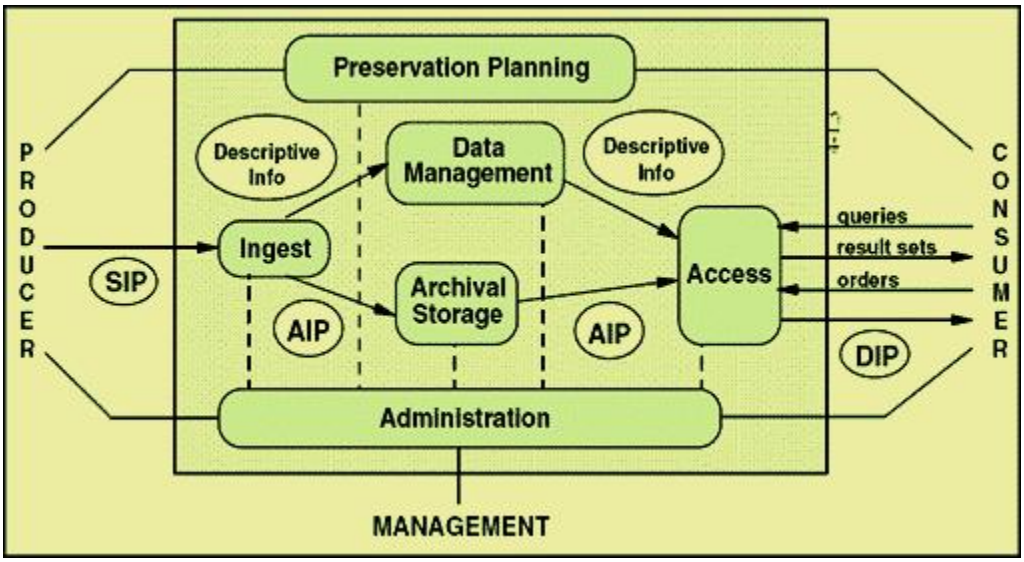
2016 정보검색 보조교재

Annotated by S.H. Kim



대구대학교 사회과학대학 문헌정보학과

>>BEGINNING:

*Indexing/Retrieval - Basics

►2 Main Stages
-Indexing process: 사전 처리하여 레포지터리에 정보를 저장하는 것을 포함한다 - an *index*
-Retrieval/runtime process: 쿼리를 작성하여 색인에 접근한 다음에 그 쿼리에 relevant한 도큐를 찾는 것을 포함한다.

►Basic Concepts:
-Document:
any piece of information (book, article, database record, Web page, image, video, song)
•대체로 텍스트 데이터

-Query:
이용자의 정보요구를 대표하는 어떤 텍스트

-Relevance: 도큐와 쿼리인 R(d,q) 간의 이진적 관계(a predicate)

p. lxvi

*** metadata: "data about data".**
이 용어의 개념이 모호한데 그 이유는 기본적으로 두 가지의 다른 개념을 갖고 있기 때문이다

► Structural metadata; 데이터 구조의 디자인과 규정에 대한 것으로, 보다 정확하게는 "data about the containers of data"라 부른다.
► Descriptive metadata; 응용 데이터의 개별적 경우(instances), 즉, the data content에 관한 것이다. 따라서 "data about data content" or "content about content"이므로 metacontent라고 부르기도 한다.

Metadata (metacontent)는 전통적으로 도서관 목록에서 찾을 수 있다. 정보가 점점 더 디지털화됨으로써, 메타데이터 역시 특별한 학문영역에 전문적으로 적용되는 메타데이터 표준을 사용하여 디지털 데이터를 기술하는데 이용되고 있다. 데이터 파일의 contents와 context를 기술함으로써, 오리지널 데이터/파일의 질이 크게 높아진다. 예를 들어, 이용자의 경험을

자동적으로 갱신시킬 수 있는 브라우저에 그것의 웹페이지 사용언어, 제작도구, 그리고 어떤 주제에 대해 더 알고 싶은 경우에 가야할 장소나 사이트를 자세하게 기술하고 있는 메타데이터를 포함시킬 수 있다.

> Definition

Metadata (metacontent)란 데이터와 관련해서 다음과 같은 한 가지 이상의 요소에 대한 정보를 제공하는 데이터이다;

1) Means of creation of the data
2) Purpose of the data
3) Time and date of creation
4) Creator or author of the data
5) Location on a computer network where the data were created
6) Standards used

예) a digital image로 그림을 그릴 때, 그 그림의 크기, 색도, 해상도 등에 관한 metadata 가 포함될 수 있다. 또한 텍스트 문서의 메타데이터에는 문서의 길이, 저작자명, 작성시기, 그리고 요약문에 대한 정보가 포함될 수 있다.

Metadata도 데이터이다. 따라서 메타데이터는 데이터베이스에 저장되어 관리될 수 있다. 종종 이러한 데이터베이스를 Metadata registry 또는 Metadata repository라 부른다. 그렇지만, the context와 a point of reference가 없다면, 레포지토리는 메타데이터를 단지 살펴보기만 할 뿐이며, 그 속에 있는 메타데이터를 판별해내지는 못할 수도 있다.

예를 들어, 모두 13 digits로 이루어진 복수의 번호를 포함하고 있는 데이터베이스는 그 자체가 계산의 결과이거나 방정식에 사용할 숫자의 리스트일 수도 있다 -- 만일 어떤 다른 context가 없다면, 번호들 그 자체가 숫자 데이터로 인식될 수 있다는 것이다. 그러나 이 데이터베이스에 장서수집용 기록(log)이 라는 context가 주어진다면, 이 13자리 번호들은 ISBNs - 그 책속에 들어 있는 정보 자체가 아니라 그 책에 대하여 말하는 정보- 으로 인식될 것이다.

"메타데이터" 용어는 1968년 **Philip Bagley**가 자신의 저서 「Extension of programming language concepts」에서 처음으로 사용하였다. 그는 데이터 콘텐트인 각각의 경우(instances)에 관한 콘텐트 또는 도서관 목록에서 주로 발견되는 데이터의 종류인 메타콘텐트라는 대안적 의미보다는 구조적 데이터 즉, 데이터의 콘테이너에 관한 데이터라는 ISO 11179의 "전통적 의미"로 이 용어를 사용하였다.

그 후로 정보관리, 정보학, 정보기술, 도서관학, 그리고 GIS에서 이 용어를 널리 채택하였다. 이들 분야에서 메타데이터란 단어는 "데이터에 대한 데이터"로 정의하고 있으며, 이것이 일반적인 정의인 반면에, 또 다른 여러 분야에서는 보다 협의적인 정의를 채택하고 있다.

> Metadata types

Bretheron & Singley (1994)의 two distinct classes: structural/control metadata

and guide metadata:

► Structural metadata: 구조적 메타데이터
tables, columns and indexes와 같은 컴퓨터 시스템의 구조를 기술하기 위하여 사용한다.
► Guide metadata: 안내용 메타데이터
누군가에게 특별한 아이템을 찾도록 도와주기 위하여 사용되며, 대부분 자연어로 된 키워드의 세트로 표현된다.

Ralph Kimball는 유사한 2 가지 categories: technical metadata와 business metadata를 주장하였다. 여기서 Technical metadata는 internal metadata이고, business metadata는 external metadata를 말한다. 또한 Kimball은 a third category로 process metadata를 추가하였다.

NISO에서는 3 types of metadata: descriptive, structural and administrative로 구분하고 있다. Descriptive metadata란 title, author, subjects, keywords, publisher와 같은 사물(object)를 탐색하고 그 위치를 설정하기 위해 사용되는 정보이고, structural metadata란 사물의 구성요소의 조직 방법에 대한 기술이며, administrative metadata란 file typ과 같은 technical information이다. 그리고 Two sub-types of administrative metadata으로 rights management metadata와 preservation metadata가 있다.

>> 기술 메타데이터(Technical Metadata)

기술 메타데이터는 디지털 캡쳐 과정에서 생산 및 제작 정보, 다시 말해서, 미래에 디지털 자원의 재생을 위하여 디지털 사물, 주파일과 보조 파일의 포맷, 해상도, 칼라, 기록파일 등을 수집하기 위하여 사용된 하드웨어와 소프트웨어에 대한 정보를 갖고 있는 디지털 사물의 기술적 속성을 기록하기 위하여 사용된다. 이런 이유로 기술적 메타데이터는 행정 그리고 보존 메타데이터의 범주에 속한다.

기술 메타데이터는 보존용 메타데이터(PREMIS) 또는 구조용 메타데이터(METS)에 포함될 수 있다. 기술 메타데이터의 추가와 관련된 결정은 METS와 PREMIS가 자체의 스키마에 기술 메타데이터를 포함시키는 각자의 방법을 가지고 있으므로, 보존을 위한 기술 메타데이터의 유형에 의해 결정되기도 한다.

>> 보존용 메타데이터(PREMIS): *See Also p.72: PREMIS schema 와 METS*

In June 2003, OCLC and RLG jointly sponsored the formation of the PREMIS (***Pre****servation* ***M****etadata: Implementation* ***S****trategies*) working group, comprised of international experts in the use of metadata to support digital preservation activities.

> Metadata structures

메타데이터(메타콘텐트) 또는 보다 정확하게 말해서 메타데이터(메타콘텐트) statements를 모으기 위하여 사용된 어휘들은 잘 정의된 메타데이터 스킴(요강) -메타데이터 표준과 메타데이터 모델을 포함하는 - 을 사용하는 표준화된 개념에 따라 전형적으로 조직화 된다. 통제어휘집, 택소노미, 시소러스, 데이터 사전, 그리고 메타데이터 등록부과 같은 도구들은 추가적인 표준을 메타데이터에 적용하기 위하여 이용될 수 있다. 구조 메타데이터 공통적 속성은 데이터 모델 개발과 데이터베이스 디자인에 있어서 매우 중요하다.

> Metadata syntax

메타데이터 구문은 메타데이터의 필드 또는 요소를 조직하기 위하여 만들어진 규칙을 말한다. 단일 메타데이터 스킴은 서로 다른 구문을 필요로 하는 수많은 다양한 markup 또는 프로그래밍 언어로 표현될 수도 있다. 예를 들어, Dublin Core는 plain text, HTML, XML 그리고 RDF로 표현이 가능하다.

메타콘텐트의 한 가지 일반적인 예는 서지 분류, 주제, DDC 분류번호이다. 이것은 항상 어떤 사물의 “분류”를 나타내는 하나의 암시적 표현이다. 예를 들어, 듀이분류번호 514(형태학)처럼 어떤 사물(다시 말해서 책등에 514라는 숫자를 가지고 있는 책)를 분류하기 위한 그것의 암시적 표현은 “<book><subject heading><514>” 이다.

이것은 주체-술어-사물 트리플이거나 더욱 중요하게 말해서, 부류-속성-값(catagory-attribute-value) 트리플이다. 이 트리플의 첫 번째 2가지 요소인 부류와 속성은 분명하게 정의된 어의를 갖고 있는 어떤 구조 메타데이터의 단편들이며, 3번째 요소는 어떤 참고(마스터) 데이터, 즉 어떤 통제어휘에서 나온 값이다. 메타데이터와 마스터 데이터 요소의 결합은 메타콘텐트인 하나의 문장, 다시 말해서 "metacontent = metadata + master data"을 만들어냈다. 이들 요소 모두는 “어휘(vocabulary)”로 여겨질 수 있다. 메타데이터와 마스터 데이터 둘 다 메타콘텐트에 모여질 수 있는 어휘들이다. 메타와 마스터 데이터 둘 다를 가지고 있는 어휘집에 대한 많은 정보원이 있다.

1) **UML**(Unified Modeling Language (UML):
UML은 소프트웨어 공학 분야에서 사용하는 표준화 및 범용의 모델링 언어이다. 이 언어에는 객체 지향적 소프트웨어 -고급형 시스템의 시각적 모델을 만들기 위한 한 세트의 그래픽 표기 기법이 포함되어 있다.

2) **EDIFACT**(United Nations/Electronic Data Interchange For Administration, International Organization for Standardization (ISO) as the ISO standard ISO 9735).
유엔에서 개발한 국제 EDI 표준이다. 이 표준은 ISO standard ISO 9735로 채택되었다.

3) **XSD**(XML Schema)
XSD는 2001년에 W3C 권고문으로 출판된 XML 스키마이며, 여러 가지 XML 스키마 언어들 중의 하나이다. 이것은 W3C의 권고사항을 충족시키기 위하여 XML용으로 만든 첫 번째 독립된 스키마 언어이다. W3C의 세부규정으로서의 XML 스키마와 일반적으로 스키마 언어를 기술하기 위하여 동일한 용어를 사용함으로써 발생하는 혼란을 피하기 위하여, 몇몇 이용자 집단에서는 이 언어를 WXS(W3C XML Schema의 두문자어)로 언급하고 있는 반면에, 다른 집단에서는 XSD(XML Schema Definition의 두문자어)라고 부르고 있다. 버전 1.1에서 W3C는 XSD라는 용어를 채택하

였다.

4) Dewey/UDC/LoC,

5) SKOS(Simple Knowledge Organization System (SKOS)
이것은 시소러스, 분류표, 택소노미, 주제표목 시스템 또는 어떤 다른 정형화된 통제어휘용으로 디자인된 W3C의 권고사항이다. SKOS는 RDF와 RDFS를 근거로 구축된 시멘틱 웹의 한 부류이며, 이것의 주요 목적은 링크된 데이터처럼 어휘들을 사용하여 쉽게 출판할 수 있도록 하는 것이다.

6) ISO-25964(ISO 25964 is the international standard for thesauri):
아래처럼 2 부분으로 된 시소러스의 국제 표준이다.

Part 1: 정보검색용 시소러스
Part 2: 기타 어휘와의 상호운영성

메타콘텐트 문장의 구성요소용으로 통제어휘를 사용하는 것은, 색인용이든 또는 찾기용이든, ISO-25964에 의해 정해져 있다: " 만일 색인자와 탐색자 둘 다 똑같은 개념으로 똑같은 용어를 선택하도록 유도할 수 있다면, 적절한 문헌이 검색될 것이다." 이것은 인터넷의 공룡인 Google이 텍스트 스트링을 간단하게 색인한 다음에 매칭시키는 것을 고려할 때 특히 적절하다. 따라서 어떠한 지능이나 "간섭"도 필요하거나 발생하지 않는다

7) Pantone: Pantone Inc.
Pantone Matching Systme으로 가장 잘 알려져 있으며, 비록 때때로 유색의 페인트, 직물, 플라스틱의 제조에 사용되더라도, 이 시스템은 여러 산업에서 기본적으로 printing에서 사용하는 독점적인 칼라 공간이다. Pantone Color Matching System은 거의가 표준화된 칼라 재생 시스템이다. 칼라를 표준화함으로써, 서로 다른 위치에서 서로 다른 제조업자들 모두 서로 간의 직접적인 접촉없이 칼라의 매치를 확실하게하기 위하여 Pantone system을 사용한다.

8) Linnaean Binomial Nomenclature(Binomial nomenclature (also called binominal nomenclature or binary nomenclature) ; 이항식 명명법은 비록 다른 언어로 된 단어를 근거로 하더라도 각각의 생물 종에게 두 부분으로 구성된 이름을 둘 다 라틴어의 문법을 사용함으로써, 그것의 이름을 붙이는 공식적 시스템이다. 그러한 이름을 binomial name(줄여서 "binomial"), binomen 또는 scientific name이라 하며, 보다 비공식적으로는 그것을 라틴이름으로 부르는 것이다. 이 이름의 첫 번째 부분은 그 종이 속한 類를 구분하는 것이며, 두 번째는 類 안에 있는 種을 구분하는 것이다. 예) 인간은 Homo 류에 속하며, 그 류 속에 있는 Homo sapiens에 속한다.

