

웹 기반 XML 개발도구 (An XML Development Tool based on the Web)

유 형 상, 오 세 만
동국대학교 컴퓨터공학과
{hsyou, smoh}@dgu.ac.kr

요 약

웹 기반 클라이언트-서버 환경을 이용한 XML 개발환경은 기존의 서버 중심적인 개발환경과는 달리 세 가지로 요약된 장점을 갖는다. 첫째는 개발자의 공간적 제약조건을 배제할 수 있다는 점과 둘째는 프로젝트의 일관성을 유지할 수 있다는 점이며, 마지막으로 웹 브라우저를 개발 및 실행환경으로 이용할 수 있다는 점이다. 따라서 본 논문에서는 웹 기반 프로그램 개발환경의 효율성과 가능성을 보여주기 위하여 웹 기반 클라이언트-서버 개념의 XML 개발도구를 설계 및 구현하고자 한다.

1. 서론

프로그램의 개발환경이란 프로그램 에디팅 환경, 프로그램 컴파일환경, 프로그램 실행환경, 프로그램 관리환경으로 구성된 모든 환경을 총칭한다. 이러한 프로그램 개발환경이 기존에는 서버 기반으로 이루어졌거나 클라이언트일지라도 서버와 동일한 기능을 가져야만 개발할 수 있다는 제약을 가졌다. 이것

은 개발환경 자체를 서버 기반으로 설계한데서 야기되는 단점이며, 이에 인터넷 기반 웹 환경을 활용하여 클라이언트-서버 개념의 개발환경으로 확장함으로써 다음과 같은 유용성을 얻을 수 있다.

첫째, 웹 기반 클라이언트-서버 개념의 개발환경은 개발자의 공간적 제약조건을 배제할 수 있다.

둘째, 서버에서 통합 관리가 가능하므로 프로젝트의 일관성을 유지시키며 개발을 진

행할 수 있다.

마지막으로, 웹 브라우저를 개발 및 실행 환경으로 이용할 수 있다는 점이다.

따라서 본 논문에서는 인터넷 기반 웹 환경을 통하여 클라이언트-서버 개념의 XML 개발 도구를 설계하고 구현하고자 한다. 본 논문의 구성은 2장에서 관련연구, 3장에서 웹 기반 하에서 XML 개발도구 설계, 4장에서는 웹 기반 하에서 XML 개발도구 구현 내용에 대해 설명하고, 5장에서는 결론 및 향후 연구방향을 제시한다.

2. 관련연구

새로운 개발환경을 설계한다는 것은 기존 개발환경의 언어적 이해와 도구적 이해가 필수적이며 이러한 바탕에서 좀 더 개선되고 유용한 개발환경을 만들어 낼 수 있다.

2.1 XML

XML은 eXtensible Markup Language를 말하며 여기서의 extensible은 HTML과 같이 고정된 format이 아니라는 것을 의미한다. 또한 markup language를 고안할 수 있는 meta language를 나타내기도 한다.

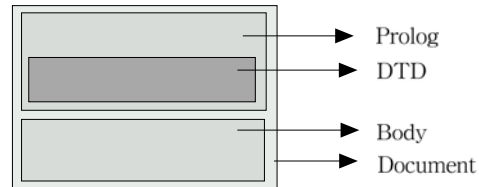
XML은 SGML의 복잡한 기능을 단순화하여 쉽게 markup language를 정의할 수 있는 도구이며 일반적인 XML 파서가 개발되어 정의된 언어의 처리가 매우 융통성이 있다는 특징을 가진다.

2.2 XML 문서의 구성

XML은 논리적 구조와 물리적 구조를 가진다. 물리적 구조는 엔티티라고 불리는 단

위로 구성되어지고 논리적 단위는 선언, 엘리먼트들, 주석들, 문자 참조들, 처리 명령어들로 구성되어진다.

