

□특집□

차세대 웹 서비스 기술 개발 동향

김 태 석[†] 박 철 제^{††} 임 환 섭^{†††}

◆ 목 차 ◆

- | | |
|---------------|---------------|
| 1. 서 론 | 3. 웹서비스 기술 개발 |
| 2. 개발 배경 및 목적 | 4. 결 론 |

1. 서 론

21세기에는 정보 통신 관련기술의 발전으로 초고속 정보 통신망에 의한 다양한 응용 서비스가 진행될 수 있으며, 차세대 인터넷의 기술 개발 추세는 초고속 통신망의 초고속 전송을 위한 망구조 자체의 개발과, 인터넷 서비스를 기반으로 한 초고속 응용 프로그램 개발 및 고속 분산 컴퓨팅 환경 구축 등을 들 수 있다[1]. 웹이 개발된 지난 10년간 웹 기술 자체는 팔목할 만큼 성장하였으며, 최근 인터넷의 웹 기술을 바탕으로 기하급수적인 이용 증가율을 보여 왔던 WWW은 현재 가장 광범위한 사용자 계층을 확보하고 있고, 기존의 다양한 기술들이 웹과 접목이 시도되고 있는 상황이다. 기존의 웹은 텍스트 기반의 구조를 통한 비동기 적인 형태의 서비스 유형으로 문서검색과 단순한 표현 방식을 사용하였으나 컴퓨팅 환경과 웹 기술의 발전으로 인터넷상의 데이터들이 복잡해지고 구조화되어 가면서 다양한 멀티미디어 정보를 동기화 하여 표현하고자 하는 요구가 증대되고 있다. 이러한 요구는 웹의 새로운 구조 및 표현방식을 요구하게 되었다. 차세대 웹

은 기존의 구조형식을 토대로 새로운 표현 기술을 이용하여 확장되고, 이를 통한 웹서비스 기술 개발로 이어지고 있다.

본 논문에서는 차세대 웹서비스 환경을 제공할 수 있는 웹 기술 개발에 대한 동향을 분석하고, 각 분야별 기술 개발 내용에 대해서 기술한다.

2. 개발 배경 및 목적

웹과 관련된 기술 개발은 W3C(WWW Consortium)를 중심으로 표준화가 이루어지고 있으며, 웹서비스에 대한 기술의 개발은 웹 인터페이스 환경을 통하여 이루어진다.

본 장에서는 HTML, Style Sheet, DOM, MetaData, RDF에 관한 기술 개발 동향을 고찰한다.

■ HTML(HyperText Markup Language)

HTML은 WWW의 표준언어로 최근까지 여러 번의 변화를 겪어왔으며 현재는 1997년 12월에 Recommendation으로 채택된 HTML 4.0 버전이 있다[2]. 기존의 HTML 4.0 규약을 XML의 응용으로 변환하며 여러 가지 모듈로 나뉜 모듈화 된 태그들을 사용하도록 해서 HTML의 응용범위를 휴대폰, TV, Audio, PDA 등에까지 확대되고 있다.

■ 스타일 시트

스타일 시트란 문서가 스크린, 프린터, 또는 기타 출력기로 표현되어 질 때 어떻게 그 문서를

[†] 정회원 : 동의대학교 컴퓨터공학과 교수

^{††} 정회원 : 동의대학교 경영정보학과 교수

^{†††} 정회원 : 금아시스템(주) 개발실장

표현할 것인가에 대한 정의이다. 스타일 시트에 대한 연구는 1994년부터 W3C에서 시작되었으며, WWW 상의 문서 스타일에 대한 표준을 생성하는 것을 그 목적으로 하고 있다[3]. 스타일 시트는 크게 CSS와 XSL로 나뉘어져 있다.

◆ CSS(Cascading Style Sheets)

CSS는 웹 문서에 스타일(폰트, 컬러, 간격 등)을 추가할 수 있는 간단한 메커니즘이다[5]. CSS는 HTML 파일로 작성된 웹 문서에 대한 일괄적인 스타일을 지정하는 표준을 작성하게 된다. 현재 CSS에서는 Cascading Style Sheets에 대한 level-1과 level-2를 작성하였다. 표-1에는 현재 스타일 시트의 문서 상황을 나타내었다[3,4,5].