> Metadata standards

핵심 표준은 ISO/IEC 11179-1:2004와 후속 표준(see ISO/IEC 11179)이다. 이 표준으로 인하여 기존에 이미 출판된 모든 사물들은 단지 메타데이터의 정의만을 다루고 있으며, 어떤 행정적 표준과 마찬가지로 메타데이터의 저장 또는 검색의 구조를 지원하지 못하고 있다. 중요한 것은 이 표준이 메타데이터를 데이터의 콘테이너에 대한 데이터로 언급하고 있지 데이터의 콘텐트에 대한 데이터로 메타데이터를 언급하지는 않고 있다는 것이다.

Dublin Core 메타데이터 용어들은 한 무리의 어휘 용어들이며, 찾기용의 자원을 기술하는데 사용될 수 있다. Dublin Core Metadata Element로 알려진 15개의 전통적인 메타데이터 용어들의 초기 세트는 다음과 같은 표준 문서들에 의해 보증을 받았다:

IETF RFC 5013
ISO Standard 15836-2009
NISO Standard Z39.85

비록 표준은 아니지만, microformat(sometimes abbreviated μF)는 semantic markup을 위한 웹 의존형 시도이며, 이것은 웹 페이지에 있는 메타데이터와 기타 속성, 그리고 RSS처럼 (X)HTML을 지원하는 기타 contexts를 전달하기 위하여 기존의 HTML/XHTML 태그를 재사용한다. 이 시도는 소프트웨어로 하여금 최종이용자가 의도한대로 contact information, geographic coordinates, calendar events, and similar information와 같은 정보들을 자동적으로 처리하도록 한다. microformat은 XHTML과 HTML 기준을 따르지만 그 자체가 표준은 아니다. 비록 웹 페이지의 콘텐트가 기술적으로 이미 웹의 초기부터 자동처리 능력을 갖추었다하더라도, 그 같은 처리는 어려운데 왜냐하면 정보를 디스플레이하기 위하여 사용되는 전통적인 markup tags는 그 정보가 무엇을 의미하는지를 기술하지 못하기 때문이다. microformats는 어의를 첨부함으로써 이러한 문제를 해결함으로써, natural language processing or screen scraping과 같이 더욱 복잡한 자동처리방법을 피할 수 있게 되었다. microformats의 사용, 적용과 처리를 통하여 데이터 아이템을 색인하고, 탐색하고, 저장하거나 상호참조할 수 있도록 정보를 재사용하거나 결합할 수 있다.

> Library and information science and metadata

도서관은 가장 일반적으로 통합도서관관리시스템(ILMS)의 한 부분으로 도서관 목록에 메타데이터를 사용하고 있다. 메타데이터는 책, 정간물, DVD, 웹 페이지 또는 디지털 이미지와 같은 자원을 편목 하는데서 얻어진다. 이러한 데이터는 MARC 메타데이터 표준을 사용하는 통합도서관관리시스템인 ILMS에 저장된다. 이것의 목적은 이용자에게 그들이 요구하는 아이템이나 지역의 물리적 또는 전자적 위치로 직접 유도해 줄 뿐만 아니라 궁금해 하는 아이템에 대한 설명을 제공하는 것이다.

도서관 메타데이터에 대한 보다 최신의 그리고 전문화된 경우가 e-print repositories와 digital image libraries와 같은 디지털 도서관에서 나타나고 있다. 도서관 원칙에 따르면서, 종종 메타데이터를 제공하는데 있어서 비-도서관적인 방법을 사용하는 경우가 있는데, 이것은 도서관들이 반드시 전통적이거나 일반적인 편목방법을 따르지 않는다는 것을 의미한다. 자료의 관습적 속성, 예를 들어 분류 분야, 위치 분야, 키워드 또는 저작권 문장에 대한 메타데이터가 가끔 특별하게 만들어지지만, 일반적으로 파일의 크기와 포맷과 같은 표준 파일 정보는 자동적으로 여기에 포함된다.

도서관 운영을 위한 표준은 수십 년 동안 ISO의 중요한 논제이다. 디지털 도서관의 메타데이터 표준에는 Dublin Core, METS, MODS, DDI, ISO standard Digital Object Identifier (DOI), ISO standard Uniform Resource Name (URN), PREMIS schema, Ecological Metadata Language, 그리고 OAI-PMH가 있다. 세상을 선도하는 도서관들은 자신들의 메타데이터 표준 전략에 대하여 다양한 힌트를 제공하고 있다.

*** Dublin Core**

See p.72

* METS

METS는 W3C의 XML 스키마를 사용하여 표현된 디지털 도서관의 객체에 대한 기술, 행정 및 구조 메타데이터를 암호화하기 위한 메타데이터 표준이다. 이 표준은 the Network Development and MARC Standards Office of the Library of Congress에 의해 유지 관리되며, the Digital Library Federation의 주도하에 개발 중이다.

METS는 다음과 같은 목적용으로 디자인된 XML 스키마이다:

(1) 디지털 도서관 사물의 계층적 구조를 표현하는 XML 도큐멘트 경우(instances)의 제작.
(2) 그 같은 사물을 구성하는 필드들의 이름과 위치의 기록.
(3) 관련된 메타데이터의 기록하기. 그러므로 METS는 특별한 다큐멘트 유형과 같은 실세계의 사물을 모델화하는 도구로 사용될 수 있다.

이러한 용도에 따라서, METS 다큐멘트는 Open Archival Information System Reference Model의 Submission Information Package (SIP), Archival Information Package (AIP), or Dissemination Information Package (DIP)의 역할을 담당할 수 있다.

* OAIS: An Open Archival Information System

OAIS란 정보를 보존하여 어떤 특정한 커뮤니티용으로 이용시킬 책임이 있는 사람과 시스템의 조직으로 구성된 archive 이다. 디지털 정보를 장기간 보존하는 데 필요한 시스템, 즉, 아카이브를 위한 개념적 구조틀을 마련한 ISO 표준(ISO 14721)이다. ISO의 요청으로 미국 항공 우주국(NASA)의 CCSDS(Consultative Committee for Space Data Systems)가 주체가 되어 개발했다.

1999년 초안이 발표된 후, 국제적 · 다학문적 의견 수렴 과정을 거쳐 2002년 국제 표준으로 확정되어, 정보 모형, 정보 패키지 모형, 아카이브 기능 모형 등 디지털 아카이빙과 관련된 기본적인 개념틀을 정의하였다. 현재 디지털 아카이빙과 관련된 거의 모든 실험과 프로젝트가 이 표준을 기반으로 진행되고 있을 정도로 커다란 의미를 갖고 있다. OAIS 참조 모형의 서문에서 이 문건을 “아카이브가 디지털 정보를 영구 혹은 무기한 장기간 보존하는 데 있어서 광범위한 의견 일치에 도달하기 위해 개발된 기술적 권고안”이라고 소개하고 있다. 광범위한 의견 일치란 디지털 정보를 장기간 보존하는 활동을 수행하는 모든 기관들 사이의 의사소통 기반을 마련해 앞으로의 협력을 활성화시킨다는 의도를 요약한 것이다. 따라서 이 참조 모형의 가장 즉각적인 의미는 수년에 걸친 개발과 의견 수렴 과정을 통해 디지털 정보 보존에 관한 기본 개념과 용어에 대한 의견 일치를 도출해낸 데 있다. OAIS 참조 모형은 정부 기관, 도서관, 아카이브즈, 그리고 기업체나 대학 등 디지털 정보를 보존하여 이용하는 모든 기관, 심지어는 현재로서는 스스로가 아카이빙 기능을 수행하고 있다고 믿지 않는 기관들까지 그 적용 대상으로 포함시키고 있다.

> The reference model:

(1) 장기간의 디지털 정보 보존과 접근에 필요한 문서 개념에 대한 이해력과 자각심의 기본틀을 제공한다.

(2) 보존과정에서 효과적인 참여자가 되기 위하여 비-문서적 기관에 의해 요구되는 개념을 제공한다.

(3) 현재와 과거의 문서관의 운영과 구조를 비교하고 설명하기 위한 용어와 개념에 대한 기본틀을 제공한다.

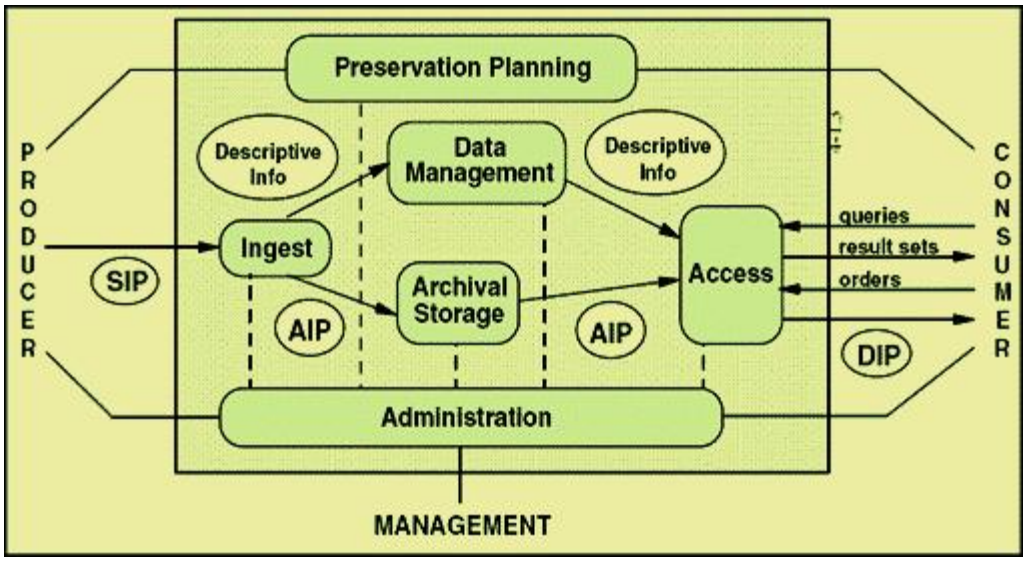
(4) 서로 다른 장기 보존 전략과 기술을 비교하고 설명하기 위한 기본틀을 제공한다.

(5) 문서관에 의해 보존된 디지털 정보의 데이터 모델을 비교하고 데이터 모델과 그것의 주요 정보가 시간이 지나면서 어떻게 변하는지를 논의하기 위한 근거를 제공한다.

(6) 비-디지털 형태(예, 물리적 매체와 물리적 표본) 정보의 장기보존을 다루기 위하여 다른 연구에 의해 확장될 수도 있는 토대를 제공한다.

(7) 장기적 디지털 정보의 보존과 접근을 위한 요소와 과정에 대한 일치된 의견을 확대시키고, 기업이 지원할 수 있는 보다 큰 시장을 육성시킨다.

(8) OAIS-관련 표준의 확인과 제정을 안내한다.



OAIS Functional Entities

>> Submission Information Package (SIP)

>> Archival Information Package (AIP)

>> Dissemination Information Package (DIP)

*** BagIt**

BagIt는 임의적인 디지털 콘텐트의 네트워크 전달과 디스크형 저장을 지원하기 위한 계층

형 파일 패키징 포맷이다. 하나의 “bag(임의적 콘텐트)”은 bag의 저장과 전송을 다큐멘트하려는 목적을 가진 메타데이터 파일들인 “payload”와 “tags”로 구성된다. 필요한 태그 파일에는 checksum을 갖고 있는 payload에 있는 모든 파일을 리스트하고 있는 manifest(적하목록, 승객명단)가 포함된다. BagIt이란 이름은 때때로 “bat it and tag it”으로 말하기도 하는 “enclose and deposit” 방법에 의해 유래하였다.

Bags은 하나의 파일집단처럼 정상적으로 보관된 디지털 콘텐트용으로는 이상적이다. 이것들은 또한 문서보관용 데이터베이스에 정상적으로 보관된 콘텐트를 export하는데 매우 적합하다. cross-platform (Windows and Unix) file system naming conventions에 따라, bag의 payload에는 어느 정도의 디렉토리들과 하위 디렉토리들(폴더와 하위 폴더)이 포함될 수 있다. bag은 그 bag을 완성하기 위하여 그 네트워크의 밖에서 가져올 수도 있는 콘텐트용의 URL들을 리스트하고 있는 “fetch.txt”파일을 통하여 간접적으로 payload content를 검사할 수 있다: 간단한 paralleization(평행, 비교) - 예를 들어, Wget의 연속적인 10가지 경우들 - 은 커다란 bags를 매우 빠르게 전송하기 위하여 이러한 기능을 활용하고 있다. bags의 장점은 다음과 같다:

1) 디지털 도서관에서 널리 채택하고 있다(예, 미 의회 도서관)
2) 일반적으로 잘 알려진 파일 시스템 도구를 사용하여 설치하기가 용이하다.
3) 파일처럼 취급받는 콘텐트는 단지 payload 디렉토리에서만 복사될 수 있다.
4) ML wrapping과 비교해서, 콘텐트는 시간과 저장 공간을 절약하기 위하여 암화화될 필요가 없다.
5) 접수된 콘텐트는 친숙한 파일시스템 트리로 쉽게 갈 수 있다.
6) 병행적으로 일반적인 전이 도구를 기동시킴으로써 신속한 네트워크 전이가 용이하다.

1) In computing, **cross-platform**, or multi-platform,
복수의 컴퓨터 프랫폼에 설치되어 사로 운영할 수 있는 컴퓨터 소프트웨어나 컴퓨팅 방법과 개념에서 사용되는 속성을 말한다.

2) **GNU Wget** (or just Wget, formerly Geturl)
웹 서버로부터 콘텐트를 검색하는 컴퓨터 프로그램이며, GNU Project의 일부이다. 이것의 이름은 World Wide Web and get로부터 초래되었다.

***MODS**

MODS는 XML 기반의 서지 기술 구조형식이며, 미 의회에서 개발하였다. MODS는 도서관에서 사용하는 MARC 포맷의 복잡성과 Dublin Core 메타데이터의 지나친 단순성 간의 타협을 위해 디자인되었다. MODS 레코드는 MARC 레코드로부터 key data elements를 전달하도록 디자인되었지만, 모든 MARC 필드들을 정의하지는 않으며 MARC 표준으로부터 tag하고 있는 필드나 하위필드를 사용하지도 않는다. MODS에는 MARC 레코드와 호환할 수 없는 데이터 요소들이 존재하기 때문에, MARC에서부터 MODS로 변환시킬 때, 또는 그 반대의 경우에 어느 정도의 손실이 발생한다. MODS가 이용자에게 아무리 편리하다고 하더라도 이 두 가지 메타데이터 포맷의 호환성을 유지하는데 미 의회도서관은 어떠한 책임도 지지 않는다.

MODS의 사용은 다른 메타데이터 구조식과 비교하여 여러 가지 장점을 제공 한다:

1) 기존의 자원 기술들과 높은 호환성을 갖는다.
2) 다양한 내적 레코드 요소 세트들을 MODS용으로 작성할 수 있도록 MARC보다 세부 내용이 덜 까다롭다.
3) DC로부터 작성된 아이템 기술과 기타 보다 간단한 포맷을 사용하여 개선될 수 있다.

*** DDI:** The Data Documentation Initiative (DDI)

DDI는 통계 및 사회과학 데이터를 기술하기 위한 정보 표준 작성의 국제적 프로젝트이다. 1995년에 시작하였으며, 전세계의 데이터 전문가들이 참여하였다. XML로 쓰여진 DDI 규칙은 정보의 콘텐트, 교환, 보존용의 포맷을 제공하고 있다.