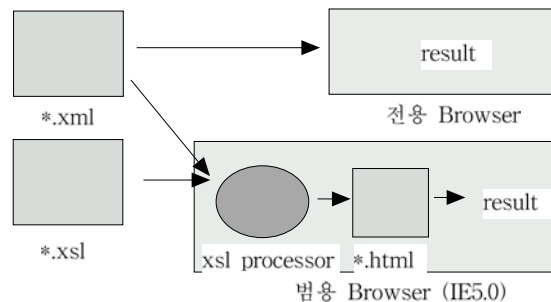
XML문서는 일반적으로 [그림1]에서와 같이 Prolog와 DTD(외부에서 정의되어 포함될 수 있다), 그리고 문서의 몸통(Body)으로 구성되어져 있다.



[그림1] XML 문서의 구성

2.3 XML의 문서처리 모델

문서처리 모델은 [그림2]에서와 같이 전용 브라우저를 사용할 경우 *.xml을 통한 결과를 볼 수 있다.



[그림2] *.xml 문서 처리 모델

하지만 범용 브라우저에서 실행하는 경우에는 *.xml문서만으로는 볼 수가 없다. 따라서 이 경우에는 XML의 stylesheet language인 XSL을 이용하게 되는데 *.xml문서에 *.xsl

```
<?xml:stylesheet type="text/xsl" href="*.xsl"?>
```

을 참조하는 위와 같은 정보를 포함하게되면 내부적으로 *.html문서로 변환 후 결과를 볼 수 있도록 되어있다.

2.2 XML의 개발환경

XML의 개발환경은 XML 파서를 기준으로 에디터와 브라우저 그리고 이 도구들을 통합한 환경이 만들어져 있다. 제공되고 있는 XML 개발도구 및 환경을 정리하면 아래와 같다.

* Parser

- XP : James Clark이 개발한 XML Parser이다.
- XML for Java : IBM에서 Java로 구현한 Validating XML Parser이다.
- MSXML : MS에서 개발한 XML Parser로서 Java로 구현되어져 있다.

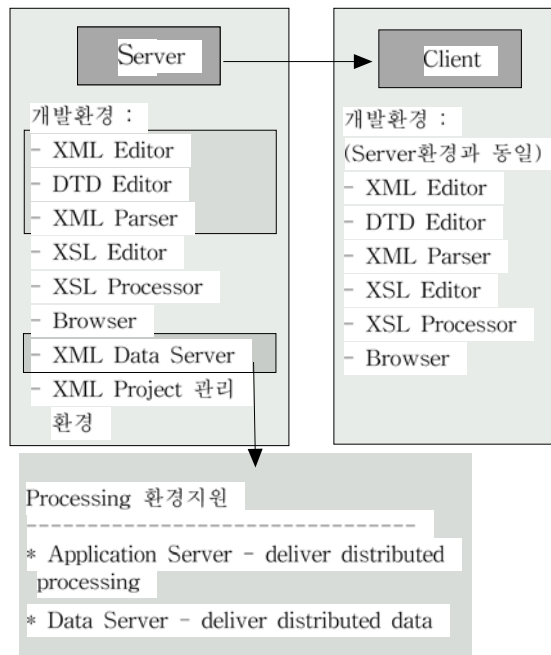
* Editor & Browser

- XML Notepad : MS에서 개발한 XML Editor이다.
- XED : Document Instance Editor이다.
- Xpose : Validating Editor로써 주요기능으로는 XML source viewer, graphical DTD display, customizable display기능을 갖는다.
- Java XML Browser
- JUMBO

* 통합 개발 환경

- eXcelon : XML의 통합 개발 환경을 제공하는 eXcelon서버는 eXcelon Data Server, eXcelonToolbox, eXcelon Xconnects로 구성되어진 개발도구로서 XML 개발 환경 및 실행 환경을 통합적으로 관리한다.

