튜 토 리 얼 Ⅳ

멀티미디어 문서 표준 - SGML, XML

▷ 연 사 : **정 회 경** 박사 (배재대)
▷ 사 회 : **고 견** 교수 (청주대)

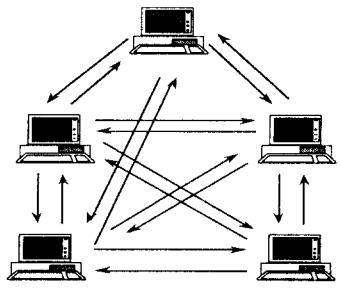
멀티미디어 문서 표준
- SGML, XML -

배 재 대 학 교
컴퓨터공학과
정 회 경
E-mail : hkjung@mail.paichai.ac.kr
URL : http://mie.paichai.ac.kr

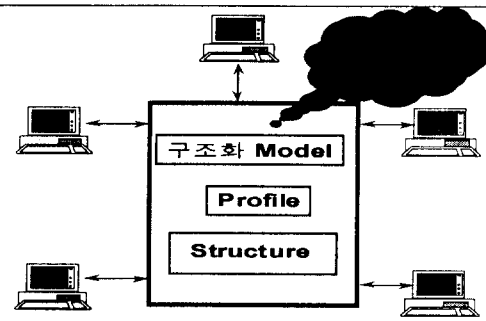
목 차

- 서론
- **SGML** 개요
- **HTML** 개요
- **XML** 개요
- **SGML/HTML/XML** 간의 관계

문서교환의 문제점

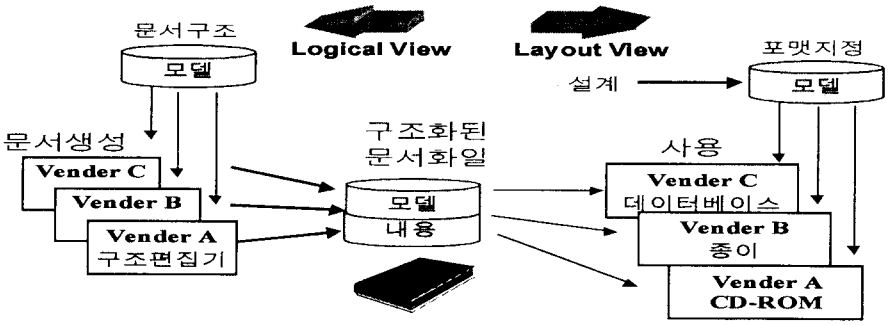


N x (N-1) 변환



2 x N 변환

구조화된 멀티미디어 문서 처리



SGML은 무엇인가 ?

- **Definition of SGML ?**
  - Standard Generalized Markup Language, ISO 8879, 1986
  - 한 문서에서 논리구조와 내용 구조를 기술하기 위한 메타 언어
- **SGML은 다음을 제공**
  - 기술적 문서 마크업을 생성하는 기법
  - 문서의 지능적 내용 이나 구조를 정의하기 위한 언어
  - 다양한 응용들 사이 구조화된 데이타를 교환을 위한 도구
- **SGML 문서의 구성**
  - **SGML Declaration** : 문서를 기술하기 위해 사용된 구체 구문을 설명
  - **Document Type Definition(DTD)**
    - Entity, Element, Attribute, 문서 타입 선언, 표기법 선언, 주석 선언
  - **Document Instance :** DTD에 따른 내용과 태그로 구성된 실제 문서

SGML 개념 및 문법(SGML 선언)

- 예 : **HTML 선언**
  - <!SGML "ISO 8879:1986" -- HTML에 대한 SGML 선언--
    CHARSET
  - BASESET "ISO 646:1983//CHARSET International Reference Version(IRV)//ESC 2/5 4/0"
  - DESCSET 0 9 UNUSED
    9 2 9
  - CAPACITY PUBLIC "ISO 8879:1986//CAPACITY Reference//EN"
    SCOPE DOCUMENT
  - SYNTAX PUBLIC "ISO 8879:1986//SYNTAX Reference//EN"
    FEATURES
  - MINIMIZE DATATAG NO OMITTAG YES RANK NO SHORTTAG YES
  - QUANTITY ... LITLEN 1024 ... NAMELEN 72..... >

SGML 개념 및 문법 (문서형 정의)

- **DTD : Document Type Definition(문서형 정의)**
  - 문서구조와 내용 설명
  - 문서 체계를 정의
- **DTD 내용**
  - Document type declaration
  - Element type declaration(s)
  - Attribute list declaration(s)
  - Entity declaration(s)
  - Notation declaration(s)
  - Comment declaration(s)
  - Short reference(s)

SGML 개념 및 문법 (문서형 정의)

- **Document Type Definition**
  - 문서의 특정 부류를 마크업 하기 위해 사용될 선언의 모든 타입을 설명
  - 첫번째 선언
  - 선언 subset으로 시작
  - 예

```
< !DOCTYPE   book  [
        other declarations
    ]>
```


