_information (author_id, name?, organization?, department?, date)>
  <!ELEMENT author_id (#PCDATA)>
  <!ELEMENT author_name (first_name, middle_name?, last_name)>
    <!ELEMENT first_name (#PCDATA)>
    <!ELEMENT middle_name (#PCDATA)>
    <!ELEMENT last_name (#PCDATA)>
  <!ELEMENT organization (#PCDATA)>
  <!ELEMENT department (#PCDATA)>
  <!ELEMENT date (#PCDATA)>
<!ELEMENT body (text, text_prefix?, items, text_suffix?)>
  <!ELEMENT text_prefix (#PCDATA)>
  <!ELEMENT text (#PCDATA)>
  <!ATTLIST text font_face CDATA #IMPLIED
                font_size CDATA #IMPLIED
                font_color CDATA #IMPLIED>
  <!ELEMENT text_infix (#PCDATA)>
  <!ELEMENT items (item+)>
    <!ELEMENT ITEM (item_label?, item_prefix?, item_character_infixes?, item_word_infixes?,
                    item_content, item_suffix?)>
    <!ELEMENT item_label (#PCDATA)>
    <!ELEMENT item_prefix (#PCDATA)>
    <!ELEMENT item_character_infixes (item_character_infix)+>
    <!ELEMENT item_character_infix (#PCDATA)>
    <!ELEMENT item_word_infixes (item_word_infix)+>
    <!ELEMENT item_word_infix (#PCDATA)>
    <!ELEMENT item_content (#PCDATA)>
    <!ELEMENT item_suffix (#PCDATA)>
  <!ELEMENT text_suffix (#PCDATA)>
]

```

Fig 2. XML DTD implementation of a form object for web-based generic form authoring program. Form object is generically defined as any input form waiting for user input and consists of header, author, and body informations. Body of form consists of text and items with corresponding prefixes and suffixes. Multiple items are allowed and can be decorated with a variety of HTML tags generated by XSL definition files.

공개된 모듈을 사용할 수 있다. 실제 프로그램 개발에서는 XSL 파일의 수정 혹은 추가만으로 시스템의 수정 없이 광범위한 개발 작업을 수행할 수 있다.

IV. 고 찰

본 논문에서는 새로이 도입되고 있는 인터넷 표준 프로토콜의 하나인 XML 언어의 데이터 구조 추상화 능력과 XML 언어의 제한점을 보완하기 위해 개발된 스타일시트 언어인 XSL 언어를 통한 데이터 객체의 행동 양식 기술의 상호 작용을 활용하여, 효율적 자료 입력 양식 개발에 필수적인 입력 폼 구성 성분 레포지토리를 정보 시스템 개발 이후에도 사후 정의하는 것이 가능한 범용 데이터 폼 편집기를 개발하였다. 자료 구조의 객체화한 데이터 표현 기법과 객체 사후 처리 기법은 클라이언트 응용 프로그램이 도메인 엔티티를 범용성 있게 모델링 하는 것을 가능하게 하고, 응용 프로그램 개발이 완료된 이후에도 그 구성 컴포넌트를 유연하게 사후 정의 혹은 수정 하는 것을 가능하게 할 것으로 기대된다.

의료 정보 교환의 가장 대표적인 HL7의 버전 3.0에서 SGML과 그 부분 집합인 XML이 정보 전송 문법으로 채용되고⁽⁶⁾⁽⁷⁾, 구조화된 보고서 전송⁽⁸⁾, 휴대 및 전송 가능한 의무기록 포맷⁽⁹⁾, 임상 가이드라인의 표준화⁽¹⁰⁾ 등에서 인터넷의 새로운 문서형식 및 통신 규약인 XML의 활발한 활용이 시도되고 있다. 때마침, 대규모 의료 정보 시스템들도 웹기반 클라이언트-서버 환경의 장점을 급속히 흡수하며 이전되고 있다.