기존의 XML 개발 환경에서 통합적인 개발 환경을 제공하는 경우는 [그림3]과 같은 개발 환경 모델을 가진다. 이 경우 클라이언트



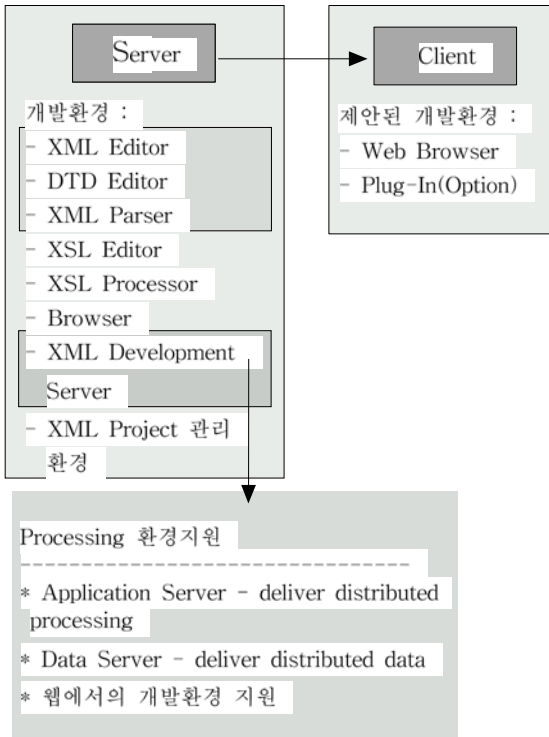
[그림3] XML 통합 개발 환경

의 개발 환경또한 서버와 크게 다르지 않다. 다른 점이라면 서버환경에 XML Data Server와 XML 프로젝트 관리 환경을 두어 프로젝트의 실행 환경을 통합적으로 관리한다는 점이다.

3. 웹 기반 XML의 개발도구의 설계

웹 기반 XML의 개발도구의 설계는 기존의 개발환경과는 다른 웹 기반의 개발환경으로 발전시키기 위해서는 기존의 서버환경에서 통합적으로 운영되어지던 프로세싱을 클라이언트-서버 환경에 적합하도록 프로세싱을 구분하여 환경을 구성하는 것이 매우 중

요하다. [그림4]에서는 이런 개선된 XML의 개발환경을 보여준다.



[그림4] 제안된 웹 기반 XML 개발 환경

기존의 개발환경과의 차이점은 XML Development Server를 두어 실행환경의 지원뿐만 아니라 웹 기반 개발환경을 지원한다는 데 커다란 차이를 보인다. 또한 이러한 지원은 클라이언트 개발 환경에 차이를 가져오게 한다.

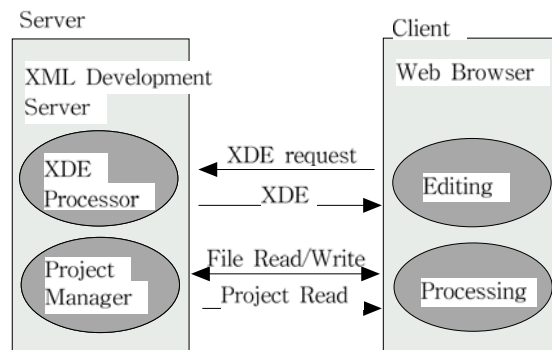
3.1 개발 환경의 세부구성 및 기능

개발 환경의 세부구성은 크게 웹 기반 XML 개발환경과 XML 프로젝트 관리환경 2부분으로 나누어 설계하였다.

XML 개발환경을 세분화하면 문서 에디팅 환경, 프로세싱환경으로 나눌 수 있다.

첫째로 문서 에디팅 환경은 *.xml 문서를 생성할 수 있는 문서 에디팅 환경을 제공하게 된다. 또한 분리되어진 DTD문서의 에디팅 환경과 *.xsl 문서 에디팅 환경을 제공함으로써 문서를 좀더 편리하게 작성할 수 있도록 하는 부분이다.

둘째로, 프로세싱환경은 결과가 *.html로 실행하는 경우 *.xml과 *.xsl을 읽어서 실행하는 환경을 제공하게 되어지는 부분이며 다른 프로그램에 Import되어지는 경우를 고려하여 설계되어야 한다. 또한 결과가 *.html로 나오지 않는 경우에 *.xml을 읽어 들여서 프로세싱을 별도로 지원하여야 한다. 이러한 경우 웹 브라우저에서 제공하지 않으므로 별도로 환경을 구축하여야 한다. 이를 위해 실행환경에 맞는 Plug-In을 실행할 수 있는 별도의 환경설정을 할 수 있도록 설계되어야 한다.