SGML 개념 및 문법 (Element 선언)

- 문서 논리구조의 형식적 정의
- Element 명 지정
- 마크업 최소화(Minimization) 허용
- 내용 모델(내용과 구조를 정의)
  - 연결자[seq(,), and(&), or(|)], 발생지시자[opt(?), plus(+), rep(*)]
- 선언된 내용(PCDATA, RCDATA, EMPTY, CDATA)
- 임의 예외 지정
- 예
  - <!ELEMENT chapter - O (title, section+) +(fn)>
  - <!ELEMENT title - - (#PCDATA)>
  - <!ELEMENT section - - (title, para+)>
  - <!ELEMENT fn - - (#PCDATA) -(fn)>
  - <!ELEMENT (send, receive) - - (name, age? | address*) >

SGML 개념 및 문법 (속성 선언)

- Element 특성 설명
- Element당 하나의 속성 리스트를 갖음
- 값
  - 키워드
    - #FIXED, #REQUIRED, #CURRENT, #IMPLIED, #CONREF
  - 리터럴

예 - DTD에서

```
<!ELEMENT letter - - (front, body, rear)>
<!ATTLIST   letter    status (draft | final ) "draft"
                      author CDATA #REQUIRED)>
```

- 실제 문서에서 : <letter status = "final" author = "jhk">

SGML 개념 및 문법 (엔티티 선언)

- 문서에서 여러 번의 반복 입력이 필요한 경우에 사용
- 참조되기 전에 선언
- 두 타입
  - **general entities**
    - "&"사용
    - 블럭이나 텍스트를 참조시 사용
    - 예
      - DTD에서 : <!ENTITY univ "Paichai">
      - 실제 문서에서 : &univ; university
  - **parameter entities**
    - "%"사용
    - 자주 사용되는 내용 모델이나 속성을 참조시 사용
    - 예

```
<!ENTITY % text "(#PCDATA | xref)+">
<!ENTITY % list  "(seqlist | ranlist)">
<!ENTITY % para "(%text; | %list;)+">
```


SGML 개념 및 문법 (엔티티 선언)

- 교체될 데이터 위치
  - 내부엔티티(Internal Entity)
    - 처리중인 SGML 문서 내에 존재
  - 외부엔티티(External Entity)
    - 한 시스템 내에서만 사용 가능**(SYSTEM)**
    - 두개 이상 시스템에서 사용 가능**(PUBLIC)**
  - (예)
    - <!ENTITY ch3 SYSTEM "/usr/text/chapter3.doc" >
    - <!ENTITY % HTMLlat1 PUBLIC "-//IETF//ENTITIES Added Latin 1 for HTML//EN">

SGML 문법 및 개념(Notation 선언)

- 비 SGML 데이터(NDATA)의 데이터 내용 표기를 정의
  (적절한 S/W가 데이터를 처리)
- External entities로 사용
- 예

```
<!NOTATION a-w PUBLIC "
-//addison --wesley //NOTATION maths //EN">
<!NOTATION tex PUBLIC "-//Local //NOTATION
Tex Fomula //EN">
<!ELEMENT eqn RCDATA -- EQUATION -->
<!ATTLIST eqn type NOTATION (a-w | tex) a-w >
```


SGML 문법 및 개념(Comment 선언)

- 실제 문서의 구조나 내용과는 무관
- 두 유형
  - 선언
    - <!-- comment sample -->
  - 다른 선언 내에 사용
    - <!ATTLIST list type (a|b|c) a -- a=alpha -->

SGML 문서 예

Dec 12, 1998
Ref : 980000/PCP

대전 서구 도마동 배재대학교
컴퓨터공학과 Mr. Jung

멀티미디어 문서표준

Dear Mr. Jung.

이번 세미나에서 .....

SGML은 .....

I am looking forward to ....

Sincerely,

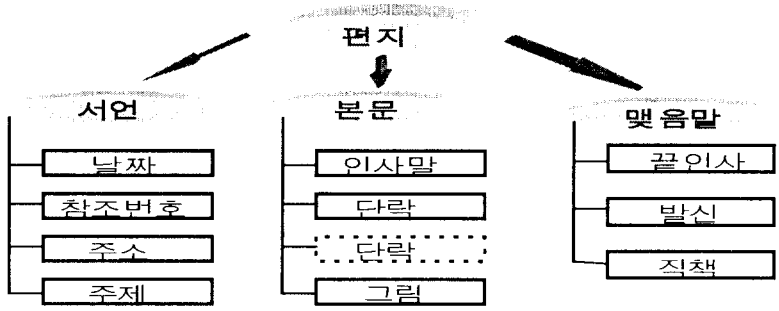
Paichai University
Jung
Prof.

서언
(날짜, 참조번호,
주소, 주제)

본문
(인사말, 단락,...)