(표 1) CSS 문서상황

발표 일자	문서 제목
1996년 12월 17일	Cascading Style Sheets (CSS) Level 1 Specification
1998년 5월 12일	Cascading Style Sheets (CSS2), level 2 Specification

◆ XSL

XSL에서는 차세대 인터넷 표준 문서인 XML의 스타일에 관련된 표준을 제정한다. 현재 XSL에서는 Extensible Style sheet Language (XSL), XSL Requirements Summary 을 Working Draft로 제출한 상태이며, 이를 보완하여 Recommendations로 버전업 할 예정이다[3,4]. (표 2)에 현재 XSL의 문서 상황을 나타내었다.

(표 2) XSL 문서상태

발표 일자	문서 제목
1998년 5월	XSL Requirements Summary
1998년 12월	Extensible Style sheet Language (XSL)

■ DOM (Document Object Model)

DOM은 내부적으로 DOM WG와 DOM IG의 두 워킹 그룹으로 구분되어 있다. DOM WG는 DOM 표준안 개발을 하고 있으며, DOM IG는 DOM에 대한 포럼 역할만 하고 있다. DOM은 HTML 문서 또는 XML 문서와 같이 인터넷 문서에 대하여 문서의 내용 및 구조를 객체로 표현하고 그 객체를 핸들링 할 수 있는 플랫폼에 독립적이고 언어 중립적인 인터페이스이다. 따라서 DOM 워킹 그룹의 목적은 이러한 인터페이스의 개발에 있다[7].

DOM 워킹 그룹의 목표는 DOM을 지원하는 인터페이스에 대한 표준안을 만드는 것으로 표준안은 레벨 1, 레벨 2, 레벨 3으로 나뉜다. 표-3에서 볼 수 있듯이 현재 표준안은 레벨 1까지 진행되고 있으며, 레벨 2는 1999년을 목표로 하고 있다. 현재까지 작성된 문서는 Document Object Model (DOM) Level 1 Specification, DOM FAQ, Document Object Model Requirements이다.

(표 3) DOM 작업문서

발표 일자	내 용
1998년 10월	Document Object Model (DOM) Level 1, Recommendation
1999년 3월	Document Object Model (DOM) Level 2

■ 메타 데이터

일반적으로 메타 데이터란 “데이터에 관한 데이터”로서 콘텐츠 이외의 콘텐츠와 관련된 정보(예를 들어, 변경일자, 수정일자, 생성자, 버전 등)를 일컫는 말로서, 웹에서의 메타 데이터는 웹에 대한 정보를 의미한다. 메타 데이터 그룹은 웹 정보에 관한 정보를 위한 프레임워크를 설정하는 것을 목적으로 그 활동을 하고 있으며 크게 PICS 분야와 RDF 분야가 있다[8,9].

◆ RDF

RDF는 인터넷상의 정보 자원에 대한 정보를 기술하는 표준을 이루는데 목적이 있으며, 구문과 스키마의 RDF와 관련된 프레임워크 작업을 진행하고 있다. RDF는 그 디자인에 있어 다음과 같은 목표를 만족하도록 하고 있다[9]. 먼저 각각의 응용들에 있어 가장 알맞은 메타 데이터 속성들을 정의할 수 있도록 하고, 프로그램들 사이나 웹을 통해 메타 데이터를 교환할 수 있는 단일하고 상호 호환되는 수단을 제공할 수 있도록 하며, 속성들에 대해 사람이 읽을 수 있고, 기계가 이해 가능한 정의도구를 제공할 수 있어야 한다는 것이다.

3. 웹서비스 기술 개발

3.1 HTML(HyperText Markup Language)

HTML은 1997년 12월에 Recommendation으로 채택된 HTML 4.0 버전을 기초로 1998년 4월 California, San Jose에서 열린 Future of HTML 워크샵에서 발의된 개정판 HTML 4.0에 근거하고 있다. 이 워크샵에서 다음과 같은 결론을 얻게 되었다. 먼저 HTML 4.0이상의 개발은 힘들 것으로 여겨짐으로 인해 4.0을 XML응용으로 변환하여야 한다. HTML NG는 XML 태그들에 기반을 둘 것이다. 워크샵에서는 이와 함께 데이터베이스와 워크플로우 응용제품과의 관련성 향상에 대한 요청이 있었으며, 마찬가지로 이동형/소형기기에 대한 지원 요청도 함께 있었다. 이런 요청들을 만족하기 위해서는 모듈화 된 HTML이 필요하며 이를 충족시키기 위하여 XHTML(Extensible HyperText Markup Language) 1.0이 제안되었다. 현재의 HTML 4.0은 웹의 매우 빠른 확장에 따라 지난 6년여 동안 요구된 다양한 태그들이 존재하고 있다. 이러한 HTML의 모듈화는 HTML을 더 관리하기 쉽게 만들어 줄 수 있다. 또한 수학이나 음악, 벡터 그래픽, 멀티미디어 표현 등과 같은 다