* XDE : XML Development Environment

[그림5] XML 개발환경의 프로세싱과정 및 모듈

XML 프로젝트 관리환경은 서버에서 다수의 사용자가 접속한 경우 이에 맞게 파일을 관리하고 프로젝트를 중복되지 않게 하여 일관성을 유지하도록 관리하기 위해 제공하는

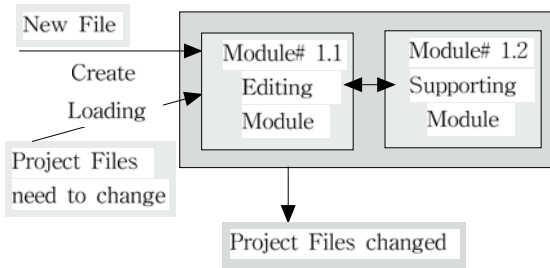
환경이다.

[그림5]에서는 위에서 설명한 개발환경의 프로세싱의 흐름 및 모듈을 보여주고 있다.

3.2 구성별 세부항목의 설계

앞에서 구성한 웹 기반 XML 개발 환경을 모듈별로 구현에 필요한 설계를 하고자한다. 설계 시 가장 주의해야할 사항은 클라이언트와 서버간의 적절한 기능의 배분이다. 모든 기능이 서버에 치중될 경우 클라이언트-서버 환경의 효율성을 충분히 살리지 못하게 되며 기능이 클라이언트에 편중될 경우 기존의 환경과 크게 다르지 않은 상태에서 작업이 이루어지게 됨으로 기능의 배분은 매우 중요하다. 이에 맞게 웹 기반 XML 개발도구를 크게 세 모듈로 분류를 하고 모듈별로 기능을 정리해 보면 아래와 같이 나눌 수 있다.

첫째로 XML문서의 에디팅 모듈은 기존의 개발 환경에서는 서버 또는 클라이언트 독립적으로 이루어지는 프로그래밍 환경인데 반하여 제안된 프로그램 개발 환경은 클라이언트가 웹 브라우저를 통하여 서버에서 다운로드 받는 형태로 이루어진다. 다운로드된 에디팅 모듈은 웹 브라우저를 통하여 에디팅 환경을 지원하게 된다. [그림6]에서 설계되어진 모듈을 볼 수 있다.



[그림6] 에디팅 모듈(외부모듈)

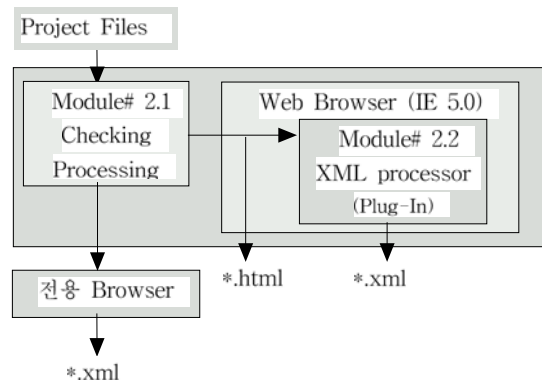
[그림6]의 에디팅 모듈의 세부 모듈별 설명 및 기능을 살펴보면 아래와 같다.

- * Module#1.1 Editing Module : XML문서를 만들고 에디팅하는 모듈로서 *.xml, *.xsl, *.dtd문서를 만들고 에디팅할 수 있는 기능을 제공한다.
- * Module#1.2 Supporting Module : XML 문서의 에디팅 환경을 이용하기 좀 더 편리하도록 하기 위해 제공되는 모듈이다.

둘째, 프로세싱 모듈은 프로세싱 형태에 따라 아래와 같이 세 가지로 구분할 수 있다.