맺음말
(끝인사, 발신, 직책)

문서예의 논리구조

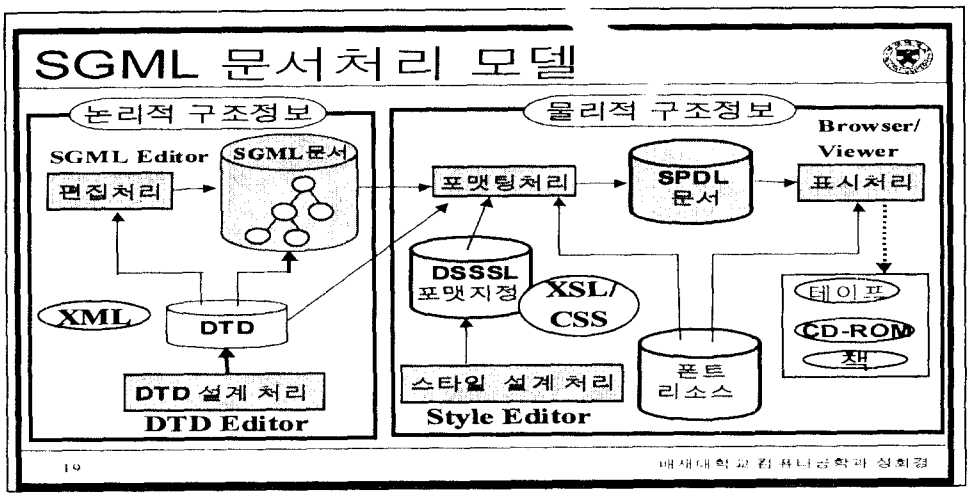


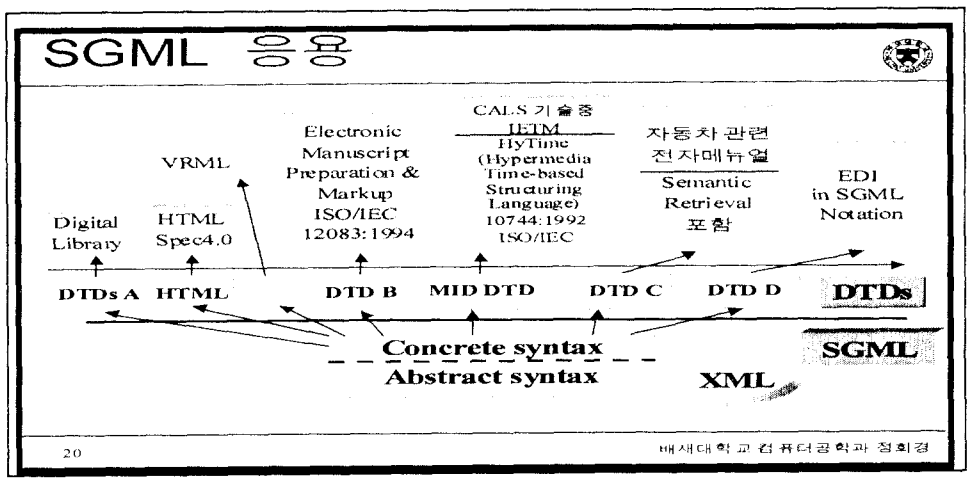
문서예에 대한 DTD

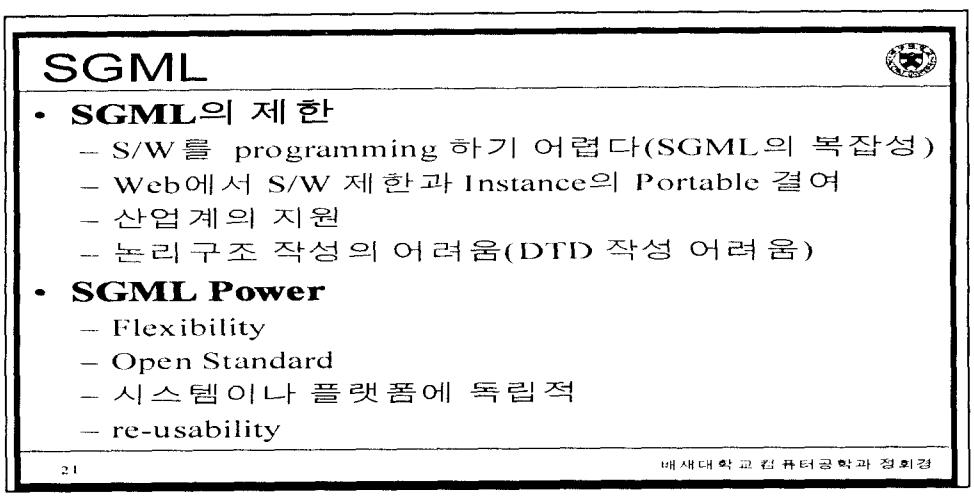
```
<!- - DTD for simple office letter - - >
<!DOCTYPE 편지 [
<!NOTATION tex PUBLIC "-//Local//NOTATION Tex Fomula//EN">
<!NOTATION mat PUBLIC "-//addison-wesley//NOTATION math//EN">
<!ELEMENT 편지 -O (서언, 본문, 맺음말)>
<!ATTLIST 편지 STATUS (public|confiden) public>
<!ELEMENT 서언 -O ((날짜 & 참조번호), 주소, 주제?)>
<!ELEMENT 날짜 -O (#PCDATA) >
<!ELEMENT 참조번호 -O (#PCDATA) -(참조번호) >
<!ATTLIST 참조번호 refid IDREF #CONREF >
<!ELEMENT 주소 -O (#PCDATA) >
<!ELEMENT 주제 -O (#PCDATA) >
<!ELEMENT 본문 -O (인사말, 단락+) >
<!ELEMENT 인사말 -O (#PCDATA) >
<!ELEMENT 단락 -O (#PCDATA | 그림*) >
<!ELEMENT 그림 - O EMPTY>
<!ATTLIST 그림 유형 NOTATION (tet | mat)>
<!ELEMENT 맺음말 -O (끝인사, 발신, 직책?) >
<!ELEMENT 끝인사 -O (#PCDATA) >
<!ELEMENT 발신 -O (#PCDATA) >
<!ELEMENT 직책 -O (#PCDATA) >        ]>
```