*** ISO standard Digital Object Identifier (DOI)**

See p.85

*** ISO standard Uniform Resource Name (URN)**

URN이란 the urn:scheme을 사용하는 URI의 오래된 역사적 이름이다. RFC 2141에서 1997년에 정의된, URNs는 단일 URN namespace에 간단하게 namespace의 mapping이 가능하도록 함으로써, 자원들에 대한 항구적이고 위치-독립적인 식별자로서 활동하도록 만들어졌다. 이 같은 URI가 있다는 것은 식별된 자원의 이용가능성을 의미하는 것이 아니라, 자원들이 사라지거나 이용불가능하게 될 때조차도 그 URIs는 전 세계적으로 유일하고도 항구적으로 남아있어야 한다는 것이다.

1) A Request for Comments (RFC)

인터넷과 관련된 중요한 기술개발과 표준을 설정하는 the Internet Engineering Task Force (IETF)와 the Internet Society의 출판물이다.

2005년에 RFC 3986 이래로, 이 용어의 사용은 W3C와 IETF의 공동실무진에 의해 제안된 개념이지만, 엄격성이 다소 떨어진 “URI”에 대한 선호로 인하여 줄어들었다.

URNs(이름)과 URLs(위치명) 둘 다는 URIs이며 하나의 특별한 URI는 동시에 이름이면서 위치이름일 수도 있다. URNs은 원리 1990년대에 하나의 메타데이터 기본틀로 URLs과 URCs와 함께 인터넷의 3-부분 정보구조의 일부로 만들어졌다. 그렇지만, URCs는 결코 과거의 개념적 단계를 벗어나지 못했으며, RDF와 같은 다른 기술이 후에 그 자리를 차지하였다.

> URC: a uniform resource characteristic (URC)

URC는 일련의 문자열이며 URI(웹 자원을 식별할 수 있는 문자열)의 메타데이터를 표현한

다. URC는 URI의 조합적 URN(웹 자원의 유일한 이름)을 그것의 URL(웹 자원을 찾을 수 있는 위치)에 결합시킨 것이다.

*** PREMIS: PREservation Metadata: Implementation Strategies**

PREMIS는 디지털 보존을 위해 사용할 수 있는 메타데이터의 개발을 위한 국제실무집단이다. 2003년에 OCLC와 RLG는 PREMIS 실무그룹을 만들었으며, 관리와 사용을 위한 가이드라인 및 권고문과 더불어 실행 가능하고 핵심적인 보존용 메타데이터를 정의하기 위하여 이들은 문화, 정부, 사설 기관에서 온 30명 이상의 대표들로 된 다국적 멤버들로 구성하였다. PREMIS는 설치에 있어서 독립적이고 실무 지향적이며, 대부분의 보존기관에서 필요로 하는 한 무리의 어의적 요소를 정의하고 있다.

2005년 5월에, PREMIS는 Data Dictionary for Preservation Metadata: Final Report of the PREMIS Working Group를 해제하였다. 이 237 페이지짜리 보고서에는 다음과 같은 것들이 포함되어 있다:

- ► PREMIS Data Dictionary 1.0- 디지털 아카이빙 시스템에서 보존 메타데이터를 실행하기 위한 포괄적이고 실재적인 자원;
- ► 딸림 보고서 - context, data model, assumptions을 제공;
- ► 특별한 논제, 용어, 사용의 예제들;
- ► Data Dictionary의 사용을 지원하기 위하여 개발된 XML 구조식의 세트.

PREMIS 2.0 번전은 2008년 3월에 해제되었다.

> Entities

PREMIS 데이터 모델은 5개의 상호 연관된 객체들로 이루어져 있다:

- ► Intellectual entity:
- ► Object & Event entities:
- ► Agent:,
- ► Rights - 위의 4가지 분야 중 하나에 이미 짜여진(mapped) 각각의 어의적 개체에 대한 권리.

intellectual entity는 책이나 데이터베이스처럼 독립적이고 서로 연결되어 있는 지적 unit로 구성되어 있는 한 무리의 콘텐트이다. 이것들은 다른 지적 객체를 포함하는 복합적 객체일 수도 있으며, 복수의 디지털 표현을 가질 수도 있다. descriptive metadata는 대체로 이 단계에 적용된다. 이것들은 다른 지적 객체를 포함하는 복합적 객체일 수도 있으며, 복수의 디지털 표현을 가질 수도 있다.

데이터 사전에 나열되어 있는 대부분의 어의적 units는 사물과 사건에 연관되어 있다. object entity는 추가로 3가지의 하위 유형인 file, bitstream, representation 으로 세분된

다. 파일은 대부분의 최종 이용자가 “운영체계에 의해 밝혀진 바이트들의 이름이 붙이고 순서를 정하는 작업을 하는 단계이다. 여기에는 운영체계에 의해 이해될 수 있고, 보존 목적을 위하여 의미있는 공동의 성질을 가지고 있는 파일 속에 들어있는 연속 또는 비연속 데이터인 bitstreams을 품고 있는 다양한 파일 시스템 속성들이 포함된다. 묘사(representation)는 어떤 의미에서 이 모델의 가장 높은 단계이다. 왜냐하면 이것은 지적 객체의 구조와 콘텐트를 적합하게 제공하기 위하여 여러 가지 파일을 포함할 수 있기 때문이다.

모든 레포지토리들이 객체의 디지털적인 “intrinsic value(본질적 가치)”로 여겨질 수 있는 것을 보존하기 위하여 자신들의 목적과 the curatorial body의 요구를 근거로 묘사를 보존하는 것과 관련이 있는 것은 아니다. 더구나 지적 객체는 한 레포지토리 안에 다수의 묘사를 가질 수도 있다.

events는 사물과 결합된 agents(“events와 결합되어 있는 사람, 기관, 또는 소프트웨어 또는 한 객체에 붙어있는 Rights)와 그들에게 영향을 끼치고 있는 행동과 관련이 있는 사물들과 상호 연관되어 있다.

최종적으로 right entity는 저작권과 계약권의 법률적 필요조건에 대한 관심과 자성을 높이는 것과 관련되어 있다. 또한 여기에는 허가된 특별한 행동들에 대한 정보가 포함되기도 한다.

> Data dictionary

PREMIS 데이터 사전 항목들에는 12가지의 속성분야가 포함되어 있다. 이것들 모두가 semantic unit(다른 메타데이터에 있는 “element(요소)”와 비슷한)에 응용되는 것은 아니며, 개체의 이름 그리고 정의와 더불어, 해당 필드가 객체용으로 포함되어져야 하는 이론적 근거, 용도 설명서, 그리고 값을 채우는 방법과 같은 것들이 기록된다. 이 속성들 중에서 4가지 – 객체범주, 적용성, 반복성, 그리고 책임성 – 는 서로 연관되어 있으며, 마지막 3가지는 각각의 file, bitstream, representation인 객체의 수준에 따라 정의된다. 그리고 이 사전은 계층적이다: 어떤 어의적 개체들이 다른 것 속에 포함되기도 한다.

*** Ecological Metadata Language**

EML은 생태학 분야에서 개발된 메타데이터 표준이다. 이것은 the Knowledge Network for Biocomplexity를 포함하여 the Ecological Society of America 등에 의해 수행된 이전의 연구를 근거로 삼고 있다. EML은 메타데이터의 구조적 표현용으로 사용 가능한 한 무리의 XML 구조식 다큐멘트이다. 이것은 생태학의 연구자로 하여금 전형적인 데이터 세트를 도큐멘테이션 할 수 있도록 특별히 개발되었다. EML은 주로 디지털 자원을 기술하도록 설계되었지만, 그것은 또한 종이 지도와 기타 비디지털 매체와 같은 비디지털 자원을 기술하는데도 사용할 수 있다.

*** OAI-PMH: Open Archives Initiative Protocol for Metadata Harvesting**

the Open Archives Initiative에 의해 개발된 프로토콜이다. 이것은 많은 아카이브즈들로부터 메타데이터를 이용하여 서비스가 이루어지도록, 아카이브에 있는 레코드의 메타데이터 기술(description)을 수집하는데 이용된다. OAI-PMH의 실행은 Dublin Core에 있는 메타데이터의 표현을 지원하면서, 또한 추가적인 표현도 지원하기도 한다. 이 프로토콜은 일반적으로 OAI Protocol라 말하며, HTTP보다는 XML을 사용한다.

*** Z39.50**

Z39.50은 원거리 컴퓨터 데이터베이스로부터 정보를 찾거나 검색하기 위하여 사용되는 클라이언트-서버 프로토콜이다. 이것은 ANSI/NISO standard Z39.50 이면서 ISO standard 23950 이다. 이 표준의 관리기관은 미 의회도서관이다. Z39.50은 도서관 환경에서 널리 사용되고 있으며, 종종 통합도서관시스템 및 인문서지 참고 소프트웨어에서 사용되기도 한다. 그리고 도서관 상호대차를 위한 상호대차목록 탐색 역시 종종 Z39.50 쿼리로 이루어지기도 한다.

Contextual Query Language (formerly called the Common Query Language)는 Z39.50 어의론에 근거하고 있다. Z39.50은 웹 이전의 기술이며, 다양한 실무 그룹이 오늘날의 환경에 보다 잘 적응하도록 갱신을 시도하고 있다. 이러한 시도는 the designation ZING(Z39.50 국제판)에서 나타나고 있으며 현재 다양한 전략이 추구되고 있다.

1) **CQL: Contextual Query Language**
탐색엔진, 서지카탈로그 그리고 박물관 소장정보와 같은 정보검색시스템에서 쿼리를 표현하는 공식적 언어이다. Z39.50의 어의론을 근거로, 이것의 목적은 사람이 읽고 쓸 수 있도록 쿼리하는 것이며, 또한 보다 복잡한 쿼리언어의 표현이 가능한 직관적 언어로 디자인하는 것이다.

가장 중요한 것은 쌍둥이 프로토콜들인 SRU/SRW(***See next page)***이다. 이것들은 Z39.50 통신 프로토콜(HTTP로 그것을 대체한)에 속하지만, 쿼리 구문의 장점을 보존하려고 한다. SRU는 REST을 기반으로 하고 있으며, 쿼리를 URL 쿼리 스트링으로 표현할 수 있다: SRW는 SOAP를 사용하고 있다. 둘 다 탐색결가를 XML로 바꿀 수 있다.

이러한 프로젝트들은 상대적으로 시장이 작은 도서관 소프트웨어를 가지고 훨씬 커다란 시장용으로 개발된 웹 서비스 도구로부터 이익을 얻을 수 있도록 함으로써, 최초의 Z39.50 프로토콜보다 개발자들이 이런 프로젝트에 참여하기가 훨씬 좋아졌다.

이것의 대안들은 다음과 같다:

- ► Search/Retrieve Web Service, successor to Z39.50
- ► Open Archives Initiative Protocol for Metadata Harvesting
- ► SPARQL

1) **SPARQL** (pronounced "sparkle", a recursive acronym for SPARQL Protocol and RDF Query Language) 데이터베이스용 쿼리 언어이며, Resource Description Framework format.에 저장된 데이터를 검색하고 조작할 수 있는 RDF query language 이다.

*** UDDI: Universal Description Discovery and Integration**

UDDI(pronounced /ˈjʊdiː/)는 전 세계의 기업들이 인터넷 상에 스스로 리스트를 올릴 수 있는 프랫폼-독립적이고 XML-의존형인 레지스트리이며, 웹 서비스 어플을 등록하고 그 위치를 설정할 수 있는 메카니즘이다. UDDI는 the Organization for the Advancement of Structured Information Standards (OASIS)에 의해 후원을 받는 open industry initiative 이며, 그 목적은 기업들로 하여금 서비스 listings를 출판하여, 서로를 발견할 수 있고, 그리고 그 서비스와 소프트웨어 어플이 인터넷에서 상호작용하는 방법을 정의하기 위한 것이다.

UDDI는 원래는 하나의 핵심적인 웹 서비스 표준으로 제안되었다. 이것은 SOAP 메시지로 interrogate(질문)하고, 그것의 디렉토리에 열거되어 있는 웹 서비스들과 상호작용하는데 필요한 protocol bindings and message formats를 기술하고 있는 Web Services Description Language (WSDL) document로의 접근하도록 설계되었다.

*** SRU: Search/Retrieval via URl**

Search/Retrieve via URL (SRU)는 인터넷 탐색 쿼리용으로 만든 표준 탐색 프로토콜이며, 쿼리를 표현하기 위한 표준 쿼리 구문으로 Contextual Query Language (CQL)를 사용하고 있다. Applications: Image search, Video search engine, Enterprise search, Semantic search.

*** SRW: Search/Retrieve Web service**

이것은 탐색 및 검색용 웹 서비스이다. SRW는 쿼리용이며, 그것의 동반자적인 프로토콜 SRU에 의해 제공된 URL 인터페이스가 더욱 활성화될 수 있도록 SOAP 인터페이스를 제공하고 있다. SRU와 SRW에서의 쿼리들은 CQL을 사용하여 표현되며, SRW, SRU, and CQL의 표준은 모두 미 의회도서관에서 공표하고 있다.

*** Representational State Transfer (REST)**

SOAP와 WSDL에 근거한 웹 서비스보다 더욱 간단한 대안으로 웹에서 널리 사용되고 있으며, 웹 서비스의 제작을 위한 가이드라인으로 사용되는 architecture styles 또는 design pattern 이다. REST는 탁월한 웹 API(응용 어플 프로그램 - 서로 통신하기 위하여 소프트웨어 구성요소들에 의해 하나의 인터페이스처럼 사용될 목적으로 만든 프로토콜)처럼 등장하였다. API란 routines, data structures, object classes, and variables와 같은 design model용의 스펙을 포함하기도 하는 하나의 library 이다.

REST는 서로 다른 서비스간의 헐거운 이중성을 허용함으로써, 웹 서버들 간의 교신을 용이하게 한다. REST는 그것의 상대방인 SOAP보다는 형식에 있어서 다소 강하지 못하다. REST 언어는 명사와 동사로 사용되며 가독성을 강조하고 있다. SOAP와 달리, REST는 XML

검사가 필요하지 않으며, 서비스 제공자로부터 또는 그것으로 전달되는 메시지 헤더를 필요로 하지 않는다. 이것은 궁극적으로 적은 bandwidth를 사용하며, 에러 처리 또한 SOAP에서 사용되는 것과는 차이가 있다.

1) **bandwidth, network bandwidth, data bandwidth, or digital bandwidth**
bits per second 또는 multiples of it (bit/s, kbit/s, Mbit/s, Gbit/s, etc.)로 표현되며, 이용가능하거나 소비된 데이터 커뮤니케이션 자원의 bit-rate 척도이다.

* **SOAP: Simple Object Access Protocol**

컴퓨터 네트워크에서 웹 서비스를 실행하는데 정형화된 정보를 교환하기 위한 프로토콜 스펙이다. 이것은 메시지 포맷용으로 XML을 사용하고 있으며, 보통 다른 Application Layer protocol - 가장 유명한 것은 HTTP나 SMTP(Simple Mail Transfer Protocol) - 에 의존하고 있다.

1) the application layer
인터넷 모델에서 어플 레이어란 Internet Protocol (IP) computer network에서 process-to-process communications용으로 디자인된 커뮤니케이션 프로토콜과 방법을 위해 예약된 abstraction layer 이다. 이 프로토콜은 ports를 통해 process-to-process connections을 수립하는데 있어서 기초가 되는 transport layer protocols을 사용한다.

OSI 모델에서, 이것의 어플 레이어의 정의는 범위가 보다 협소하다. OSI 모델에서는 어플 레이어를 user interface로 정의하고 있으며, OSI 어플 레이어는 인간이 인지할 수 있는 포맷으로 이용자에게 데이터와 이미지를 보여주는데, 그리고 그 밑에 있는 presentation layer와 인터페이스하는데 책임을 갖는다.