많은 프로그램들이 아직은 자료의 구조화를 가능하게 한다는 XML의 장점을 주로 활용하고 있지만 "확장 가능한 extensible" 이란 그 이름에서 암시되는 바와 같이 XML의 더욱 중요한 장점은 자료의 추상화를 가능하게 한다는 점이다. 본 논문의 범용 데이터 폼 편집기는 XML의 이러한 자료 추상화 기능을 응용한 것으로 XML의 무한한 응용 범위를 살펴볼 수 있다. 그러나 본 프로그램의 개발에 시험 적용된 XML 파서와 XSL 프로세서는 상용 프로그램의 경우도 아직까지 언어의 표준 사양을 충분히 지원하는

것이 출시되어 있지 않고, XSL 편집기의 경우는 실용적으로 사용하기에는 어려운 부족한 것들이었다. 프로그램간 상호 호환성이 충분치 않아 많은 부분은 직접 텍스트 편집기로 작성하였다. XML 파일 서버의 경우도 아직 그 기능성을 충분히 보장할 수 있는 성능 좋은 제품이 출시되지 않아 본 프로그램에서는 XML DTD만을 정의하고 그의 실제 구축은 관계형 데이터베이스에서 구현하였다. 그러나 빠른 속도로 도입되고 있는 XML의 발달은 매우 빠르며 빠른 시간 안에 실용적으로 사용하기에 무리가 없는 각각의 호환 모듈들이 개발될 것으로 예견된다. 이러한 분산 개발 환경은 프로그램의 컴포넌트화를 촉진하여 효율적인 정보 시스템 구축 및 관리를 가능하게 할 것으로 기대된다.

객체의 사후 정의는 특정 프로그램에 알맞은 적절한 객체 데이터 유형 및 프로시저 기술 언어를 개발함으로써 가능하다. 그러나 그러한 경우는 표준화가 거의 불가능할 뿐 아니라 새로이 작성된 언어를 습득하기 위한 학습 곡선이 매우 가파르다. 그러나 XML/XSL 기술의 결합은 웹기반 프로그래밍에 필요한 거의 모든 객체 구성 기술 요소를 제공할 뿐 아니라 이미 전 세계적으로 표준화된 HTML과 같이 실질적인 표준으로 자리잡아가고 있음으로서 특정 기술 언어를 배우기 위한 학습곡선이 필요치 않다. 또한 이러한 범세계적인 표준화는 XML 파서, XSL 프로세서, XSL 편집기와 같은 각각의 개발 모듈들의 분산 개발을 가능하게 함으로써 프로그램의 개발 주기를 유의하게 절약해 줄 것으로 기대된다.

참고문헌

1. World Wide Web Consortium. The World Wide Web consortium issues XML 1.0 as a W3C recommendation: key industry players, experts collaborate to develop interoperable data format for the web. 1998 Available from: URL:
<http://www.w3.org/Press/1998/XML10-REC>.
2. Sittig DF, Stead WW. Computer-based Physician Order Entry: The State of the art. J Am Med Inform Assoc 1994;1(2):108-123.
3. Kim JH, Ferziger R, Sands DZ. Development of a Web-based Patient Interview Program. Proc AMIA Annu Fall Symp 1999::1099.
4. Franklin MJ, Sittig DR, Schmitz JL. Modifiable templates facilitate customization of physician order entry. Proc AMIA Annu Fall Symp 1998::315-319.
5. Johnson SB. Generic data modeling for clinical repositories. J Am Med Inform Assoc 1996;3(5):328-339.
6. HL7 SGML/XML Special Interest Group web site. Available from: URL:
<http://www.mcis.duke.edu/standards/HL7/sigs/sgml/>.
7. Dolin RH, Rishel W, Biron PV, Spinosa J, Mattison JE. SGML and XML as interchange formats for HL7 messages. Proc AMIA Annu Fall Symp 1998::720-724.
8. Kahn CE, Cruz NB. Extensible Markup Language (XML) in health care: Integration of structured reporting and decision support. Proc AMIA Annu Fall Symp 1998::725-729.
9. Chueh HC, Raila WF, Berkowicz DA, Barnett OG. An XML portable chart format. Proc AMIA Fall Symp 1998::730-734.
10. Dubey AK, Chueh HC. Using the Extensible Markup Language (XML) in automated clinical practice guidelines. Proc AMIA Annu Fall Symp 1998::720-724.

Development of a Generic Data Form Authoring Program Based on XML/XSL Technology

- XML/XSL-based Generic Form Editor -

Ju Han Kim, M.D., Ph.D.

Division of Medical Informatics, Harvard Medical School

Abstract

The World Wide Web provides the means for the integration of clinical computer systems. Extensible Markup Language (XML) and Extensible Stylesheet Language (XSL) are newly developed and standardized Internet protocols. Development of custom data entry forms requires significant programming resources in clinical information system development and management. Visual design tools and a modifiable, template-driven approach may facilitate this process. However, these approaches generally require the predefinition of data form element types. This paper describes an XML/XSL-based technology enabling post hoc definition of elementary and composite data entry form element types (**Journal of Korean Society of Medical Informatics 6-2, 65~72, 2000**)

Keywords : XML/XSL, Information System, Data Entry, Internet