문서예의 Markup

```
<편지 STATUS=confiden>

<서언>
<날짜>Dec 12, 1998</날짜>
<참조번호>Ref:9800</참조번호>
<주소>대전 서구 도마동 배재대학교
컴퓨터공학과  Mr. Jung</주소>
<주제>멀티미디어 문서표준</주제>
</서언>

<본문>
<인사말>안녕하십니까 ?</인사말>
<단락>첨부된 프로그램. ....</단락>
<단락>첨부된 발표자료.....</단락>
</본문>

<맺음말>
<끝인사>감사합니다.</끝인사>
<발신>배재대 컴퓨터 공학과</발신>
<직책>조교수</직책>
</맺음말>

</편지>
```

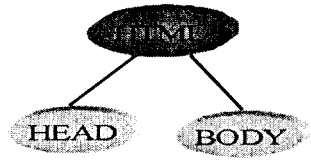

SGML 문서처리 모델
논리적 구조정보
물리적 구조정보
SGML Editor
편집처리
SGML문서
XML
DTD
DTD 설계 처리
DTD Editor
포맷팅처리
DSSSL
포맷지정
XSL/
CSS
스타일 설계 처리
Style Editor
SPDL
문서
Browser/
Viewer
표시처리
폰트
리소스
테이프
CD-ROM
책
19
배재대학교 컴퓨터공학과 정회경


SGML 응용
Digital
Library
HTML
Spec4.0
VRML
Electronic
Manuscript
Preparation &
Markup
ISO/IEC
12083:1994
CALS 기술중
HTML
HyTime
(Hypermedia
Time-based
Structuring
Language)
10744:1992
ISO/IEC
자동차 관련
전자메뉴얼
Semantic
Retrieval
포함
EDI
in SGML
Notation
DTDs A
HTML
DTD B
MID DTD
DTD C
DTD D
DTDs
Concrete syntax
Abstract syntax
SGML
XML
20
배재대학교 컴퓨터공학과 정회경


SGML
• SGML의 제한
– S/W를 programming 하기 어렵다(SGML의 복잡성)
– Web에서 S/W 제한과 Instance의 Portable 결여
– 산업계의 지원
– 논리구조 작성의 어려움(DTD 작성 어려움)
• SGML Power
– Flexibility
– Open Standard
– 시스템이나 플랫폼에 독립적
– re-usability
21
배재대학교 컴퓨터공학과 정회경

HTML(HyperText Markup Language)

- W3C의 Specification
- 웹상에서 Hypertext 문서를 만들 수 있는 간단한 마크업 언어
- SGML의 한 응용(HTML=SGML Declaration + A DTD)
- **HTML 제한**
  - Fixed tag set
  - page layout
  - 임의 구조화 능력 부족
  - 효과적인 검색, 재사용, 검증이 불편
- **HTML Power**
  - 이식성과 사용이 편리
  - HTML Instance를 Web에서 쉽게 download 가능



XML 이란 무엇인가 ?

- eXensible Markup Language - Meta Language
- W3C working group에서 1996년도에 작성
- Web상에서 구조화된 문서를 전송가능토록 설계된 표준화된 텍스트 형식
- SGML/HTML과 Interoperability
- SGML의 한 응용
- 일반 Web 응용 S/W에 의해 처리되고 사용자-정의 문서를 다양하게 배치 가능
- 배경
  - HTML의 한계 : Web Publishing 요구, data checking, 구조 확장
  - SGML의 복잡성과 방대 함
- **XML 1.0 Spec.**
  - 확장 마크업
  - 확장 hypertext linking/stylesheet mechanism