- 1) 범용 브라우저를 사용하여 프로세싱이 이루어지는 경우
- 2) 범용 브라우저에 내장된 플러그인을 사용하여 프로세싱이 이루어지는 경우
- 3) 전용 브라우저를 사용하여 프로세싱이 이루어지는 경우

프로세싱 모듈은 위에서 구분한 프로세싱 방법에 따라 [그림7]에서와 같은 처리 세부모듈을 가지게 된다.



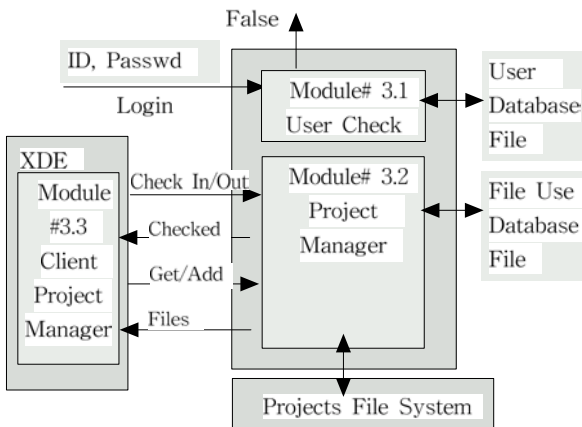
[그림7] 프로세싱 모듈 (내부모듈)

[그림7]의 프로세싱 모듈의 세부 모듈별 설

명 및 기능을 살펴보면 아래와 같다.

- * Module#2.1 Checking Processing : 선택되어진 프로세싱 메뉴에 따라서 프로세싱 방법을 결정하는 모듈이다
- * Module#2.2 XML Processor : 개발도구에 의해 구현되는 모듈이기보다는 XML Project 개발자에 의해 구현되어지는 모듈로써 개발하려고 하는 XML의 스타일에 맞게 프로세싱 형태를 결정하여 실행 결과를 보여주는 모듈이다.

마지막으로 프로젝트 관리 모듈은 프로젝트의 일관성을 유지시켜주고 권한을 부여받은 개발자에 의해서만 개발이 이루어질 수 있도록 관리를 해주는 모듈이다. 모듈은 [그림8]에서와 같은 구조를 가지게 된다.



[그림8] 프로젝트 관리모듈(내부모듈)

[그림8]의 프로젝트 관리 모듈의 세부 모듈별 설명 및 기능을 살펴보면 아래와 같다.

- * Module#3.1 User Check : 웹에서 서버에 접속하여 개발도구를 사용하고자 할 때 개발자를 검증하는 모듈이다.
- * Module#3.2 Project Manager : XDT를 통해 접속한 클라이언트 모듈인 Client

Project Manager와 상호작용을 통하여 프로젝트의 일관성 및 보안을 담당한다.

- * Module#3.3 Client Project Manager : 서버의 Project Manager와 상호작용으로 프로젝트의 일관성 및 보안을 유지시키는 작업을 담당하는 모듈이다.

4. 웹 기반 XML개발 도구의 구현

웹 기반 XML의 개발도구의 설계에서 설계되어진 모듈을 기반으로 개발도구를 모듈별로 구현한다.

4.1 XDE Interface 환경 및 기능

웹 기반 XML의 개발도구의 구현은 개발 환경과 운용환경에 의하여 개발도구의 차이를 가져올 수 있다. 그러나 XML 개발도구의 목적은 차이를 가지지 않는다. 이번 논문에서는 [표1]에서와 같은 개발 환경과 테스트 운용 환경에서 구현이 이루어졌다.