른 XML 태그들과의 통합적인 사용을 가능하게 한다. HTML 자체가 XML의 태그 집합으로 정의될 것이며, HTML은 핵심집합(core set)을 이루는 태그들과 머리말, 문장, 관계 데이터와 폼, scalable graphics, 멀티미디어 등과 같은 다른 태그들로 구성될 것이다. 따라서 이러한 여러 가지 태그들은 필요한 경우에 조합하여 사용할 수 있을 것이다.

XML은 문서의 구조적 표현제작과 markup이 표현하는 객체집합으로 문서를 매핑 하는데 있어 분명한 구분을 제공한다. 먼저 태그들의 구조를 만들고 객체 집합으로 결과를 만들어 내는데 스타일시트나 스크립트, 프로그램과 같은 형태의 다른 정보를 사용하게 된다. 이와 같이 하여 HTML과 다른 태그들간의 조합을 손쉽게 하며, 여러 가지 기기들에서의 필요에 따라 HTML을 조정할 수 있게 된다.

미래에는 웹 상의 정보들을 컴퓨터모니터를 통해서보다 다른 기기를 통해 보기를 원할 것이다. 이와 같은 미래의 웹은 데스크탑 컴퓨터뿐만 아니라, 휴대폰, TV, 오디오 기기, PDA, embedded 기기 등과 같은 곳에서도 사용가능 할 것으로 전망된다. 즉, 사용자 인터페이스가 필요한 어디든 웹이 사용될 것이다. 따라서 HTML의 추후 디자인은 이런 경향을 고려하여야 한다.

3.2 스타일 시트

일반적으로 스타일 시트란 문서에 대한 스타일(문서의 폰트, 폰트 크기, 마진, 여백 등)을 지정하는 것이다. 스타일 시트를 사용하는 가장 큰 이유는 스타일 시트를 정의함으로써 그 스타일 시트에 적용되는 모든 문서들이 동일한 형태로 유지될 수 있으며, 스타일 시트만 수정함으로써 문서전체의 표현 형태를 바꿀 수 있기 때문이다. 따라서 대용량 문서 제작이나, 공동 작업을 이용하여 문서 작성을 하는 경우에 스타일 시트는 필수적이 된다. 웹에서의 스타일은 크게 HTML 문

서에 대한 스타일과 XML 문서에 대한 스타일로 구분할 수 있으며 각각 CSS, XSL이라 부른다.

3.2.1 CSS(Cascading Style Sheets)

인터넷 문서의 가장 일반적인 형식은 HTML이지만, 문서를 자유롭게 편집하는 데에 많은 제약이 따른다. 이를 보완하기 위해 만들어진 것이 스타일 쉬트다. 스타일 쉬트의 문법과 형식은 W3C의 Cascading Style Sheet(CSS)로 표준화되어 있다.

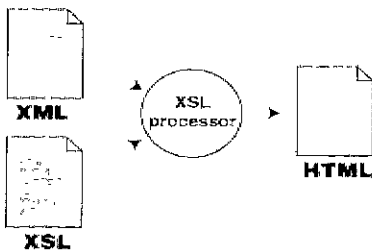
CSS의 특징은 다음과 같다.

- HTML 요소의 기능을 확장한다.
- 통일된 문서양식을 디자인 할 수 있다.
- 문서의 형식을 다양하게 구성할 수 있다.
- 사용자의 환경 User Agent에서 독립된 문서를 제작한다.

CSS1은 단순히 브라우저 상에서 보이게 되는 HTML의 각 엘리먼트(태그)에 대한 스타일을 지정할 수 있는 것으로 글꼴, 색상과 바탕, 문장, 윤곽, 문단 형식 등이 지정되어 있다. 또한 CSS2는 기존의 CSS1을 기반으로 하여 단점을 보완하고, 새로운 기능들이 추가되었다.