<?XML!>

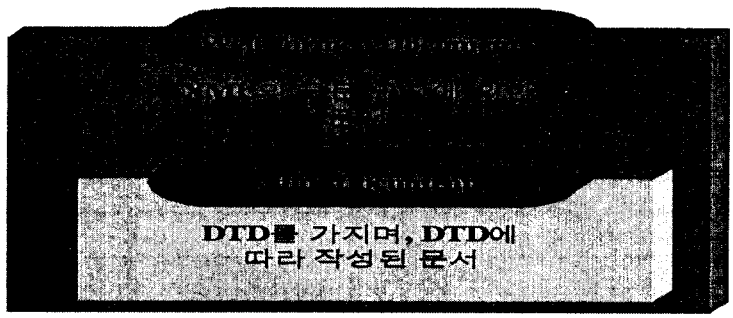
XML 특징

- 간단성
- SGML에 기본
- 지금까지의 경험에 기본(SGML, HTML)
- 확장성(태그에 대한 제한 없음)
- 링크의 자동 생성이 가능
- 차세대 하이퍼텍스트 기능을 제공
- 문서의 재사용 용이
- 플랫폼, 응용에 독립적
- 웹 사이트들의 간단한 시스템 관리 제공
- Primary character set으로 8-bit ASCII와 Unicode를 사용

XML 활동 개요

- **Part 1 (xml-lang)** : 구현의 용이성을 위한 SGML (ISO 8879)의 부분집합
- **Part 2 (xml-link)** : HyTime (ISO/IEC 10744)과 Text Encoding Initiative (TEI)의 확장 포인터 메커니즘에 기본한 표준 하이퍼텍스트 메커니즘 설계
- **Part 3 (xml-style)** : DSSSL (ISO/IEC 10179)의 부분집합에 의한 임의 구조화된 정보에 대한 표준 스타일 시트 언어를 설계
- **XUA(XML User Agent)**

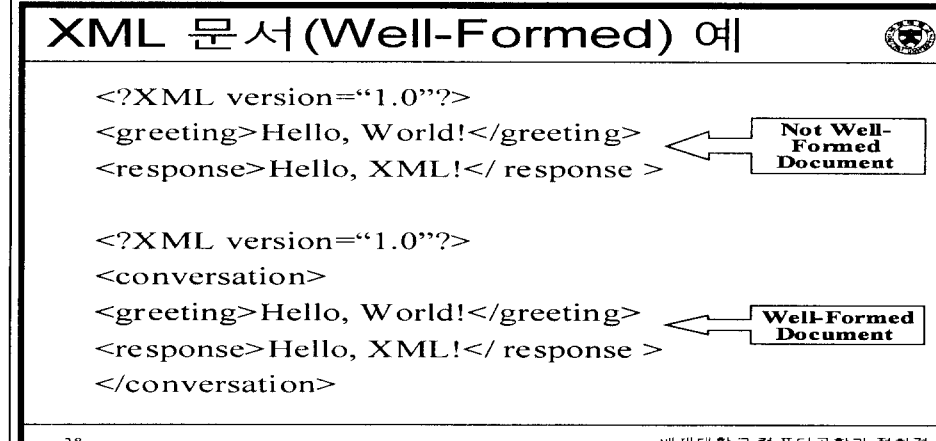
XML 문서



XML 문서(Well-Formed)

- Document instance must conform to the grammar of XML documents
- Looks right, but logical structure is not validated
- Need to have a single outermost element (root, document element)
- Elements may not overlap
- All tags balanced(except empty)
- EMPTY elements in form of <tag/> or have end tag
- All attribute values in quotes
- Attributes must be of type CDATA
- Has Required Markup Declaration(RMD)
- All entities declared
- No Markup characters in character data

XML 문서(Well-Formed) 예



XML 문서(Valid)

- 적절한 Document Type Definition(문서형 정의)을 포함
- 문서는 포함된 DTD의 제한에 따른다
- 문서형 정의는 선택적인 처리 명령과 주석 다음으로 문서내에서 첫부분에 선언
- 예
  - DTD와 인스턴스를 별개 파일로 관리

```
<?XML version = "1.0" encoding="UTF-8" RMD='INTERNAL'?>
<!DOCTYPE sample SYSTEM "sample.dtd" >
<greeting>Hello, world !</greeting>
```

DTD

  - DTD와 인스턴스를 한 파일로 관리

```
<?XML version = "1.0" encoding="UTF-8" RMD='INTERNAL'?>
<!DOCTYPE sample [ <!ELEMENT greeting (#PCDATA) > ]>
<greeting>Hello, world !</greeting>
```

DTD

XML 문서(DTD)

- 태그들에 대한 문법 규칙들 집합
- XML DTD는 다음을 포함
  - Elements
  - Attributes
  - Entities
  - Notations
  - Comments
  - Processing Instructions
  - CDATA Sections

XML 문서 구조

- 엘리먼트 선언
  - 내용 특성을 정의
  - 시작 태그(<element>)로 시작하고, 종료태그(</element>)로 종료
  - 공백 엘리먼트 : <tag-name/>

발생지시자(+, ?, *), 연결자 (',' '|' '&'[X])를 사용

```
<!ELEMENT a (#PCDATA | %font | %special | %form)* >
<!ELEMENT spec (front, body, back?)>
```