* **SPARQL** (pronounced "sparkle", a recursive acronym for SPARQL Protocol and RDF Query Language)

SPARQL은 RDF 쿼리 언어, 즉 데이터베이스 언어로서 RDF 포맷에 저장된 데이터를 검색하고 처리라는데 사용한다. 이것은 the RDF Data Access Working Group (DAWG) of the World Wide Web Consortium의 표준이며, 시멘틱 웹의 주요 기술 중의 하나로 인식되고 있다. SPARQL은 triple patterns, conjunctions, disjunctions, 그리고 optional patterns으로 구성된 쿼리를 사용하며, 또한 복수의 프로그래밍 언어용으로 설치가 가능하다.

1) **triplestore**
트리플의 저장과 검색을 목적으로 구축된 데이터베이스이다(triple이란 subject-predicate-object, like "Bob is 35" or "Bob knows Fred"처럼 구성된 데이터 엔티티 이다).

관계형 데이터베이스처럼 많은 것이 트리플스토어에 정보를 저장하며 쿼리언어를 통해 검색한다. 관계형 데이터베이스와의 차이는 트리플스토어는 트리플의 저장과 검색용으로 최적화된다는 것이다. 쿼리에 따라서, 트리플은 대체로 RDF나 기타 포맷을 사용하여 imported/exported 될 수 있다.

2) **conjunction(and)**
논리학이나 수학에서 이항논리연산자는 and는 논리적 conjuction으로 알려져 있으며, 만일 그것의 양쪽 피연산자(operands)가 참이면 그 결과의 값이 참이고, 그렇지 않다면 부(false)의 값을 갖는다.

3) **disjunction (or)**
논리학이나 수학에서, or는 disjunction과 alternatio으로 알려진 a truth-functional operator 이다. 이 연산자를

표현하는 논리적 connective는 “or”로 알려져 있으며, 전형적으로 or라고 쓴다. or 연산자는 그것의 피연산자 하나 이상이 참일 때마다 참의 값을 생산한다. 예를 들어, A가 참이거나 B가 참일 때, 또는 A와 B 모두 참일 때, “A or B”는 참이다.

*** WSDL: Web Service Description Language**

WSDL은 XML-기반 인터페이스 기술 언어이며, 웹 서비스에서 제공하는 기능성을 기술하기 위하여 사용된다. 웹 서비스의 WSDL description(WSDL 파일이라고도 함)에서는 서비스의 호칭는 방법, 예상되는 매개변수, 반환할 데이터의 구조 형태에 대한 기계가독형 기술을 제공한다. 이것의 2.0 버전에서는 두문자 D가 Definition으로 바뀌었다.

*** CGI: Common Gateway Interface**

CGI란 웹 페이지와 웹 어플에서 역동적인 콘텐트를 생산하는데 사용하는 표준 방법이다. 웹 서버를 실행할 때, CGI는 웹 서버와 그것의 웹 콘텐트를 생산하는 프로그램 사이에 인터페이스를 제공한다. 이러한 프로그램들은 CGI scripts 또는 단순히 CGIs라 부르며, 보통 scripting language로 작성된다.

1) **scripting language or script language**
사람에 의해 하나씩 대안적으로 수행되는 업무를 자동적으로 해석하여 수행하는 특별한 run-time 환경용으로 작성된 프로그램인 scripts를 지원하는 프로그래밍 언어이다.

p. lxvii

*** Open URL**

OpenURL이란 인터넷 이용자가 접근 가능한 자원을 보다 쉽게 찾을 수 있도록 하는 URL의 표준화 포맷이다. 비록 OpenURL이 인터넷에서 다양한 종류의 자원에 사용될 수 있다 하더라도, 이용자를 구독예약 콘텐트에 연결시켜주는 도서관에서 가장 많이 사용하고 있다.

OpenURL 표준은 초록 및 색인 데이터베이스(정보원)와 같은 정보자원으로부터 온라인으로나 인쇄물로나 또는 다른 포맷으로 학술지와 같은 도서관 서비스(목표물)로의 링크가 가능하도록 디자인되었다. 이런 링킹은 OpenURL의 요소들을 조사하고 OpenURL 지식 베이스를 이용함으로써, 도서관을 통해 이용 가능한 올바른 목표물로 링크를 제공하는 “link resolvers” 또는 “link-servers”에 의해 이루어진다.

OpenURL을 생산하는 정보원은 전형적으로 학술 기사, 책, 특허 등과 같이 도서관에서 종종 발견되는 정보자원을 색인한 데이터베이스의 서지 인용이나 서지 레코드이다. 이러한 데이터베이스의 예로는 Ovid, Web of Science, SciFinder, Modern Languages Association

Bibliography, Google Scholar가 있다.

1) Ovid Technologies, Inc. (or just Ovid for short),
online bibliographic databases, academic journals, and other products, chiefly in the area of health sciences의 접근을 제공하는 the Wolters Kluwer group의 한 회사이며, The National Library of Medicine's MEDLINE database가 이 회사의 주요 제품이고, 현재 이 데이터베이스는 무료로 PubMed을 통해 이용할 수 있다.

2) Web of Science (WoS)
an online subscription-based scientific citation indexing service 이며, 포괄적으로 인용문 탐색을 제공하는 Thomson Reuters에 의해 관리되고 있다. 또한 학술 및 과학 분야에 있는 전문화된 하위 분야에 대한 심오한 탐색을 가능하게 하는 cross-disciplinary research를 참조하는 multiple databases에 대한 접근을 제공한다.

3) CAS 데이터베이스는 두 가지의 중요한 DB 시스템을 통해 이용할 수 있다: **STN** & **SciFinder**.

 # STN
STN (Scientific & Technical Information Network) International은 CAS 와 FIZ Karlsruhe가 공동으로 운영하고 있으며, command language interface를 사용함으로써 주로 정보전문가용이다. CAS databases와 더불어, STN 또한 Dialog와 같은 많은 다른 데이터베이스에 대한 접근도 제공하고 있다.

 # SciFinder
SciFinder는 화학 및 서지 정보 데이터베이스이다. 웹 버전인 원래의 클라이언트 어플은 2008년에 원래의 클라이언트 어플의 해금되었다. 이것은 graphics interface를 갖고 있으며 화화구조를 탐색할 수도 있다.

 # CASSI
CASSI란 Chemical Abstracts Service Source Index이다. 이것의 이전 인쇄판 데이터베이스는 지금 무료 온라인 정보원이며 출판 정보를 확인하고 찾는데 사용된다. CASSI는 선택된 저널의 titles and abbreviations, CODEN, ISSN, publisher, and date of first issue (history)를 제공하며, 또한 요약의 텍스트와 언어에 대한 정보도 제공한다. 이것의 범위는 1907년부터 현재까지이며, 과학기술분야의 연속 그리고 비연속 간행물을 포함하고 있다.

4) The Modern Language Association of America (referred to as the Modern Language Association or MLA)
언어 및 문학 학자들을 위한 미국의 중요한 전문학회이며, 이것의 목적은 언어와 문학의 강의와 연구를 강화시키는 것(strengthen the study and teaching of language and literature) 이다.

5) **Google Scholar**
학술문헌의 full text를 색인하고 있는 a freely accessible web search engine 이다. the Google Scholar index 은 most peer-reviewed online journals of Europe and America's largest scholarly publishers을 포함하고 있으며, 또한 scholarly books and other non-peer reviewed journals도 포함하고 있다. 또한 이것은 무료인 Scirus(from Elsevier, CiteSeerX, and getCITED)와 비슷한 기능을 가지고 있으며, 이것은 Elsevier's Scopus and Thomson ISI's Web of Science와 비슷한 예약-의존형 도구이다.

여기서 목표물이란 자원이나 서비스이며 리것은 이용자의 정보 요구를 만족시키는 것을 돕는다. 이러한 목표물의 예로는 full-text repositories, online journals, online library catalogs 그리고 기타 Web resources와 services가 있다.

NISO는 ANSI 표준 Z39.88로서 OpenURL과 그것의 데이터 콘테이너(the Context Object)를 개발하였으며, 2006년 6월 22일에 OCLC가 이 표준의 유지관리기관으로 지명되었다.

> Use

OpenURL의 가장 일반적인 용도는 웹 자원(온라인 학술기사와 같은)의 요청과 관련된 해결책을 제시하는 것이다. 하나의 OpenURL에는 참고한 자원과 콘텍스트 정보 – OpenURL이 발생한 콘텍스트(예를 들어, 도서관 목록으로 얻은 탐색 결과의 페이지)와 리퀘스트의 콘텍스트(예를 들어, 리퀘스트를 한 특별한 이용자) - 에 대한 정보 둘 다를 포함하고 있다. 만일 다른 콘텍스트가 그 URL에 표현되어 있다면, 그것은 다른 카피를 제시하게 된다. 콘텍스트의 변화는 가능하며 다른 콘텍스트용으로 다른 URLs를 처리할 경우에도 최초의 하이퍼링크(예를 들어, 학술지 출판사)를 만든 제작자를 필요로 하지는 않는다.

예를 들어, 쿼리 스트링에서 기본 URL이나 매개변수가 변한다는 것은 그 OpenURL이 다른 도서관에 있는 자원을 통해 문제를 해결한다는 것을 의미한다. COinS 도 봐라.

1) ContextObjects in Spans, commonly abbreviated COinS,
web pages의 HTML code안에 서지 메타데이터를 내장하는 방법이다. 이것은 서지 소프트웨어로 하여금 서지 메타데이터를 검색하기 위하여 기계가독형 서지 아이템과 client reference management software을 출판하도록 한다. 이 메타데이터는 또한 OpenURL resolver로 보내질 수 있다. 그리고 예를 들어 이것은 one's own library(도서관)에 있는 책의 사본을 탐색할 수 있도록 한다.

> Format

OpenURL은 이용자의 기관 링크-서버의 주소를 포함하고 있는 하나의 기본적 URL과 그 뒤에 전형적으로 key-value pairs의 형태를 갖고 있으며 contextual data를 포함하고 있는 쿼리 스트링으로 구성되어 있다. contextual data에는 대부분이 서지 데이터이지만, 1.0 버전에서처럼, OpenURL 역시 요구자, 하이퍼링크를 포함하고 있는 자원, 요청된 서비스의 유형 등에 대한 정보가 포함될 수 있다.

<<예 1>>
Citation (as found in an information resource):

Moll JR, Olive & M, Vinson C. Attractive interhelical electrostatic interactions in the proline- and acidic-rich region (PAR) leucine zipper subfamily preclude heterodimerization with other basic leucine zipper subfamilies. J Biol Chem. 2000 Nov 3 ; 275(44):34826-32. doi:10.1074/jbc.M004545200

Examples of possible OpenURL's that could be included by the information resource as a means to allow for open linking for the above citation. The OpenURL's that are shown comply with the current draft of the OpenURL specifications. They are encoded as HTTP GET requests:

http://sfx1.exlibris-usa.com/demo?
sid=ebsco:medline
&aulast=Moll&auinit=JR&date=2000-11-03&stitle=J%20Biol%20Chem&volume=275&issue=44&spage=34826

<<<%20: space, %23: #, %2F: / .....>>

http://sfxserv.rug.ac.be:8888/rug

?id=doi:10.1074/jbc.M004545200

Legend:

BASE-URL of service component: http://sfx1.exlibris-usa.com/demo
identifier of the resource where the user clicks the OpenURL: ?sid=ebsco:medline
metadata and identifiers: aulast=Moll&auinit=JR&date=2000-11-03
&stitle=J%20Biol%20Chem&volume=275&issue=44&spage=34826

BASE-URL of service component: http://sfxserv.rug.ac.be:8888/rug?
metadata and identifiers: ?id=doi:10.1074/jbc.M004545200

<<예 2>>

http://resolver.example.edu/cgi
?genre=book
&isbn=0836218310
&title=The+Far+Side+Gallery+3

위의 예는 OpenURL의 버전 0.1로 책에 대해 기술하고 있다.

http://resolver.example.edu/cgi 는 link-server의 기본적인(base) URL 이다.

version 1.0에서 이것은 다음과 같이 다소 길어졌지만 똑같은 링크이다:

http://resolver.example.edu/cgi
?ctx_ver=Z39.88-2004
&rft_val_fmt=info:ofi/fmt:kev:mtx:book
&rft.isbn=0836218310&rft.btitle=The+Far+Side+Gallery+3

위의 query string을 분석해 보면, 다음과 같은 값들이 설정되어 있다는 것을 알 수 있다:

1) version 1.0 OpenURL ContextObject: **ctx_ver = Z39.88-2004**
2) 기술대상(book or journal)에 대한 metadata 포맷:
   rft_val_fmt = info:ofi/fmt:kev:mtx:book(journal)
3) an object named "rft"란 이름의 사물의 표현:
   rft = { isbn:"0836218310", btitle:"The Far Side Gallery 3" }

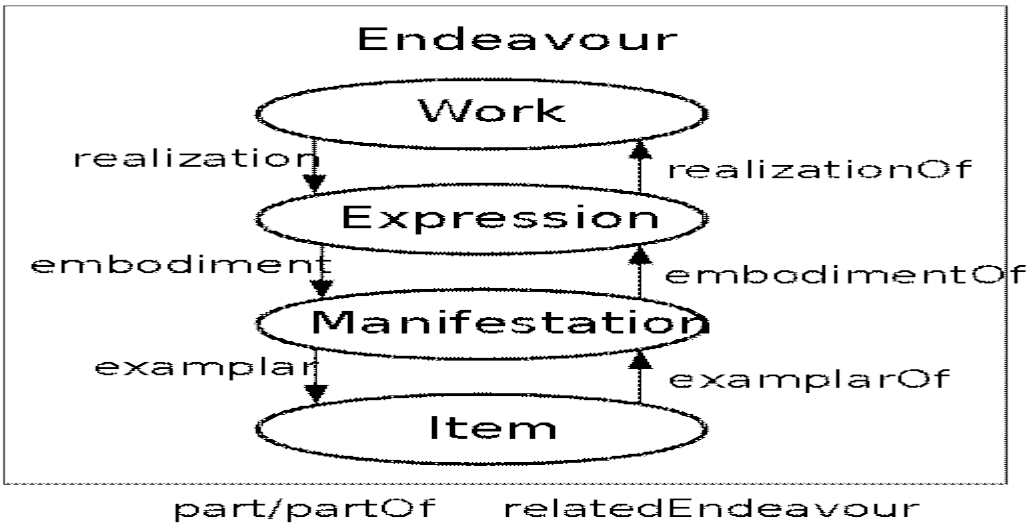
OpenURL은 1990년대 말에 Ghent 대학의 사서인 Herbert Van de Sompel이 개발하였다.

*** FRBR: Functional Requirements for Bibliographic Records;** (/ˈfɜrbər/)

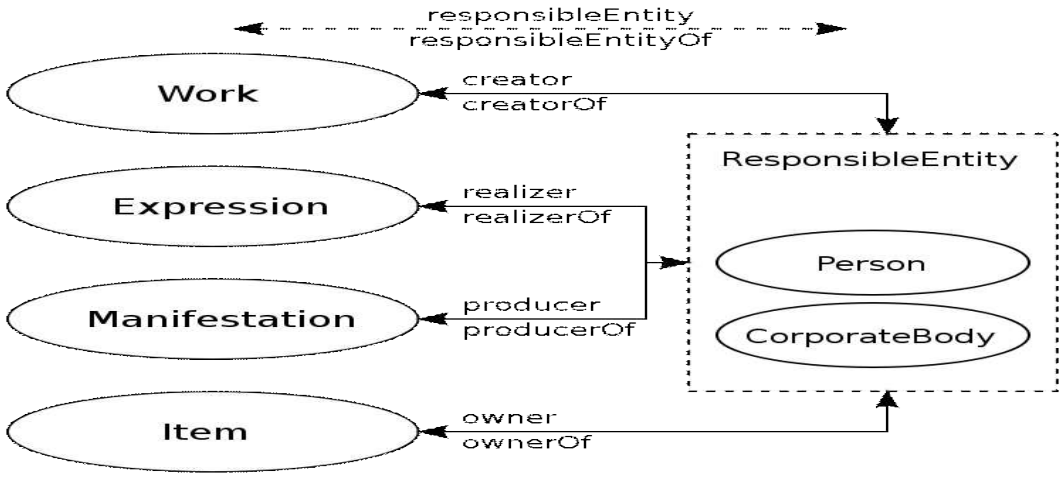
FRBR는 IFLA에서 개발한 개념적 entity-relationship model이며, 이것을 이용자 측면에서 보면, 온라인 도서관 목록과 서지 데이터베이스의 접근과 검색에 대한 이용자 업무와 관련된 것이다. 또한 객체간의 관계는 관계의 계층간을 항해할 수 있는 링크를 제공하기 때문에,

이것은 검색과 접근에 대한 보다 총체적인 방법을 제공한다. 이 모델이 중요한 이유는 AACR2나 ISBD와 같은 특수한 편목 기준과는 차별화되기 때문이다.