3.2.2 XSL

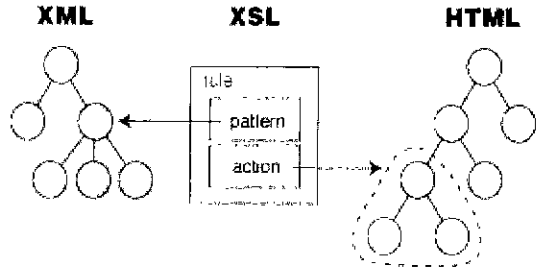
XSL은 XML 문서에 대한 스타일을 지정할 수 있는 메커니즘으로써 XML 인스턴스와 DTD, 그리고 스타일 지정 파일을 통하여 사용자가 원하는 형태의 문서로 변환하는 방법을 기술한 표준이다. 그림-1은 XSL을 통하여 XML 인스턴스가 HTML로 변환되는 과정을 보이고 있다.



(그림 1) XSL 프로세서의 동작 과정

XSL에서의 가장 기본적인 빌딩 블록은 생성자 룰(creator rule)이다. 생성자 룰은 XML의 엘리먼트가 어떻게 변환되어 디스플레이 될 것인가에 대한 기술로서 패턴(pattern)과 액션(action)으로 구성된다. 패턴은 XML 엘리먼트의 타입을 정의하는 부분이고, 액션은 패턴에 정의되어 있는 엘리먼트가 어떻게 보이게 되는지 기술하게 된다.

XSL 프로세서는 이와 같은 생성자 룰을 이용하여 XML 문서를 HTML, RDF, 또는 기타 문서의 형태로 변환하게 하는 소프트웨어이다. XML 브라우저는 브라우저 내에 이와 같은 XSL 프로세서를 내장하고 있어야 하며, 화면상에는 XSL 프로세서에 의하여 변환된 문서를 보게 되는 것이다.



(그림 2) XSL 생성자 룰

HTML의 스타일로 CSS를 사용하며, XML의 스타일로 XSL을 사용하는데, 표-5는 CSS와 XSL의 차이점을 나타내었다.

(표 5) CSS와 XSL 비교

비교 내용	CSS	XSL
HTML과 같이 사용 가능한가?	예	아니오
XML과 같이 사용 가능한가?	예	예
변환 언어인가?	아니오	예
기반으로 하는 문법은?	CSS	XML

CSS와 XSL은 각각 HTML과 XML 같이 다른 영역에서 사용될 것이다. 이 둘은 같은 사항에 대하여 반대의 개념이 아닌 서로 공존하게 될 것이다. 특히 XML의 스타일은 HTML 또는 DSSSL로 변환하는 것을 기반으로 표준안이 작성되었기 때문에 XSL 내부의 생성자 톨의 액션 부분에 단순한 HTML 태그가 아닌 CSS를 사용하면 보다 다양한 효과를 볼 수 있을 것이다.

3.3 DOM (Document Object Model)

DOM은 일반적인 웹 페이지에 포함되어 있는 핸들링 가능한 객체(엘리먼트, 링크, 이미지 등)를 위한 인터페이스의 개발을 의미한다. DOM의 표준 인터페이스 사용이 보편화 될 경우 얻을 수 있는 기대 효과는 모든 웹에서 사용하는 톨이 DOM을 지원하면 문서의 저장 상태가 파일 또는 DBMS에 있던 간에 같은 인터페이스를 사용하기 때문에 접근이 용이하게 되며, 파서의 결과가 항상 DOM을 지원하게 되면 현재까지 파서의 결과가 서로 상이하야 발생하는 문제를 최소화 할 수 있게 된다.

현재 DOM 표준의 개발은 레벨 1을 종료한 상태이며, 1999년도에는 레벨 2를 위한 작업에 들어간다.

DOM의 주요한 특징은 다음과 같다.

- 독자적인 데이터 구조와 API에 대한 인터페이스 명세 제공
- DOM 인터페이스만을 통하여 문서를 만들
- 웹 상에서 상호 호환성을 높일 수 있음

3.3.1 Document Object Model의 개념

DOM은 HTML과 XML 문서를 위한 응용 프로그래밍 인터페이스(API)이며, DOM은 문서를 접근하고 조작하기 위한 방법으로 문서의 논리적 구조를 정의한다. 따라서 DOM을 이용하여 사용자들은 문서를 생성하고 그 문서의 구조에 따라

진행하고, 엘리먼트와 문서 내용을 추가/수정/삭제할 수 있다. 이렇게 하기 위해서는 웹에 대한 문서를 구조화하여야 한다.

제안된 표준 문서에 의하면 DOM의 목적이 표준화된 프로그래밍 인터페이스를 제공함으로써 환경과 응용 프로그램에 독립적으로 만드는 것이 목적중의 하나이다. 따라서 DOM은 언어에 독립적이고, 어떠한 프로그래밍 환경에서도 사용이 가능하도록 설계되어 있다.