개발 환경	
Programming Language	JDK1.2.2, Delphi4.0
Extended Package	Swing, Servlet, RMI ActiveX
테스트 운용 환경	
Operating System	WindowNT
Web Server	Java Web Server 2.0

[표1] 개발 환경과 테스트 운용환경

4.2 XML 개발도구의 인터페이스 및 기능

XML 개발도구의 인터페이스는 개발자로 하여금 좀더 편리한 방법으로 개발에 참여할

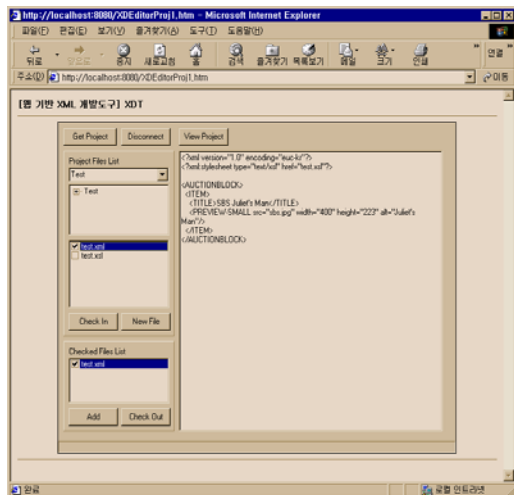
수 있도록 모든 개발자에게 익숙한 인터페이스로 설계하였다.

- * XML 개발환경 로그인 화면 구성 :
XML 개발도구로 접속하기 위해 인증을 받는 화면이다.



[그림9] 사용자 로그인 화면

- * XDT(XML Development Tool) 개발자 화면 : XML의 개발자의 에디팅 모듈 및 프로젝트 파일을 웹 상에서 작업할 수 있도록 지원하는 인터페이스 화면을 [그림10]에서 보여주는 것이고 [표2]에서는 XDT의 기능을 나타낸다.

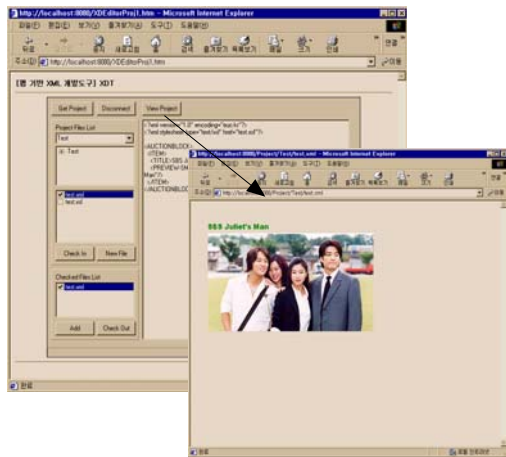


[그림10] XDT 개발 화면

컴포넌트	이름	기능
Button	Get Project	Project List에서 선택된 프로젝트의 파일을 Project File List로 가져온다.
	Check In	Project File List에서 선택된 파일을 Checked File List로 가져오면서 쓰기 권한을 갖는다.
	New File	새로운 파일을 만든 후 Project List에 등록을 한다.
	Add	Check In한 File을 작업 후에 Project 에 Update한다.
	Check Out	작업이 끝난 파일의 쓰기 권한을 반납하고 Checked File List에서는 삭제된다.
	Disconnect	Get Project에 의해서 연결된 Project Manager와의 연결을 끊는다.
Selection List	View Table	XML source를 Table형태로 보여준다.
	Projects List	서버에 접속함과 동시에 Project들의 이름이 Selection List에 나열된다.
File list	Project Files List	선택된 Project의 파일들의 이름이 나열된다.
	Checked Files List	Check In에 의해 얻어온 파일들의 이름이 나열된다.
Edit	Source Edit	XML문서를 에디팅하는 환경이다.

[표2] XDT 기능 테이블

- * XDT에서의 XML문서 실행화면 : 작업된 XML문서를 실행하여 보여지는 화면이다. View Project메뉴를 사용하여 아래와 같은 결과를 볼 수 있다.



[그림11] 실행화면

5. 결론 및 향후 연구방향

본 논문에서 설계 및 구현한 웹 기반 XML 개발 도구의 구현은 기존의 개발 도구와 비교하여 개발자로 하여금 공간적 제약을 최소화하고 시스템의 제약 또한 많은 부분 감소시킬 수 있을 것이다. 뿐만 아니라 자연적인 프로젝트 관리효과를 가져올 수 있으므로 프로젝트 관리에 따른 추가 비용을 줄일 수 있는 효과를 가져올 것이다. 또한 XML이라는 언어적 특성을 충분히 살려서 설계되었으므로 기존의 웹 브라우저를 이용한다는 점에서 매우 효율적이다.