> FRBR entities



Group 1 entities and basic relation



Group 2 entities and relation

FRBR는 엔티티의 그룹으로 구성된다:

> 그룹 1 엔티티는 **WEMI(work, expression, manifestation, and item)** 이다. 이것들은 지적 또는 예술적 노력의 결과물을 표현한다.
> 그룹 2 엔티티는 사람, 가족, 그리고 공동체이며, 그룹 1의 지적 또는 예술적 노력을 관리하는 책임을 진다.

> 그룹 3 엔티티는 그룹 1과 그룹 2의 지적 노력의 주제이며, 개념, 대상, 사건, 장소가 포함된다.

그룹 1 엔티티는 FRBR 모델의 기초이다.

► Work란 "분명한 지적 또는 예술적 창조물"이다. 예를 들어, 베토벤의 교향곡 9은 그것을 표현하는 모든 방법과는 달리 하나의 작품(work) 이다. 우리가 "베토벤 9은 훌륭하다"고 말할 때, 일반적으로 그 작품을 말하는 것이다.

► Expression이란 "특별한 지적 또는 예술적 형태이며, 한 작품이 '현실화'될 때 발생한다." 베토벤의 9의 expression은 그가 쓴 악보의 초안일 수도 있다. 종이 그 자체가 아니라 악보에 의해 음악으로 표현된다.

► Manifestation 이란 "어떤 작품을 표현하는 물리적 구체화"이다. 하나의 엔티티로서 mainfestation은 지적인 콘텐트와 물리적 형태 둘 다와 관련해서 동일한 특성을 갖고 있는 모든 물리적 객체를 manifestation이라 한다. " 1996년에 베토벤 9번 교향곡의 런던 필하모니 연주는 manifestation이다. 비록 기록되지는 않더라도, 물론 manifestation이 레코딩이나 프린팅과 같은 항구적인 형태로 표현되어질 때 많은 관심을 받는다 하더라도, 이것은 물리적 구체화이다." 런던 필하모니의 1996년 연주에 대한 레코딩의 핵심은 베토벤 교향곡 9번이므로 우리는 일반적으로 이것을 manifestation이라고 부른다.

► Item이란 "a manifestation의 단일 예이다." 아이템으로 정의된 엔티티는 확실한 엔티티이다." 1996년 레코딩의 사본 각각은 하나의 아이템이다. 우리가 "베토벤 교향곡 9번의 런던 필하모니의 1996년 연주 사본이 나의 지역 도서관에서 대출되었다"고 말할 때, 우리는 일반적으로 특정 아이템에 대해 말하는 것이다.

> Relationships

FRBR은 엔티티 간에 그리고 그 안에 있는 관계를 설정한다. "관계는 엔티티 간의 링크를 설명하는데 필요한 기본적적 도움을 제공하므로, 서지, 목록, 또는 서지 데이터베이스와 같은 세상을 항해하려는 이용자를 도와주는 수단이다. 관계 유형의 예에는 다음과 같은 것이 포함된다:

► Equivalence relationships: 지적 콘텐트와 저작권이 보존되는 한, 한 작품의 동일한 manifestation인 사본들 간에 존재하거나 또는 원작과 그것의 재생품 사이에 존재하는 관계이다. 예로는 copies, issues, facsimiles and reprints, photocopies, and microfilms과 같은 재생품(reproductions)이 있다.

► Derivative relationships: 원저작을 근거로 한 서지물과 수정판(modifications) 사이에 존재하는 관계이다. 예를 들어, Editions, versions, translations, summaries, abstracts, and digests 가 있다. 그리고 또한 새로운 저작이지만 옛 작품을 근거로 한 개작물

(Adaptation), 장르의 변경, 작품의 주제적 콘텐트와 스타일은 유지하는 새로운 저작도 포함된다.

► Descriptive relationships: 저작과 그것을 기술하고 있는 서평 사이에 있는 서지 엔티티와 그 엔티티의 a description, criticism, evaluation, or review 사이에 존재하는 관계이다. 이것의 예로는 기존 저작의 annotated editions, casebooks, commentaries, critiques 도 있다.

p. lxxii <제 3장 메타데이터>

*** Dublin Core**

DC 메타데이터 용어들은 한 무리의 어휘 용어들이며, 찾기를 목적으로 자원을 기술하는데 사용할 수 있다. 이 용어들은 모든 웹 자원(비디오, 이미지, 웹 페이지 등) 그리고 물리적 자원(책과 예술작품)과 같은 모든 자원을 기술하는데 사용될 수 있다. DC 메타데이터 용어들의 풀 세트는 Dublin Core Metadata Initiative (DCMI) website에서 찾을 수 있으며, Dublin Core Metadata Element Set로 알려진 전통적인 15개의 메타데이터 용어 세트는 다음과 같은 표준에서 인증하고 있다.

IETF RFC 5013
ISO Standard 15836-2009
NISO Standard Z39.85

DC 메타데이터는 간단한 자원의 기술에서부터 서로 다른 메타데이터 기준을 가지고 있는 메타데이터 어휘를 결합하는데, 뿐만 아니라 링크된 데이터 클라우드와 시멘틱 웹의 실행에 따른 메타데이터 어휘들의 상호호환성을 제공하는데 까지 사용할 수 있다.

> Levels of the standard

The Dublin Core standard에는 두 가지의 수준이 존재한다: Simple and Qualified. Simple Dublin Core는 15 elements로 구성되어 있다; Qualified Dublin Core는 3개의 추가적인 요소를 포함하고 있으며(Audience, Provenance and RightsHolder), 그 뿐만 아니라 qualifiers라 부르는 한 그룹의 element refinements도 포함하고 있다. elements refinements란 자원발견에 도움을 줄 수 있는 방법으로 요소들의 어의를 정제하는 것을 말한다.

◉ Simple Dublin Core

The Simple Dublin Core Metadata Element Set (DCMES)의 15 metadata elements:

1. Title 2. Creator 3. Subject 4. Description 5. Publisher
6. Contributor 7. Date 8. Type 9. Format 10. Identifier
11. Source 12. Language 13. Relation 14. Coverage 15. Rights

Example of code: **<meta name="DC.Publisher" content="publisher-name" >**

The Dublin Core Metadata Initiative ;
다양한 목적과 경영 모델을 지원하기 위하여 호환적 운영이 가능한 online metadata standards의 개발에 열중하고 있는 공개적 조직이다.

DC-dot: Dublin Core metadata editor);
이 서비스는 웹 페이지를 검색하여 HTML tags처럼 또는 RDF/XML처럼 그 페이지의 section에 적합하게 내재되도록 자동적으로 DC 메타데이터를 생산한다. 필요하다면, 생산된 메타데이터는 여러 가지 포맷(USMARC, SOIF, IAFA/ROADS, TEI headers, GILS, IMS or RDF)으로 제공되거나 변환된 형태(form)를 사용하여 편집할 수 있다.

Editor-Converter Dublin Core metadata;
이 온라인 프로그램은 두 가지의 목적으로 사용될 수 있다: a Dublin Core metadata editor, 그리고 a converter to UNIMARC. UNIMARC format으로 변환이 이루어진 후, 메타데이터는 여러분의 하드 드라이브에 an ISO-2709 file로 저장될 수 있다.

DC-assist;
메타데이터 어플용으로 작고 융통성 있는 help utilty이며 기존의 소프트웨어에 있는 help pages를 보완해주는 목적을 가지고 있다.

*** SOIF: Summary Object Interchange Format**

SOIF는 인터넷상의 각종 자원을 표현하여 이를 네트워크상에서 운용할 수 있는 효과적인 메타데이터이다. SOIF는 미국 콜로라도 대학에서 개발한 Harvest Architecture의 일부로 디자인되었다. 객체개념을 도입한 SOIF는 각 자원의 기술을 객체로 저장하기 때문에 하나의 SOIF 스트림에 여러 개의 객체를 동시에 전송할 수 있다.

Harvest:

하베스트란 인터넷 정보에 접근하고, 복사하고, 캐쉬하고, 색인하고, 수집하기 위하여 측정하고 맞춤화할 수 있는 구조를 제공하는 시스템이다. Harvest 수집가들과 브로커들은 일련의 객체 요약들로 기술되어 있는 SOIF라고 부르는 속성-값 스트림 프로토콜을 사용하여 통신한다. 그리고 SOIF의 레코드들은 Harvest gatherers에 의해 생산된 다음에, Harvest brokers에 의해 이용자 탐색용으로 사용되도록 설계되었으며, 또한 SOIF summary를 plain text, SGML (including HTML), PostScript, MIF and RTF formats으로 생산할 수 있다.

* IAFA/ROADS

ROAD templates는 ROADS 소프트웨어를 사용하는 the subjects services에 의해 사용된다. 이 템플릿들은 1994년의 the Internet Anonymous FTP Archive(IAFA) templates를 발전시킨 것이다. Dublin Core를 이것의 템플릿으로 mapping하는 것은 여러 메타데이터 종류들 간에 교환이 가능한 포맷으로 사용하기 위해서이다.

1) **Template**

> drawing, painting, etc.처럼 그래픽 아트와 letters, shapes or designs를 복사하기 위한 sewing(봉재)에서 사용되는 a stencil(형판), pattern or overlay.

> 비슷한 similar design, pattern, or style을 가지고 새로운 페이지를 만드는데 사용된 전자 또는 종이 매체에 미리 만들어 놓은 page layout

> 가변적 데이터나 텍스트를 개별적으로 맞춤화할 때 본래의 의도를 유지하고 있는 a predefined letter인 form letter

> sample

◉ Dublin Core record:

Title: A Unifying Syntax for the Expression of Names and Addresses of Objects on the Network as used in the World-Wide Web.
Title: (Subtitle) Universal Resource Identifiers in WWW
Creator: Berners-Lee, T.
Subject: IETF, URI, Uniform Resource Identifiers
Publisher: CERN
Date: 1994
Type: Internet RFC
Format (scheme=IMT): text/plain
Identifier(scheme=URL): gopher://gopher.es.net:70/0R0-57601-/pub/rfcs/rfc1630.txt
Relation (type=child)(identifier=URL): http://ds.internic.net/ds/dspg1intdoc.html
Relation (type=sibling)(identifier=URL): http://ds.internic.net/rfc/rfc1738.txt

◉ IAFA / ROADS template record:

Author-Name: Berners-Lee, T.
Category: Internet RFC
Creation-Date: 1994
Format: text/plain
Keyword: IETF, URI, Uniform Resource Identifiers
Publisher-Name: CERN
Title: A Unifying Syntax for the Expression of Names and Addresses of Objects on the Network as used in the World-Wide Web.
Title: Universal Resource Identifiers in WWW
Template-Type: DOCUMENT
URI-v1: gopher://gopher.es.net:70/0R0-57601-/pub/rfcs/rfc1630.txt

* TEI(Text Encoding Initiative) headers:

Text Encoding Initiative는 디지털 형태로 된 텍스트의 표현을 위한 표준을 공동으로 개발하고 유지관리하기 위한 a consortium이다. 주목적은 인문학, 사회과학, 그리고 언어학

에 초점을 맞추어 기계가독형 텍스트의 encoding 방법을 규정하는 Electronic Text Encoding and Interchange를 위한 가이드라인을 제공하는 것이다. 그리고 모든 TEI-conformant text로 된 header에서는 전자 타이틀 페이지를 만드는 데 필요한 기술적이고 선언적인 정보(descriptive and declarative information)을 제공한다.

TEI는 1980년대 이래로 지속적으로 활동하고 있는 디지털 인문학분야의 실질적인 텍스트 지향적 커뮤니티 이다. 이 커뮤니티는 현재 a mailing list, meetings and conference series를 발간하고 있으며 a wiki, a SourceForge repository 그리고 a toolchain과 같은 이름의 시조가 되는 기술 표준을 관리하고 있다.

1) A **wiki** (i/'wɪki/ WIK-ee)
다른 사람과 협력해서 컨텐트를 추가, 변경, 삭제하도록 하는 웹 어플을 말한다. 텍스트는 일반적으로 a simplified markup language or a rich-text editor로 작성되며, 가장 인기있는 백과사전은 Ikipedia이다.

2) **SourceForge**
a web-based source code repository이며, 무료 및 공개적인 source software development을 통제하고 관리하기 위하여 소프트웨어 개발자에게 핵심적 역할을 한다. 무료이면서 공개적인 소스 프로젝트를 위해 이 서비스가 제공되는 것이 처음이었다.

3) **toolchain**
전형적으로 또다른 컴퓨터 프로그램이나 프로그램 시스템과 같은 제품을 만드는데 사용되는 프로그램 도구의 세트이다. 각 도구의 출력을 다음 도구의 입력으로 사용하기 위하여 이 도구들은 chain으로 사용될 수도 있으나, 이 용어는 어떤 linked development tools이라고 폭넓게 부르고 있다.

XML 포맷을 총괄적으로 정의한 TEI Guidelines는 실무 집단의 명확한 결과물이다. 이것의 포맷은 텍스트용으로 잘 알려진 다른 개방형 포맷(프레젠테이션보다는 기본적으로 시멘틱적인 포맷인 HTML과 OpenDocument)과는 다르다; 모든 태그와 속성의 시멘틱스와 해석은 정해져 있다. 약 500개의 서로 다른 텍스트 구성 요소와 개념(단어, 문장, 인물, 상형문자, 사람, 등) 각각은 하나 이상의 학술분야에 그 뿌리를 두고 있으며 예들을 제공하고 있다.

1) The Open Document Format for Office Applications (ODF), also known as **OpenDocument,**
spreadsheets, charts, presentations and word processing documents를 위한 an XML-based file format 이다. 이것은 office applications을 위한 an open, XML-based file format specification을 제공하려는 목적으로 개발되었다.

이것의 표준은 a technical committee in the Organization for the Advancement of Structured Information Standards (OASIS) consortium에서 개발하였으며, OpenOffice.org을 위한 디폴트 포맷인 OpenOffice.org XML용으로 the Sun Microsystems specification을 기본으로 하고 있다.

OASIS standard와 더불어, version 1.1은 an ISO/IEC international standard, ISO/IEC 26300:2006/Amd 1:2012 – Open Document Format for Office Applications (OpenDocument) v1.1로 출판되었다..

이 표준은 두 부분으로 구분 된다: 확장된 예와 토론이 있는 추론적인 텍스트 기술과 tag-by-tag 세트의 정의. 대부분의 최신 포맷(DTD, RELAX NG, W3C Schema)에서의 구조식은 자동적으로 tag-by-tag 정의를 통해 생산된다. 수많은 도구가 이 가이드라인의 생산과 특별한 프로젝트에 적용시키는데 도움을 주었다. 그리고 수많은 특별한 태그들이 기초가 되는 Unicode(요구되는 엄격한 선형성의 극복을 위하여 Unicode의 내포와 선택에 대한 자격을 요구하지 않는 문자들의 표현을 허용하는 그림문자)에 의해 부여된 제한을 완화시키는데 사용되

고 있다.