현재 표준안은 CORBA 2.2의 OMG IDL, ECMAScript, 자바 언어 등에 대한 랭귀지 바인딩을 포함하고 있다.

3.3.2 DOM Level 1 API

DOM Level 1은 크게 Core 부분과 HTML로 구성되어 있다. level 0에는 Core, HTML, XML 이렇게 세 부분으로 구성되어 있었으나 level 1에서는 Core와 HTML에 대한 부분만 존재한다. XML 부분은 Core로 흡수되어 사실상 Core, HTML 이라고 구성되어 있는 것이다.

■ DOM Core

DOM Core는 최소한의 객체와 문서를 조작하고 접근하여 사용할 수 있는 인터페이스를 정의한 것으로 parsed HTML이나 XML문서의 내용이 손실 없이 표현을 할 수 있으며, HTML 4.0, XML 1.0을 지원하고, 문서의 계층적 구조를 생성할 수 있으며, 상위 레벨의 API가 쉽게 사용될 수 있도록 확장이 용이하게 설계되었다.

■ DOM HTML

DOM HTML은 DOM Core를 기반으로 하여 HTML 문서의 핸들링을 위한 인터페이스이다. HTML이 Core 부분과 구별되는 것은 Core는 HTML, XML 등의 문서에 사용하는 인터페이스에 대한 정의가 있으며, HTML은 HTML에 맞는(또는 필요한) 인터페이스만 추가로 정의되어 있다. DOM HTML을 이해하기 위해서는

DOM HTML Type Hierarchy의 구조를 이해하여야 한다.

3.4 메타 데이터/RDF

■ RDF Model and Syntax

RDF의 문법은 XML을 사용하여 만든 DTD를 기반으로 하고 있으며, XML을 이용하기 때문에 RDF에 대한 모델링과 데이터 조작은 XML 관련 툴 또는 프로그램을 이용하여 사용할 수 있다.

현재 "Resource Description Framework (RDF) Model and Syntax Specification"이 Recommendation으로 1999년 2월 22일 확정되었다

■ RDF Schema

RDF 스키마는 "Resource Description Framework (RDF) Model and Syntax Specification"을 기반으로 하여 RDF를 적용할 때 발생할 수 있는 문제점을 해결하기 위함이다. 즉, RDF를 사용하여 기타 메타 데이터를 수용하거나 또는 PICS, P3P(Privacy Preferences Project) 등과 같은 분야에 RDF를 적용하려면 기존의 RDFMS는 모델링과 문법 위주의 스펙이기 때문에 부족하게 된다. 따라서 RDF 스키마는 RDF 데이터 모델에서 제공하는 문장들에 대한 인터프리터 정보를 제공하는데 그 목표를 두고 있으며 이에 따라 "Resource Description Framework(RDF) Schema Specification"을 제출하였고, 이를 Recommendation으로 만드는 것을 최종적인 목표로 하고 있다.

RDF는 인터넷 정보 자원에 대한 정보를 기술하는 표준으로써 인터넷이 확산됨에 따라 그 중요성이 점점 증가하고 있다. 현재 인터넷에서는 HTML에 기반한 WWW 정보, 소프트웨어 정보를 제공하는 Archie, 특정 기관의 정보를 체계적으로 정리해 놓은 Gopher, 공용으로 활용 가능한 FTP 정보 등 다양한 형태로 각종 분야에 대한 정보를 제공하고 있다. 이와 같이 인터넷상에서 제공되는 정보는 기존에 인쇄되어진 도서나 문서 등과는

여러 가지 측면에서 다른 점을 가지고 있다. 따라서 인터넷 정보 자원의 메타 데이터 해석을 하는데 있어서 기존의 탐색 엔진들이 갖는 위와 같은 문제점들을 해결해야 할 방안이 필요하게 되었으며, 메타 데이터를 표준화된 형태로 기술하기 위해서는 메타 데이터를 정의하는 프레임워크가 필요하다. 이를 위해 인터넷의 정보 자원에 대한 메타 데이터에 대한 메타 데이터가 최소한 가져야 할 요소들을 SGML의 DTD로 정의하였는데, 이를 Dublin Metadata Core라 하며, 기타 다른 형식의 메타 데이터의 기술까지도 포함하도록 DTD를 확장하여 Warwick DTD를 정의하였다. 사용자들의 메타 데이터에 대한 인식의 확산에 따라 Dublin Core를 포함할 수 있는 RDF를 개발하기 시작하면서 그 범위를 확대하여 XML을 기반으로 RDF를 기술하고, RDF의 문법을 기반으로 RDF 응용과 PICS 2.0, P3P등을 수용하게 되었다.