- 속성 선언
  - 엘리먼트에 대한 속성값 선언
  - 모든 속성값들은 quote

```
<!ATTLIST list
        type (bullets | ordered | glossary) "ordered">
<!ATTLIST form
        method  CDATA  #FIXED "POST" >
```


XML 문서 구조

- 엔티티 선언
  - '&'로 시작, ';'으로 종료(예 : &sample;)
  - 특수 문자 들(XML 예약 문자들)을 표현하는데 사용
    - decimal 참조 : N / hexadecimal 참조 : &#x1E;
  - 반복되거나 교체되는 텍스트를 참조하는데 사용

```
• <!ENTITY PCU "PaiChai University">
• <!ENTITY %STR "TITLE, SUBJECT">
• <!ENTITY % open-hatch SYSTEM "http://www/openhatch.html">
```

- 주석
  - 의미를 갖지 않으며 '<!--'로 시작, '-->'로 종료
  - 선언이나 태그 내에 올 수 없다.
  - 문자열 "--"를 제외한 임의 데이터를 포함

```
<!-- 주석 부분입니다 -->
```


XML 문서 구조(처리 명령 : PI)

- 응용들에 정보를 제공하기 위해 사용
- 형식 : <?name pidata?>
  - name : PI target(응용에 PI를 정의)
  - pidata : optional
  - PI names beginning with XML are reserved for XML standardization
- XML PI Options
  - version : 현재는 "1.0" 만 유용
  - rmd (required markup declaration)
    - internal : 내부 선언이 처리
    - all : 내부와 외부 선언이 처리(외부보다 내부가 우선)
    - none : 선언 없이 처리
  - encoding
    - UTF-8, UTF-16, ISO-10646-UCS-2, ISO-10646-UCS-4
    - ISO-8859-*, ISO-2022-*, EUC-*
    - 예 : <?XML ENCODING='UTF-8'?>

XML 문서(CDATA)

- CDATA부분은 파서에 마크업 문자를 무시하도록 지시
- 문자데이터의 모든 곳에 위치가능
- 충첩되지 않는다
- 모든 문자 데이터는 응용에 직접 전달
- CDATA 부분내 주석은 인식되지 않음
- 예 :

```
- <!CDATA[<greeting>Hello, world !</greeting>]]>
- <![CDATA[*p = &q; b = (i <= 3);]]>
```


XML 문서예 일부

DTD

```
<!ELEMENT FAQ          (INFO, PART+)>
<!ELEMENT INFO         (SUBJECT, AUTHOR, EMAIL?,
   VERSION?, DATE?)>
<!ELEMENT SUBJECT      (#PCDATA)>
<!ELEMENT AUTHOR       (#PCDATA)>
<!ELEMENT EMAIL        (#PCDATA)>
<!ELEMENT VERSION      (#PCDATA)>
<!ELEMENT DATE         (#PCDATA)>
```

실례

```
<?XML VERSION="1.0" ENCODING="UTF-8" RMD="NONE"?>
<!DOCTYPE FAQ SYSTEM "FAQ.DTD">
<FAQ>
    <INFO>
        <SUBJECT>XML</SUBJECT>
        <AUTHOR>Hoekyung Jung</AUTHOR>
        <EMAIL>hkjung@mail.paichai.ac.kr </EMAIL>
        <VERSION>1.0</VERSION>
        <DATE>Dec. 12, 98</DATE>
    </INFO>
```


XLL : XML Linking

- TEI(Text Encoding Initiative)와 HyTime(Hypermedia/Time-based Sturcturing Language) 표준에 기본
- 양방향, 다중 방향 링크 기능 제공
- XLL은 두 부분(Xlink, Xpointer)으로 표현
- **XML-LINK Types**(선언가능한 값)
  - SIMPLE, EXTENDED, LOCATOR, GROUP, DOCUMENT
- **Link Behavior**
  - SHOW axis
    - 링크 운행시 문서가 윈도우내에 어떻게 표현될 지 결정
    - 값 : EMBED, REPLACE, NEW
  - ACTUATE axis
    - 링크가 어떻게 운행되는 지 결정
    - 값 : AUTO, USER

XLL : XML Linking

- **Extended Pointers**
  - 문서의 엘리먼트 트리내 자원를 지정하도록 허용
  - XPointer 구조
    - locator는 하나나 두개의 Xpointers를 포함가능; 두개이면 "**..**"로 구분
    - 인수 : (INSTANCE, NodeType)
      - Instance : ALL, +, -
      - CHILD(2,CHAP)(4,SEC)(3, PER)
      - CHILD(1,CHAP) .. CHILD(-1,CHAP)

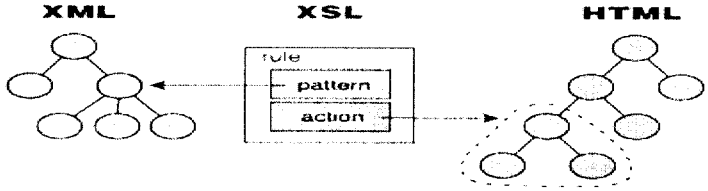
XSL : Style sheet

- XML 데이터 및 문서와 함께 사용되도록 설계된 스타일 시트 언어
- **XSL 처리**

- 출력 : HTML, RTF, 텍스트, etc.