1) A document type definition (DTD)
한 세트의 markup declarations이다. 이것은 SGML-family markup language (SGML, XML, HTML)용으로 사용되는 다큐멘트의 유형을 정의한다. DTD는 어떠한 요소들과 참고들이 특별한 유형의 다큐멘트 어디에 나타나야 하는지 그리고 요소들의 컨텐트와 속성들이 무엇인지를 정확하게 선언하기 위하여 간결한 공식적 구문을 사용한다. 또한 DTD는 the instance document에서 사용될 수 있는 엔티티들을 선언할 수 있다. XML은 a subset of SGML DTD를 사용한다.

2) RELAX NG (REgular LAnguage for XML Next Generation)
전산에서 이것은 XML용의 스키마 언어이다 - RELAX NG 스키마는 XML document의 구조와 컨텐트용 패턴을 지정한다. A RELAX NG schema는 그 자체가 XML document이자만, 또한 RELAX NG는 a popular compact, non-XML syntax를 제공한다.

3) The World Wide Web Consortium (W3C)
이것은the World Wide Web (abbreviated WWW or W3)을 위한 the main international standards organization 이다.

<History>

The World Wide Web Consortium (W3C) was founded by **Tim Berners-Lee** after he left the European Organization for Nuclear Research (CERN) in October, 1994. It was founded at the Massachusetts Institute of Technology Laboratory for Computer Science (MIT/LCS) with support from the European Commission and the Defense Advanced Research Projects Agency (DARPA), which had pioneered the Internet and its predecessor **ARPANET**.

W3C tries to enforce compatibility and agreement among industry members in the adoption of new standards defined by the W3C. Incompatible (모순된)versions of HTML are offered by different vendors, causing inconsistency in how Web pages are displayed. The consortium tries to get all those vendors to implement a set of core principles and components which are chosen by the consortium.

It was originally intended that CERN host the European branch of W3C; however, CERN wished to focus on particle physics, not information technology. In April 1995 the Institut national de recherche en informatique et en automatique (INRIA) became the European host of W3C, with Keio University becoming the Japanese branch in September 1996. Starting in 1997, W3C created regional offices around the world; as of September 2009, it has eighteen World Offices covering Australia, the Benelux countries (Netherlands, Luxembourg, and Belgium), Brazil, China, Finland, Germany, Austria, Greece, Hong Kong, Hungary, India, Israel, Italy, South Korea, Morocco, South Africa, Spain, Sweden, and the United Kingdom and Ireland.

In January 2003, the European host was transferred from INRIA to the European Research Consortium for Informatics and Mathematics (ERCIM), an organization that represents European national computer science laboratories. In October 2012, W3C convened(소집하다) a community of large Web players and publishers to establish a MediaWiki wiki that seeks to documents open Web standards called WebPlatform and WebPlatform Docs.

이 포맷을 사용하는 많은 이용자들은 모든 태그를 사용하는 것이 아니라 그것의 하위세트를 가지고 맞춤식으로 사용한다. 다시 말해서, 이 포맷은 TEI 가이드라인과 관련된 학술분야의 장절(chapter)처럼 각각의 하위세트에 있는 태그들을 그룹화할 수 있도록 지원하고 있다. 또한 이 포맷의 몇몇 이용자들은 더 나아가서 출판을 보다 쉽게 할 수 있도록 자신들의 local house style을 근거로 schematron stylesheet를 설명하고 있다.

1) Schematron
마크압 언어에서, 이것은 XML trees 속에 패턴의 존재와 부재에 대한 assertions(주장)을 하기 위한 a rule-based

validation language 이다. 또한 이것은 소수의 요소들과 XPath를 사용하는 XML에서 표현된 구조적 스카마 언어이다.

2) XPath: the XML Path Language
XML document로부터 nodes를 선택하는데 사용되는 query language 이다. 추가로, XPath는 XML document 의 컨텐트로부터 값들(e.g., strings, numbers, or Boolean values)을 계산하는데 사용할 수 있다.
이것은 the World Wide Web Consortium (W3C)에 의해 정의되고 있다.

마지막으로 TEI Lite는 텍스트를 교환하기 위한 XML 의존형 파일 포맷이다. 이것은 TEI Guidelines의 완전판에 있는 광대한 무리의 이용 가능한 요소들로부터 다루기 쉬운 것만을 발췌해 놓은 것이다.

*** GILS: Global Information Locator Service**

사람들이 스스로 필요로 하는 모든 정보를 보다 쉽게 찾도록 하는 서비스이다; 도서관 이용의 경험이 있는 사람이면 누구나 이 서비스를 이용할 수 있으며, ISO 23950 search standard를 근거로, title, author, publish, date and place와 같은 도서관 분야의 개념을 도입하여, 전 세계 누구나 정보자원을 찾는데 가장 쉽게 이해할 수 있는 개념을 사용하고 있다. GILS record는 도서관의 목록 레코드의 일종의 강화판(souped-up version) 이다.

*** IMS: IP Multimedia Subsystem or IP Multimedia Core Network subsystem**

이것은 Internet Protocol(IP) multimedia services를 전달하기 위한 구조적 기본 틀이다. 역사적으로 엄격한 an IP packet-switched network에서 보다는 휴대전화는 a switched-circuit-style network에서 voice call service를 제공해 왔다. IP에서 음성이나 기타 멀티미디어 서비스를 전달하는 대안적 방법들(e.g., VoIP or Skype)을 스마트폰에서 이용 가능하게 되었지만, 그것들은 산업 간에 표준화가 되지 못했다. IMS는 그러한 표준화을 제공하는 구조적 기본틀 이다.

IMS는 원래 GSM을 뛰어넘는 모바일 네트워크를 발전시키려는 비전(vision)에 따라, the wireless standards body인 3rd Generation Partnership Project (3GPP)에 의해 디자인되었다. 이것의 최초의 formulation(3GPP Rel-5)은 GPRS에서 “인터넷 서비스”를 전달하려는 시도였다. 이 비전은 나중에 Wireless LAN, CDMA2000 그리고 fixed lines과 같은 GPRS보다 다른 네트워크를 지원해야 한다는 요구에 따라 3GPP, 3GPP2 그리고 ETSI TISPAN에 의해 갱신되었다.

1) GSM (Global System for Mobile Communications, originally Group Spécial Mobile),
mobile phones에서 사용된 second generation (2G) digital cellular networks용 프로토콜을 기술하기 위하여 the European Telecommunications Standards Institute (ETSI)에서 개발한 표준이다. The GSM standard 은 first generation (1G) analog cellular networks를 대체하기 위하여 개발되었으며, 원래는 full duplex voice telephony를 최적화하기 위한 digital, circuit-switched network를 기술하였다. 이것은 시간이 지나면서, 데이터 통신쪽으로 확대되었다: 처음에는 circuit-switched transport로, 그런 다음에는 packet data transport via GPRS (General Packet Radio Services) and EDGE (Enhanced Data rates for GSM Evolution or EGPRS).

결론적으로, the 3GPP는 third generation (3G) UMTS standards, 그리고 뒤이어 ETSI GSM standard의 부분

이 아닌 fourth generation (4G) LTE Advanced standards을 개발하였다.

2) General packet radio service (GPRS)

이것은 mobile communications (GSM)을 위한 the 2G and 3G cellular communication system's global system에서 이루어지는 a packet oriented mobile data service 이다. GPRS는 원래 European Telecommunications Standards Institute (ETSI)에 의해 표준화 되었으며, 이것은 이전의 CDPD and i-mode packet-switched cellular technologies에 대응하기 위한 것이다. 이것은 현재 the 3rd Generation Partnership Project (3GPP)의해 관리되고 있다. 그리고 GPRS 사용는 연결시간의 분당 요금을 청구하는 circuit switched data 와는 대조적으로, 전달된 데이터의 크기에 따라 요금을 부과하고 있다.

GPRS is a best-effort service, implying variable throughput(처리량) and latency(잠재물) that depend on the number of other users sharing the service concurrently, as opposed to circuit switching, where a certain quality of service (QoS) is guaranteed during the connection. In 2G systems, GPRS provides data rates of 56–114 kbit/second. 2G cellular technology combined with GPRS is sometimes described as 2.5G, that is, a technology between the second (2G) and third (3G) generations of mobile telephony. It provides moderate-speed data transfer, by using unused time division multiple access (TDMA) channels in, for example, the GSM system. GPRS is integrated into GSM Release 97 and newer releases.

인터넷과 통합을 쉽게 하기 위하여, IMS는 예를 들어, SIP처럼 가능하다면 어디에서나 IETRF 프로토콜을 사용한다. 3GPP에 따라, IMS는 어플을 표준화하려는 것이 아니라, 그 보다는 무선과 유선 터미널로부터 멀티미디어와 음성 어플의 접근, 다시 말해서 FMC(Fixed-Mobile Convergence(수렴))의 form을 제작하는데 도움을 준다. 이것은 서비스 레이어로부터 접근 네트워크를 고립시키는 a horizontal control layer를 갖고 있기 때문에 가능한 것이다. 논리적인 구조 측면에서 보면, control layer는 일반적으로 수평적 레이어 이므로, 서비스들은 자신들만의 제어 기능을 가질 필요가 없다. 그렇지만 이렇게 실행하는 것이 반드시 비용과 복잡성을 크게 줄인다고는 말할 수 없다.

1) The Session Initiation Protocol (SIP)

Internet Protocol (IP) networks에서 voice and video calls과 같은 multimedia communication sessions을 통제하기 위해 널리 사용되는 a signaling communications protocol 이다.

이 프로토콜은 establishment, termination and other essential elements of a call을 관리하는 peer들 간에 주고받는 메시지를 정의한다. SIP는 하나 또는 다수의 미디어 streams로 이루어진 sessions을 만들고, 변경하고, 종료하기 위하여 사용될 수도 있다. SIP는 two-party (unicast) or multiparty (multicast) sessions용으로 사용될 수도 있다. 다른 SIP 어플에서는 video conferencing, streaming multimedia distribution, instant messaging, presence information, file transfer, fax over IP and online games도 가능하다.

2) Fixed-mobile convergence (FMC)

이것은 텔레커뮤니케이션에서 고정식 그리고 이동식(fixed and mobile) networks 간의 차이를 제거시키는 a change 이다. 2004에 the Fixed Mobile Convergence Alliance에서 발표하길: “Fixed Mobile Convergence란 telecommunications industry에서의 고정식 그리고 이동식 네트워크의 차이를 최종적으로 제거시킬 수 있는 a transition point 이다. 그럼으로써 fixed broadband와 local access wireless technologies를 결합하여 매듭 없는 서비스를 제작함으로써 이용자가 집, 사무실, 건물 등에서 자신들이 필요한 것을 충족시키는데 보다 우수한 경험을 제공하게 될 것이다.”

BT's initial FMC service used Bluetooth rather than Wi-Fi for the local access wireless. The advent of picocells and femtocells means that local access wireless can be cellular radio technology.

이 정의에서 “fixed broadband”란 DSL, cable or T1와 같은 인터넷으로의 접속을 의미하며, “Local access wireless”는 Wi-Fi와 그것과 유사한 것들을 의미한다. BT의 초기의 FMC service는 local access wireless용으로 Wi-Fi 보다는 Bluetooth를 사용한다. **picocells**(전형적으로 건물 내의 offices, shopping malls, train stations, stock exchanges, etc.과 같은 작은 지역을 담당하는 소규모의 cellular base station을 말한다), or more recently in-aircraft)과 **femtocells**(전형적으로 가정이나 작은 사업체에서 사용하도록 디자인된 a small,

low-power cellular base station을 말한다.)의 출현이 의미하는 것은 local access wireless가 cellular radio technology일 수 있다는 것이다.

위에서 “seamless services”란 모호하다. FMC에 대해 이야기할 때, “seamless” 단어는 대체로 “seamless handover ”를 말한다. 여기서 seamless handover란 다음과 같이 one of the FMCA specification documents의 한 곳에서 설명했듯이, 전화(call)를 방해 없이 이동식 네트워크로부터 고정식 네트워크로 넘겨주는 것을 의미한다: seamless란 handover로 인하여 음성이나 데이터 전달에 어떠한 깨짐도 발생하지 않는 것을 말한다 (from the calling party or the called party”s perspective).

*** RDF: Resource Description Framework; see p. 186**

The Resource Description Framework (RDF)는 원래 메타데이터 데이터 모델로 디자인된 W3C 스펙의 한 부류이며, 웹 페이지의 title, author, modification date, content, and copyright information 와 같은 웹 자원을 기술하기 위한 W3C 표준이다.

*** PDF: Portable Document Format**

PDF는 응용 소프트웨어, 하드웨어, 그리고 운영체계와는 독립적인 방식으로 다큐멘트를 표현하는 파일 포맷이며, 각각의 PDF 파일은 그것을 디스플레이 하는데 필요한 text, fonts, graphics, 그리고 기타 정보를 포함하고 있는 a fixed-layout flat document 속에 들어 있다. 이것은 또한 사용자가 보거나 탐색하거나 프린트하거나 또는 다른 사람에게 전달할 수 있도록 만들어진 이미지로서 프린트 출력된 문서의 모든 요소를 갖추고 있는 파일 포맷 이다 (Adobe Reader).

*** TIFF: Tag Image File Format**

TIFF는 일반적으로 graphic artists, the publishing industry, and both amateur and professional photographers 간에서 인기 있는 raster(점방식) graphics images를 저장하기 위한 컴퓨터 파일 포맷이다. 이 포맷은 원래 Aldus 사에서 데스크 탑 출판용으로 만들었지만, 2009년부터 Adobe Systems에서 관리하고 있다.

이것의 특징은 또한 이미지 저장 포맷으로 사용자가 수정해서 쓸 수 있다는 것이고, 이것의 확장자는 .tiff 나 .tif이다; 이런 확장자의 파일 포맷 유형은 raster image(bit map style) 이다.

*** library 2.0**

라이브러리 2.0은 도서관이 오랫동안 추구해 온 이용자 중심의 서비스 제공을 위해서 참여, 공유, 개방, 소통을 모토로 하는 웹 2.0의 개념과 기술을 도서관에 접목한 개념이다. 2005년 Micheal Casey에 의해 처음으로 언급되었으며, 이용자 위주의 서비스로 변화하고 있는 도서관의 트렌드를 반영하고 있다고 할 수 있다. Library 2.0의 개념은 Business 2.0과 Web 2.0의 개념에서부터 온 것이며, 몇 가지 동일한 기본적 철학을 가지고 있다.

> **Overview**

"Library 2.0" 용어는 **Michael Casey**에 의해 그의 블로그 LibraryCrunch에서 Business 2.0 and Web 2.0.을 본 따서 처음 사용하였다. Casey가 말하길, "웹 2.0의 많은 요소들을 도서관 커뮤니티에도 적용할 가치가 있으므로, 도서관 특히 공공도서관은 technology-driven services and in non-technology based services 둘 다의 십자로에 있다."고 하면서, 특히, 도서관 이용자의 참여를 촉진하기 위하여 지속적인 변화 전략을 도서관은 추구해야 한다고 주장하였다.

With Library 2.0, library services are frequently evaluated and updated to meet the changing needs of library users. Library 2.0 also calls for libraries to encourage user participation and feedback in the development and maintenance of library services. The active and empowered library user is a significant component of Library 2.0. With information and ideas flowing in both directions – from the library to the user and from the user to the library – library services have the ability to evolve and improve on a constant and rapid basis. The user is participant, co-creator, builder and consultant – whether the product is virtual or physical.