향후 RDF는 웹에 대한 모든 메타 데이터를 수용할 것이며, 인터넷상의 분산된 다양한 자원들을 기술하기 위해 표준화된 프레임워크를 제공함으로써, 데이터베이스 질의 및 부분색인·검색 등이 가능하여 모든 정보를 온라인화 하는데 공헌할 것이며 그 중요성은 점점 증대될 것이다.

4. 결 론

초기의 웹 환경에 비추어볼 때 현재의 웹 환경은 괄목할 만한 성장을 가져왔으며, 이는 기존의 웹은 텍스트 기반의 구조를 통한 비동기적인 형태의 서비스 유형으로 문서검색과 단순한 표현 방식을 사용하였으나 컴퓨팅 환경과 웹 기술의 발전으로 인터넷상의 데이터들이 복잡해지고 구조화되어 가면서 다양한 멀티미디어 정보를 동기화 하여 표현하고자하는 요구가 증대되면서, 웹의 새로운 구조 및 표현방식을 요구하게 되었으며, 세계적으로 웹 사용자의 폭발적인 증가는 새로운

구조와 다양한 인터페이스 뿐만 아니라 기존의 웹과는 다른 새로운 웹 기술 개발과 더불어 다양한 서비스를 원하게 되었다.

차세대 웹서비스의 환경은 새로운 웹의 구조와 사용자 인터페이스를 바탕으로 실세계의 사회 구조를 기반으로 미래의 웹으로 나타날 것이며, 이는 현재 이루어지고 있는 많은 웹 관련 표준화 및 진행되고 있는 연구들에 의해서 완성되어 질 것이다.

본 논문에서는 현재 진행되고 있는 웹 기술 개발의 동향을 분석함으로써 차세대 웹 관련 기술이 어떻게 추진되고 있는지를 살펴보았다.

참고문헌

- [1] Internet2 Engineering WG, "Preliminary Engineering Report", [URL: <http://www.internet.edu>]
- [2] W3C HTML Home [URL: <http://www.w3.org/MarkUp/>]
- [3] Web Style Sheets, [URL: <http://www.w3.org/Style/>]
- [4] Style Sheets Activity, [URL: <http://www.w3.org/Style/Activity/>]
- [5] Cascading Style Sheets, [URL: <http://www.w3.org/Style/css/>]
- [6] Extensible Style Language (XSL), [URL: <http://www.w3.org/Style/XSL/>]
- [7] Document Object Model (DOM), [URL: <http://www.w3.org/DOM/Group/>]
- [8] Metadata and Resource Description, [URL: <http://www.w3.org/Metadata/>]
- [9] Resource Description Framework (RDF) Working Group(s), [URL: <http://www.w3.org/RDF/Group/>]



김 태 석

- 1981년 경북대학교 전자공학과 (공학사)
- 1988년 일본KEIO대학 계산기과 학전공 (공학석사)
- 1993년 일본KEIO대학 계산기과 학전공 (공학박사)

1992년-1993년 일본KEIO대학 객원연구원, 일본국제전신전화연구소(KDD) 기술자문위원
 1994년-현재 동의대학교 컴퓨터공학과 교수
 관심분야 : 자연어처리, 기계번역, 정보검색, 정보시스템



박 철 제

- 1986년 연세대학교 수학과 학사
- 1991년 일본와세다대학교 정보과학 (석사)
- 1998년 일본와세다대학교 정보과학 (박사)
- 1986년-1987년 포항중합제철 시스템개발부

1987년-1989년 Japan Knowledge Industry
 1995년-1997년 일본 와세다대학교 미디어 네트워크 센터 특별연구원
 1996년-1998년 현대정보기술연구소 책임연구원
 1998년-현재 동의대학교 경영정보학과 교수
 관심분야 : 인터넷/인트라넷, 자연어처리, 기계번역, 정보검색, 경영정보시스템



임 한 섭

- 1992년 부산공업대학교 전자공학과 (공학사)
- 1996년 부산외국어대학교 컴퓨터공학과 (공학석사)
- 1996년-1997년 서울시스템 선임연구원

1998년-현재 금아시스템(주) S/W 개발부 실장
 1996년-현재 웹코리아 부산 의장
 관심분야 : 원격교육, 전자상거래, 인터넷 기술, 분산 데이터베이스 시스템