XSL : Style sheet

- XSL의 기본 블록은 *construction rule* 이다
- construction rule은 특정 엘리먼트가 어떻게 디스플레이 가능한 출력으로 변환되는 지를 기술
- **construction rule의 구성 :**
  - *pattern* : XML 소스 엘리먼트의 유형을 정의
  - *action* : 패턴과 정합되는 엘리먼트가 어떻게 표현되는지 기술



XSL : Style sheet

– XSL stylesheet의 기본구조 예

```
<xsl>
 <rule>
  <target-element type="change-date"/>
  <P color="blue" font-style="italic">
   <children/>
  </P>
 </rule>
</xsl>
```

pattern

rule

– XML 문서 예

```
<document>
 <change-date>3 October 1997</change-date>
</document>
```

– 결과

3 October 1997

In HTML

```
<P STYLE="color:blue; font-style:italic">3 October 1997</P>
```


XML 응용

- **Web, Internet, Software**
  - Channel Definition Format (CDF)
  - Open Software Description (OSD)
  - WebDAV(웹상에서 분산 저작 및 버전관리)
- **Multimedia, Graphics, Speech**
  - Synchronized Multimedia Integration Language (SMIL)
  - Precision Graphics Markup Language (PGML)
  - Java Speech Markup Language (JSML)
  - Music Markup Language(MusicML)
  - Vector Markup Language(VML)
- **Metadata, Archival**
  - Resource Description Framework (RDF)
  - Meta Content Framework (MCF)
  - Web Interface Definition Language (WIDL)
  - Encoded Archival Description (EAD)
  - Genealogical Data in XML(GedML)

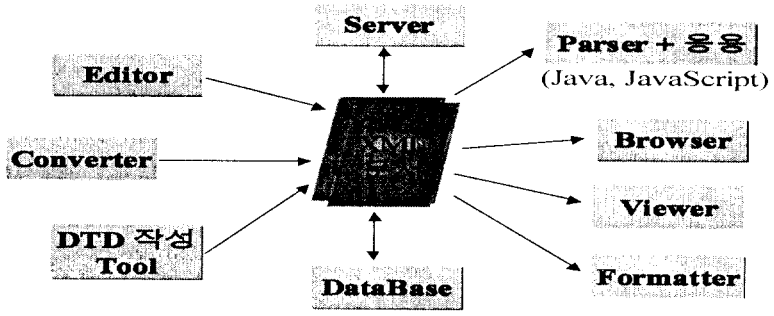
XML 응용

- **Scientific**
  - Mathematical Markup Language(MathML)
  - Chemical Markup Language(CML)
  - Bioinformatic Sequence Markup Language (BSML)
  - Telecommunication Interchange Markup (TIM)
- **Finance, Commerce, Business Information**
  - Open Financial Exchange(OFX)
  - XML/EDI
  - Open Trading Protocol(OTP)
  - Information & Content Exchange(ICE)
  - Product Information Exchange(PIX)
  - Open Buying on the Internet(OBI)
  - Common Business Language(CBL)
- **API**
  - **Document Object Model(DOM)**

XML 응용

- **Language, Knowledge Representation**
  - Translation Memory eXchange(TMX)
  - Ontology Markup Language(OML)
  - Conceptual Knowledge Markup Language(CKML)
  - OpenTag
- **Education**
  - Tutorial Markup Language(TML)
- **Digital Libraries**

XML Tool



XML

- **XML 제한**
  - 많은 SGML의 복잡한 특성 제한
- **XML Power**
  - Web상에서의 Network delivery(SGML과 차이점)
  - DTD와 Stylesheet를 사용자가 쉽게 수정
  - 다양한 Linking Type 제공
- 추가적인 정보
  - http://www.w3.org/pub/WWW/XML/
  - http://www.ucc.ie/cgi-bin/PUBLIC/

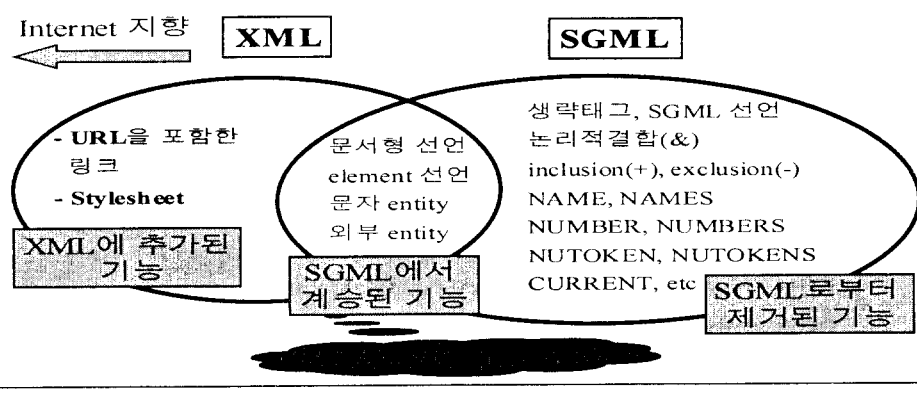
결론(XML & HTML)