> **Key principles**

1) Browser + Web 2.0 Applications + Connectivity = Full-featured OPAC
2) 서비스의 디자인과 실행 모두에 이용자가 참여(Harness:마구, 이용하다)한다.
3) 이용자는 서비스를 제공한 도서관을 발전시키고(craft) 변화시킬 수 있어야 한다.
4) 주위(peripheral)의 아이디어와 생산물을 수집하여 도서관 서비스 모델에 통합시킨다.
5) 계속해서 서비스를 조사하고 개선시키며, 언제든지 보다 새롭고 좋은 서비스로 대체시키도록 노력한다.

► 장점:

1) 정보 공유의 플랫폼으로서의 도서관
2) 학술, 연구 커뮤니티의 중심으로서의 도서관
3) 개인별 맞춤식 도서관: SDI서비스, AJAX(Asynchronous JavaScript and XML, 아이아스: 대화식 웹 애플리케이션의 제작을 위해 아래와 같은 조합을 이용하는 웹 개발 기법이다; 표현정보를 위한 HTML or XHTML과 CSS, 동적 화면 출력 및 표시 정보와의 상호 작용을 위한 DOM, JavaScript, 웹서버와 비동기적으로 데이터를 교환하고 조작하기 위한 XML, XSLT, XMLHttpRequest)나 RSS feed.
4) 이용자와 소통하는 도서관: 온라인 참고서비스(실시간 메신저, 게시판, 이메일, 전화, SMS, PDA 등 다양한 커뮤니케이션 매체를 통해 가능)
5) 정보 활용 능력을 교육하는 도서관: 온라인 튜토리얼을 통해 이용자의 정보활용능력을 향상.

p. lxxiv

*** MODS: Metadata Object Description Schema, Library of Congress**

The Library of Congress' Network Development & MARC Standards Office에서 2002년에 다양한 목적과 특히 도서관 어플용으로 사용할 수도 있는 한 무리의 서지 요소 인 MODS를 개발하였다. 이것은 an XML-based bibliographic description schema 이며, 도서관에서 사용하는 MARC 포맷의 복잡성과 DC 메타데이터의 극단적 단순성 간의 타협용으로 설계되었다. 이것은 어떤 경우에는 MARC 21 서지 포맷에서 나온 요소들을 재집단화기 위하여MARC 필드의 하위 세트를 포함하고 있으며 숫자적인 것보다는 언어-의존형 태그들을 사용하고 있다. 이것의 기준은 the Network Development and MARC Standards Office of the Library of Congress의 지원을 받아 the MODS Editorial Committee에서 관리하고 있다.

> MARC와의 관계

MODS 레코드는 MARC 레코드로부터 중요한 데이터 요소들을 가져오도록 설계되었지만, 모든 MARC 필드들을 정의하지는 않으며, MARC 표준으로부터 tagging한 필드와 하위필드도 사용하지 않는다. MODS에는 데이터 요소들이 존재하는데, 이것들은 MARC 레코드와 호환하지 않으므로 MARC를 MODS로 또는 MODS를 MARC로 번역할 때 어떤 손실이 발생한다.

> 장점

- ► 기존의 자원기술에 대하여 고도의 호환성을 제공한다.
- ► MARC보다 detail하지 못하므로, 다양한 내적 레코드 세트들을 MODS에 mapped할 수 있다.
- ► 외부에 있는 DC나 기타 보다 단순한 포맷을 사용하는 item descriptions를 mapped하고 enhanced할 수 있다.

*** MOA2 DTD; METS로 대체됨.**

The Making of America II Testbed Project는 디지털 객체 관리에 필요한 메타데이터 요소들을 잘 정리된 세트의 개발을 통해 디지털 도서관 객체용 메타데이타와 관련된 문제들을 해결하려고 시도하였다. 이 메타데이터 세트는 XML 다큐멘트 유형을 정의한 MOA2 DTD를 통해 완전하게 기술적으로 표현 되었다. 디지털 객체 메타데이터의 토론에 관한 훌륭한 출발점을 제공하는 동안, MOA2 DTD는 단지 제한된 범위의 디지털 객체(diaries, still images, ledgers(원부), and letterpress books)만을 코드화하도록 디자인되었다. DTD가 보다 널리 적용됨으로써, 그것의 본래의 디자인이 가지고 있는 문제가 더욱 분명하게 나타났다. 말 그대로 DTD로는 기술 메타데이터의 코딩을 올바르게 준비할 수가 없지만, 극히 부분적으로만 text-와 image-based 자원을 위한 기술적 메타데이터를 지원하기도 한다. 따라서 이것은 audio, video, and other time dependent media에 대한 어떠한 지원도 제공하지 않으

며, 단지 최소한의 내부적 그리고 외부적 링크 기능만을 제공한다.

이러한 단점에도 불구하고, MOA2 DTD는 디지털 도서관 객체의 기술과 관리용으로 표준화된 데이터 요소 세트와 그 정보를 표현하기 위한 기술적 메카니즘 둘 다를 발전시키는데 커다란 역할을 해왔다.

MOA2 DTD 개정판에 영향을 끼친 관련 표준은 다음과 같다: Dublin Core, SMARC, Encoded Archival Description, Indecs Metadata Framework, VRA Core, NISO Technical Metadata for Digital Still Images, Library of Congress audio/visual technical metadata, National Library of Australia Preservation Metadata for Digital Collections, Resource Description Framework, Synchronized Multimedia Integration Language, MPEG-7

1) EAD: Encoded Archival Description
이것은 공문서 발견 도구를 암호화하기 위한 XML standard 이며, the Technical Subcommittee for Encoded Archival Description of the Society of American Archivists, in partnership with the Library of Congress 에서 관리하고 있다. **See p.78**

2) indecs (an acronym of "interoperability of data in e-commerce systems"; written in lower case)
이것은 1998-2000년 사이에 수많은 메타데이터 활동에서 사용되어온 music, rights, text publishing, authors, library and other sectors를 표현하기 위하여 the European Community Info 2000 initiative와 여러 기관에 의해 자금 지원을 받은 프로젝트의 일부이다. 이것의 최종 보고서와 관련 문서가 출판되었으며, the **indecs Metadata Framework** document "Principles, model and data dictionary"는 간략한 요약본이다.

이것은 어의적 호환성에 초점을 맞추어 네트워크 환경에서 컨텐트의 e-commerce용으로 사용되는 메타데이터의 요구조건을 분석하여 제공하고 있다. 어의적 호환성이란 한 컴퓨터 시스템에서 어떻게 다른 컴퓨터 시스템에서 의미하는 용어에 대하여 알 수 있는가와 같은 의문을 다루는 것이다(e.g. if A says "owner" and B says "owner", are they referring to the same thing? If A says "released" and B says "disseminated", do they mean different things?).

3) Synchronized Multimedia Integration Language (SMIL (/'smaɪl/))
이것은 multimedia presentations을 기술하기 위하여 World Wide Web Consortium에서 추천한 Extensible Markup Language (XML) markup language 이다. 또한 이것은 다른 사물들 사이에서 이루어지는 timing, layout, animations, visual transitions, and media embedding에 관한 markup을 정의한다. SMIL은 text, images, video, audio, links와 같은 미디어 아이템들을 다른 SMIL 프레젠테이션들과 다수의 웹 서버로부터 온 파일들에 제공할 수 있도록 한다. SMIL markup은 XML로 작성되며 HTML과 유사성을 갖고 있다.

4) MPEG-7
이것은 multimedia content description standard이며, ISO/IEC 15938 (Multimedia content description interface)로 표준화되었다. 이것은 이용자가 관심대상인 자료를 신속하고 효율적으로 탐색하도록 컨텐트 그 자체에 결합시킬 수 있다. MPEG-7는 공식적으로 Multimedia Content Description Interface라 부른다. 그러므로 이것은 MPEG-1, MPEG-2 and MPEG-4처럼 moving pictures와 audio의 실제적 코딩을 다루는 표준이 아니다. 이것은 메타데이터를 저장하기 위하여 XML을 사용하며, 예를 들어 특별한 events를 tag하기 위하여 또는 가사를 노래에 일치시키기 위하여 timecode에 부착(attached)될 수 있다.

p. lxxviii

*** EAD(Encoded Archival Description)**

EAD DTD의 개발 프로젝트는 1993년 the University of California, Berkeley의 도서관에서 시작되었다. 이 버클리 프로젝트의 목표는 자신들이 소장하고 있는 자료의 이용을 지원하기 위하여, archives, libraries, museums, manuscript repositories에서 만든 inventories, registers, indexes, 그리고 기타 documents와 같은 자원들을 기계가독형 찾기 도구를 사용하여 찾을 수 있도록, 비-독점권 코딩의 표준에 대한 개발 가능성을 조사하는 것이었다.

이 프로젝트의 관리자는 소장 자료의 정보에 접근하는데 네트워크의 역할이 크게 늘어났다는 것을 인정하여 전통적으로 MARC에서 제공하는 것 이상의 정보를 포함시키려고 노력하였다. 이 버클리 프로젝트의 주요 조사자인 Daniel Pitti는 다음과 같은 기준을 근거로 코딩 표준의 필요조건을 개발하였다:

1) 고문서찾기도구로 개발된 포괄적이면서도 서로 연관된 기술정보의 표현 능력;
2) 기술 수준 차이로 존재하는 계층적 관계의 보존 능력;
3) 한 계층적 단계에서부터 다른 단계로 파급(유전)되는 기술정보의 표현 능력;
4) 계층적 정보 구조 내에서 이동할 수 있는 능력;
5) 요소-맞춤형 색인과 검색의 지원.

p. lxxxv

*** DOI:** A Digital Object Identifier

DOI는 전자 다큐멘트와 같은 객체를 유일하게 식별하기 위하여 사용되는 문자열(디지털 식별자)이다. 사물에 대한 메타데이터는 DOI 이름과 결합하여 저장되며, 이 메타데이터에는 해당 사물을 찾을 수 있는 URL과 같은 위치가 포함되기도 한다. 다큐멘트용 DOI는 항구적인 것인 반면에 그것의 위치와 기타 메타데이터들은 변할 수도 있다. DOI로 온라인 다큐멘트를 표현하는 것이 단지 URL로만 표현하는 것보다는 더욱 안정된 링킹을 제공하는데, 왜냐하면 만일 어떤 사물의 URL이 변한다면, 출판사는 새로운 URL로 링크하기 위하여 단지 그것의 DOI용 메타데이터만을 갱신하면 되기 때문이다.

> DOI names

DOI name은 ISBN이나 ISRC와 같은 standard identifier registries와는 다르다. identifier registry의 목적은 특정한 집단의 식별자들을 관리하는 것인 반면에, DOI system의 중요한 목적은 식별자들을 활동과 협력을 가능하게 만드는 것이다.

DOI name은 문자열 형태를 취하며, 슬래쉬(/)로 구분되는 접두사와 접미사, 두 부분으

로 나뉘어져 있다. 접두사는 그 이름의 등록자를 나타내며, 접미사는 그 등록자에 의해 선택되어 그 DOI와 결합된 특별한 객체를 나타낸다. 대부분의 합법적인 Unicode 문자들은 이 문자열에 포함될 수 있으며 대소문자를 구별하여 해석된다.

1) The International Standard Recording Code (ISRC)

이것은 sound recordings and music video recordings를 유일하게 식별하는 국제 표준 코드이다.

The code was developed by the recording industry in conjunction with the ISO technical committee 46, subcommittee 9 (TC 46/SC 9), which codified the standard as ISO 3901 in 1986, and updated it in 2001. An ISRC identifies a particular recording, not the work (composition and lyrical content) itself. Therefore, different recordings, edits, and remixes of the same work should each have their own ISRC. Works are identified by ISWC(International Standard Musical Work Code). Recordings remastered without significant audio-quality changes should retain their existing ISRC, but the threshold is left to the discretion of the record company.

예를 들어, DOI name 10.1000/182에서 접두어는 10.1000이고, 접미사는 182이다. 접두사의 "10."은 그 DOI의 등록지를 나타내며, 접두사에 있는 문자 1000은 등록자를 의미한다; 위의 예에서 등록자는 the International DOI Foundation 이다. 또한 182는 접미사이거나 아이템 ID이며 단일 사물을 나타낸다(위의 예는 the DOI Handbook의 최신 버전이다). DOI name을 이용하는 인용문에서는 doi:10.1000/182처럼 표기해야 한다. 인용문이 하이퍼링크일 경우에는 DOI name에서 "doi:" 접두사를 생략하고, 그 자리를 "http://dx.doi.org/"를 대체하여 하나의 URL처럼 사용하길 권장하고 있다. 예를 들어, DOI name doi:10.1000/182는 http://dx.doi.org/10.1000/182처럼 링크되도록 사용한다. 따라서 이 URL에서는 링크된 아이템의 정확한 온라인 위치로 웹 접근의 방향을 조정하도록 하는 HTTP proxy server의 위치를 제공하게 된다.

DOI name을 사용하여 전자적 또는 물리적 형태 둘 모두에 포함되는 창조적 작품(texts, images, audio or video items, and software)과 퍼포먼스, 교신 대상 등의 추상적 작품을 구분할 수 있다. 또한 이 name의 세부적 내용을 변경하는 과정을 통해 사물들을 표현할 수 있다: 그러므로 DOI name은 a journal, an individual issue of a journal, an individual article in the journal, 또는 a single table in that article 등을 구별할 수 있다. 세부내용의 수준에 대한 선택은 배정자에게 달려있지만, DOI 시스템에서 이것은 the indecs(interoperability of data in e-commerce systems) Content Model에 근거한 데이터 사전을 이용하여야 하므로, 분명하게 하나의 DOI name에 결합되어 있는 메타데이터의 일부라고 선언되어야 한다.

> Resolution

DOI name resolution은 Handle System을 통해 제공되며, DOI name을 만나는 어떠한 이용자도 무료로 이용할 수 있다. 리솔루션은 하나의 DOI name에서부터, URLs 객체의 경우들을 표현하는 URLs, 이-메일과 같은 서비스, 또는 하나 이상의 메타데이터 아이템들 같이 하나이상의 타이핑된 데이터 조각으로 이용자의 방향을 수정해 준다. Handle System에서 DOI name은 하나의 핸들이며 그렇게 때문에 그것에 할당된 한 세트의 값을 가지고 있고 한 그룹의 필드로 구성된 하나의 레코드처럼 생각할 수도 있다. 각 핸들의 값은 데이터의 구문과

어의를 정의하고 있는 그것의 “<type>” 필드에 지정된 데이터 유형만을 가져야만 한다.

1) **The Handle System**
이것은 인터넷에서 디지털 사물과 기타 자원의 항구적인 식별자(persistent identifiers)를 배정하고, 관리하고, 해결하기 위한 기술 스펙이다.

The protocols specified enable a distributed computer system to store identifiers (names, or handles), of digital resources and resolve those handles into the information necessary to locate, access, and otherwise make use of the resources. That information can be changed as needed to reflect the current state and/or location of the identified resource without changing the handle.

DOI name을 분석(resolve)하려면, DOI resolver(예: www.doi.org)에 그것을 입력하여야 하며, 문자열 http://dx.doi.org/를 사용하여 DOI name을 표현할 수도 있다. 예를 들어, DOI name 10.1000/182는 어드레스 “http://dx.doi.org/10.1000/182”로 해결될 수 있다. 웹 페이지나 기타 하이퍼텍스트 다큐멘트들은 이러한 형태의 하이퍼텍스트 링크들을 포함할 수 있다. 어떤 브라우저들은 추가기능(add-on)을 갖춤으로써, DOI(또는 다른 핸들)의 직접적인 분석을 허용하고 있다(예: CNRI Handle Extension for Firefox). The CNRI Handle Extension for Firefox에서는 그것의 브라우저로 하여금 고유한 Handle System 프로토콜을 사용하는 hdl:4263537/4000 또는 doi:10.1000/1와 같은 핸들이나 DOI RURIs에 접근 가능케 한다.