이에 향후 연구 방향은 본 논문에서 설계된 웹 기반 XML 개발도구는 설계 시에 XML의 개발환경에서 고려되어야 될 많은 부분 -편의성, 효율성 등- 을 고려하여 설계되어졌으나 구현에서는 웹 기반 XML 개발도구의 장점만을 간략하게 고려하여 구현되어 졌다. 따라서 구현된 개발도구의 장단점

을 보완, 수정하는 것이 필요하며 설계 시 개발의 편의성을 위해 첨가된 기능들의 부분적인 모듈구현이 이루어져야 할 것이다. 또한 본 논문에서 설계 시에 포함되어지지 않은 분산화와 통합화가 향후 설계 시 고려되어야 할 부분이다.

개발 환경의 분산화와 통합화는 현재 설계되고 구현된 개발 도구에서는 XML의 개발환경만을 제공하는 반면 기존의 XML 개발현황은 XML을 독립적인 문서화에 의해 개발도 이루어지지만 Java와 같은 Server-Side Language들과 결합되어 개발되어지는 것이 일반적이다. 이에 XML과 결합되어 작업이 이루어지는 다른 프로그래밍 언어와의 통합적인 작업을 할 수 있는 개발도구의 설계 및 구현이 이루어져야 하며, 이러한 통합이 이루어 질 경우 서버의 프로세싱 부담은 매우 클 것이다. 이에 좀더 효율적인 서버의 분산환경이 필요하고 또한 클라이언트의 프로세싱 부담 또한 필수적이라고 할 수 있기 때문에 향후 연구과제로 삼아서 지속적인 연구가 필요하다.

참고문헌

- [1] Alex Ceponkus, Faraz Hoodbhoy, Applied XML A Toolkit for Programmers, Wiley, 1999.
- [2] Dan Woods, Larne Pekowsky, Tom Snee, The Developer's Guide to the Java Web Server, Addison Wesley, 1999.
- [3] Dustin R. Callaway, Inside Servlets, Addison Wesley, 1999.
- [4] eXcelon <http://www.objectdesign.com>, 1999.
- [5] Extensible Markup Language(XML) 1.0, <http://www.w3.org/TR/REC-xml>, 1998.

- [6] Hiroshi Maruyama, Kent Tamura, Naohiko Uramoto, XML Developing Web Applications, Addison Wesley, 1999.
- [7] IBM Web Page
<http://www.alphaworks.ibm.com/tech/xml4j>, 2000.
- [8] Java Servlet and Java Server Page Technology, Sun Microsystems, 2000.
- [9] Merlin Hughes, Java Network Programming, Manning, 1999.
- [10] Professional Java Server Programming, Wrox Press, 1999.
- [11] S. Holzner, XML complete, McGraw-Hill, 1999.
- [12] The SGML/XML Web Page,
<http://www.oasis-open.org/cover/>, 1999.
- [13] The XML FAQ, <http://xml.t2000.co.kr/faq/>, 1999.
- [14] William.J.Pardi, XML in Action Web Technology, Microsoft Press, 1999.
- [15] 백운기, 한상홍, 박준후, Delphi 5 Contact, 대림출판사, 2000.
- [16] 유형상, 오세만, “웹 기반 XML 개발도구의 설계 및 구현”, 2000년 추계학술 발표회, 정보과학회논문



유형상

'92.03-'99.02 동국대학교 컴퓨터공학과 학사

'99.03-현재 동국대학교 컴퓨터공학과 프로그래밍언어 연구실 석사과정

관심분야 : 프로그래밍 언어, JVM, XML.



오세만

'93.03-'99.02 동국대학교 컴퓨터 공학과 대학원 학과장

'85.03-현재 동국대학교 컴퓨터 공학과 교수

관심분야 : 컴파일러, 프로그래밍 언어, XML, 무선인터넷 언어.