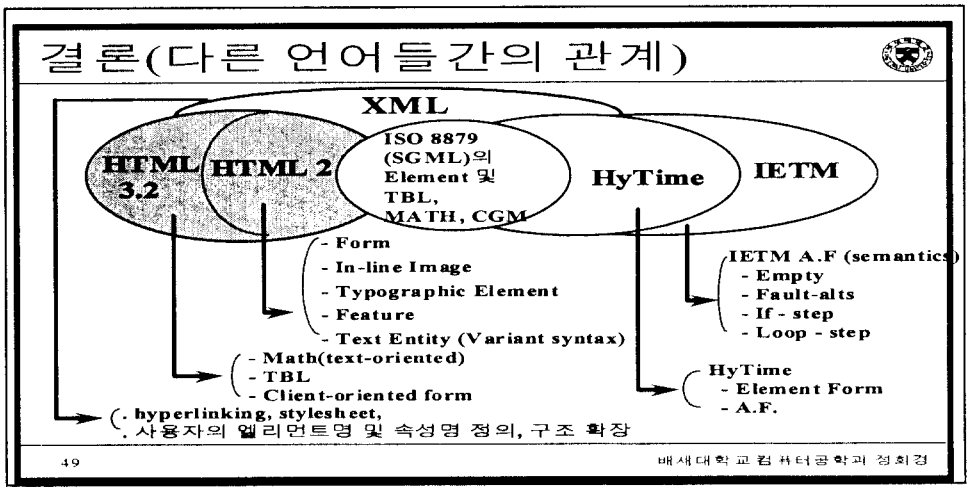
- **HTML**
  - 하나의 문서 유형(하나의 태그 집합)
  - 다소의 SGML 문법 수용
- **XML**
  - 태그와 문서 유형 집합을 생성하는 방법 제공
  - SGML 문법의 부분집합
  - Extensibility
  - Structure
  - Validation
- XML ⇄ HTML

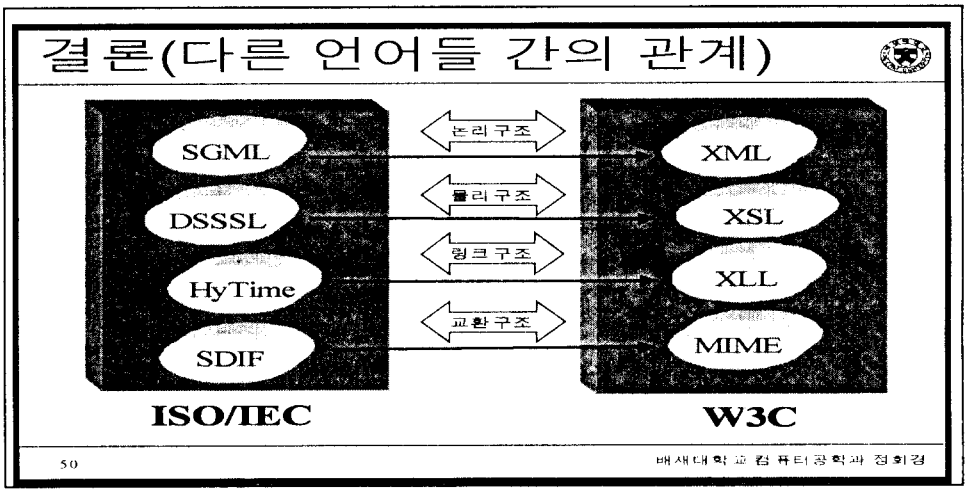
결론 (XML & SGML)

- **SGML**
  - 유연성
  - 프로그램하기 어려운 많은 선택사항 포함
- **XML**
  - 임의 특성을 제외한 SGML
  - SGML보다 덜 유연성을 가짐
- XML ⇄ SGML

결론(SGML & XML)




결론(다른 언어들간의 관계)
XML
HTML 3.2
HTML 2
ISO 8879 (SGML)의 Element 및 TBL, MATH, CGM
HyTime
IETM
- Form
- In-line Image
- Typographic Element
- Feature
- Text Entity (Variant syntax)
IETM A.F (semantics)
- Empty
- Fault-alts
- If - step
- Loop - step
- Math(text-oriented)
- TBL
- Client-oriented form
HyTime
- Element Form
- A.F.
. hyperlinking, stylesheet,
. 사용자의 엘리먼트명 및 속성명 정의, 구조 확장
49
배재대학교 컴퓨터공학과 정희경


결론(다른 언어들 간의 관계)
SGML
DSSSL
HyTime
SDIF
논리구조
물리구조
링크구조
교환구조
XML
XSL
XLL
MIME
ISO/IEC
W3C
50
배재대학교 컴퓨터공학과 정희경