1) The **Corporation for National Research Initiatives (CNRI)**,
이것은 미국의 National Information Infrastructure를 포함하여 “네트워크-기반 정보기술의 전략적 발전을 위한 활동 센터”로서 1986년 Robert E. Kahn에 의해 설립된 비영리 기관이다.

CNRI publishes D-Lib Magazine, a journal of digital library research and development. It also develops the Handle System for managing and locating digital information.

> Organizational structure

IDF는 비영리기관으로 1998년에 수립되었으며 DOI 시스템의 총괄기관이다. 이곳에서는 DOI 시스템과 관련된 모든 지적 재산권을 보호하고 있으며, 일반적인 운영상의 기능도 관리하고 있을 뿐만 아니라 DOI 시스템의 개발과 발전을 지원하고 있다. IDF에 의해 지명된 등록기관은 DOI 등록자에게 서비스를 제공 한다: 이것들은 DOI 접두어를 할당하고, DOI names을 등록하며, 등록자로 하여금 메타데이터와 상태 데이터를 선언하고 유지하는데 필요한 하부구조도 제공한다. 등록기관들은 또한 IDF와 협력하여 전체적인 DOI 시스템을 개발하는 것은 물론 활발하게 DOI 시스템의 확산을 촉진시키고 그것들의 특수한 이용자 집단을 위하여 서비스를 제공하고 있다. 최신의 RAs의 리스트는 the International DOI Foundation에 의해 관리되고 있다.

일반적으로 등록기관에서 새로운 DOI name를 할당하는 것은 무료이다: 이 비용의 일부는 IDF를 지원하는데 사용된다. 전반적으로 DOI 시스템은 IDF를 통하여 비영리적인 비용보존 방식으로 운영된다.

> Standardization

DOI 시스템은 ISO에서 개발한 국제적 표준이며, 최종 표준은 2012년 4월 23일 발표되었

다. DOI는 infoURI 스펙(IETE RFC 4452)인 “Public Namespaces에 식별자를 갖춘 정보자산용 “info” URI scheme“에 따라 등록된 URI이다; info:doi/ 는 DOI의 infoURI Namespace이다. DOI syntax는 2000년에 이미 NISO 표준이 되었다; ANSI/NISO Z39.84-2005 Syntax for the Digital Object Identifier.

1) **info:**
전산학에서 이것은 URIs로 표현되는 Library of Congress Identifiers and Digital object identifiers처럼 legacy namespaces를 허용하는 public namespaces에 있는 식별자와 함께 정보자산에 관한 a Uniform Resource Identifier (URI) scheme 이다.

It acts as a bridging mechanism for older information identifiers to be used in the more generalised and standard URI allocation.

2) a **namespace**
일반적으로 이것은 symbols, names으로 알려진 한 세트의 식별자들을 포함하고 있는 a container 이다.

Namespaces provide a level of indirection(간접적 표현) to specific identifiers, thus making it possible to distinguish between identifiers with the same exact name. For example, a surname could be thought of as a namespace that makes it possible to distinguish people who have the same first name. In computer programming, namespaces are typically employed for the purpose of grouping symbols and identifiers around a particular functionality.

p. civ

***n-gram 방식**

전산언어학 및 확률과학에서, n-gram이란 텍스트나 언어의 지정된 순서로부터 이루어진 n 개의 아이템에 대한 연속적 순서를 말한다. 조사대상인 아이템은 음소, 음절, 글자, 단어, 또는 어플에 따른 base pairs(염기쌍, 이중사슬)일 수도 있다. n-grams는 하나의 텍스트나 언어의 집성으로부터 수집된다. size 1의 n- gram은 “unigram”; size 2는 “bigram”(또는 거의 사용되지는 않지만 “digram”); size 3는 “trigram”이라고 부른다. 그리고 보다 큰 사이즈는 때때로 그 n의 값에 의해 이름이 결정 된다; 예를 들어 “four-gram”, “five-gram”, 등등.

1) n-grams for approximate matching
이것은 효율적인 근사치 매칭용으로 사용될 수 있다. 아이템의 순서를 한 세트의 n-gram으로 바꿈으로써, vector space로 처리할 수 있다. 왜냐하면 그 순서를 효율적인 방식으로 다른 순서와 비교할 수 있기 때문이다. 예를 들어, 영어 알파벳으로 된 문자들의 문자열들을 3-gram으로 변경한다면, 우리는 어떤 -차원 스페이스를 얻게 된다(첫번째 차원 “aaa”의 발생빈도를 측정하고, 두 번째는 “aab”의 발생빈도를 그리고 모든 가능한 3글자 결합을 진행한다).

Using this representation, we lose information about the string. For example, both the strings "abc" and "bca" give rise to exactly the same 2-gram "bc" (although {"ab", "bc"} is clearly not the same as {"bc", "ca"}). However, we know empirically that if two strings of real text have a similar vector representation (as measured by cosine distance) then they are likely to be similar. Other metrics have also been applied to vectors of n-grams with varying, sometimes better, results. For example z-scores have been used to compare documents by examining how many standard deviations each n-gram differs from its mean occurrence in a large collection, or text corpus, of documents (which form the

"background" vector). In the event of small counts, the g-score may give better results for comparing alternative models.

It is also possible to take a more principled approach to the statistics of n-grams, modeling similarity as the likelihood that two strings came from the same source directly in terms of a problem in **Bayesian inference**.

n-gram-based searching can also be used for ***plagiarism detection***.

2) Bayesian inference

이것은 추론해야하는 대상의 사전확률과 추가적인 관측을 통해 해당 대상의 사후 확률을 추론하는 방법이며, 또한 증거를 수집할 때 가설에 대한 확률을 갱신하는데 사용되는 Bayes' rules인 통계적 추론의 한 방법이다.

Bayesian inference is an important technique in statistics, and especially in mathematical statistics. Bayesian updating is particularly important in the dynamic analysis of a sequence of data. Bayesian inference has found application in a wide range of activities, including science, engineering, philosophy, medicine, and law. In the philosophy of decision theory, Bayesian inference is closely related to subjective probability, often called "Bayesian probability". Bayesian probability provides a rational method for updating beliefs

p. cxxi

*** Word lists by frequency**

빈도에 의한 단어 리스트는 어휘 습득을 목적으로, 어떤 차원이나 서열 리스트처럼 특정한 텍스트 집단에서 발생한 빈도를 근거로 단어들을 집단화한 리스트이다. 이 리스트는 학습자들(leaners)이 자신들의 어휘 학습 노력에 대하여 최상의 return(결과)를 확실하게 얻을 수 있다는 합리적 기초(basis)를 제공하고 있다. 그러나 이것은 주로 course writer를 목적으로 한 것이지, 학습자를 직접적으로 대상으로 한 것은 아니다. 이것의 몇 가지 주요한 단점(pitfalls)은 the corpus content, the corpus register, and the definition of "word"이다.

1) a corpus (plural corpora) or text corpus

커다란 그리고 구조적인 세트의 texts (nowadays usually electronically stored and processed)이다. They are used to do statistical analysis and hypothesis testing, checking occurrences or validating linguistic rules within a specific language territory.

반면에 단어 세기(counting)는 수천 년 된 오래된 것이지만 아직 20세기 중반까지 수작업으로 이루어지는 대규모의 분석에서도 사용하고 있으며, 예를 들어 영화 부제목(SUBTLEX megastudy)과 같은 커다란 corpus(집단)의 natural language electronic processing(자연어 처리)에서 활발하게 이루어지고 있다.

전산언어학에서, frequency list는 빈도를 가지고 단어들을 분류한 리스트이다. 여기서 빈도란 서열에서 파생될 수 있다는 의미라기보다는 일반적으로 특정한 집단에서 발생 수를 의미한다.

Type	Occurrences	Rank
the	3789654	1st
he	2098762	2nd
[...]		
king	57897	1,356th

boy	56975	1,357th
[...]		
stringyfy	5	34,589th
[...]		
transducionalify	1	123,567th

*** Ziph의 법칙:**

지프의 법칙은 자연어로 된 어떤 집단을 대상으로, 그 속에 있는 특정 단어의 빈도는 빈도 테이블에 있는 그것의 서열과는 반비례한다는 것이다. 따라서 가장 높은 빈도의 단어는 두 번째로 높은 단어의 빈도보다 약 2배 정도 많고, 세 번째 빈도 높은 단어보다는 3배가 높다는 것이다. 예를 들어, “the Brown Corpus of American English” 텍스트에서, 단어 “the”는 가장 빈도수가 높은 단어이며 전체 빈도의 약 7%를 차지하고 있다. 따라서 지프의 법칙에 의하면, 두 번째 높은 빈도의 단어 “of”는 약 3.5%가 된다.

► Theoretical review

지프법칙은 log (rank order) and log (frequency)인 축을 사용하여 a log-log graph에서 데이터를 기입(plotting)함으로써 가장 쉽게 관찰할 수 있다. For example, the word "the" (as described above) would appear at x = log(1), y = log(69971). It is also possible to plot reciprocal rank against frequency or reciprocal frequency or interword interval against rank. The data conform to Zipf's law to the extent that the plot is linear.

Formally, let:

> N be the number of elements;
> k be their rank;
> s be the value of the exponent characterizing the distribution.

Zipf's law then predicts that out of a population of N elements, the frequency of elements of rank k, f(k;s,N), is:

$$f(k;\, s,\, N) \,=\, \frac{1/k^s}{\sum_{n=1}^{N}(1/n^s)}.$$

Zipf's law holds if the number of occurrences of each element are independent and identically distributed random variables with power law distribution

$$p(f) \,=\, \alpha f^{-1-1/s}.$$

In the example of the frequency of words in the English language, N is the number of words in the English language and, if we use the classic version of Zipf's law, the exponent s is 1. f(k; s,N) will then be the fraction of the time the kth most common word occurs.

The law may also be written:

$$f(k;s,N) = \frac{1}{k^s H_{N,s}}$$

where HN,s is the Nth generalized harmonic number.

The simplest case of Zipf's law is a "1/f function". Given a set of Zipfian distributed frequencies, sorted from most common to least common, the second most common frequency will occur ½ as often as the first. The third most common frequency will occur ⅓ as often as the first. The nth most common frequency will occur 1/n as often as the first. However, this cannot hold exactly, because items must occur an integer number of times; there cannot be 2.5 occurrences of a word. Nevertheless, over fairly wide ranges, and to a fairly good approximation, many natural phenomena obey Zipf's law.

Mathematically, the sum of all relative frequencies in a Zipf distribution is equal to the harmonic series, and

$$\sum_{n=1}^{\infty} \frac{1}{n} = \infty.$$

In human languages, word frequencies have a very heavy-tailed distribution, and can therefore be modeled reasonably well by a Zipf distribution with an s close to 1.

As long as the exponent s exceeds 1, it is possible for such a law to hold with infinitely many words, since if s > 1 then

$$\zeta(s) = \sum_{n=1}^{\infty} \frac{1}{n^s} < \infty.$$

where ζ is Riemann's zeta function.

► Ziph의 제 1 법칙

이것은 텍스트의 단어들을 출현빈도순으로 배열한 다음 각각의 순위를 매기고, 그것들의 출현빈도와 순위를 곱하면 그 값들이 일정하다는 법칙이다. 한 가지 주의할 것은 이것은 고빈도 단어에만 적용되며, 저빈도 단어에는 적용되지 않는다는 것이다.

예

단어	순위(r)	출현빈도(f)	값(r *x* f)
the	1	301	301
of	2	152	304
for	3	108	324
to	4	81	324
and	5	68	340

► Ziph의 제 2 법칙: Booth(A.D. Booth)가 수정한 법칙이다.

텍스트에 한번만 출현한 단어의 수와 n번 출현한 단어의 수의 비율은 텍스트와 상관없이 일정하다는 법칙이며, 저빈도 단어에만 적용된다.

► 최소노력의 법칙(Principle of Least Effort)

최소노력의 원칙은 진화생물학에서부터 웹페이지 디자인까지 다양한 분야에서 다루어지는 포괄적인 이론이다. 이것은 동물, 사람, 심지어 잘 디자인된 기계조차도 최소한의 저항이나 노력을 선택한다는 것이 자연스러울 것이라고 가정한다.

문헌정보학과 관련해서, 이 원칙에 따라 정보를 획득하려는 이용자는 이용할 수 있고 최소한의 정확성을 갖는 방법이라도 가장 편리한 탐색방법을 사용하려는 경향이 있다는 것이며, 이 같은 정보입수행위는 최소한의 수용 가능한 결과를 얻자마자 중지된다는 것이다.

*** Bradford's law**

1934년 Samuel C. Bradford에 의해 처음으로 주장된 하나의 패턴이며, 이것은 과학학술지에서 참고문헌의 수에 대한 탐색을 확대해 보면, 점점 더 해당 참고문헌의 수가 기하급수적으로 줄어든다는 것이다. 이것의 공식은 만일 특정 분야의 학술지가 그것의 모든 기사의 수를 약 1/3씩을 가지고 있는 세 개의 집단으로 분류해 놓고 보면, 각 집단에 있는 학술지의 수의 비율은 $1:n:n^2$ 이 된다는 것이다.

많은 학분 분야에서 이러한 패턴을 Pareto 분산이라 부른다. 예를 들어, 한 연구자가 자신의 연구주제를 위하여 5가지의 핵심 저널을 갖고 있고, 이 저널들에서 12개의 기사가 관심대상이라고 가정해 보자. 그리고 추가로 이 연구자가 또 다른 12개의 관심 기사를 찾기 위해서 그는 추가로 10개의 저널을 봐야한다고 가정해 보자. 그러면 이 연구자의 브래포드 승수 bm(Bradford multiplier)는 2가 된다: 다시 말해서, 10/5이다. 새로운 12개의 기사를 추가로 이용하기 위하여 이 연구자는 bm의 배만큼 많은 저널을 봐야할 것이다. 즉, 5, 10, 20, 40, 등의 저널을 봐야할 것으려, 이럴 경우에 대부분의 연구자들은 신속하게 “there is little point in looking further.”라는 것을 깨닫는다.

이러한 결과는 최상의 저널을 출판하려는 과학자들과 핵심 저널에 접근을 보장하려는 대학교에 대한 압박감을 가중시키고 있다. 또 한편으로 핵심저널에 대한 판단을 개개의 연구자들 마다 다르며, 심지어 학파(chools-of-thought divides)에도 심각하게 차이가 난다. 또한 이러한 성향에 따라 선택된다면, 그 저널은 지나치제 주류학파의 견해를 반영하는 위험을 갖게 된다.

브래포드의 법칙은 Bradford's law of scattering 그리고 the Bradford distribution으로 알려져 있으며, 계량서지학에서의 이 법칙은 웹에 적용되고 있다.

*** The Pareto principle(the 80-20 rule, the law of the vital few, and the principle of factor sparsity)**

이 원칙은 80-20 규칙으로도 잘 알려져 있으며, 많은 사건에 있어서 결과의 약 80%가 약 20%의 원인으로부터 발생한다는 것이다.

1906년 이태리 경제학자 Vilfredo Pareto는 이태리 국토의 80%를 인구의 약 20%인 지주들이 소유하고 있다는 것을 관찰하였으며, 또한 자신의 정원에서 20%의 콩깍지가 80%의 콩알을 포함하고 있다는 것을 관찰하여 이 원칙을 개발하였다.

*** Luhn의 가설**

Hans Peter Luhn (July 1, 1896 – August 19, 1964)은 IBM의 컴퓨터 과학자였으며, the Luhn algorithm과 KWIC (Key Words In Context) indexing의 개발자이며, 그의 가장 위대한 업적들 중 두 가지는 an SDI system과 the KWIC method of indexing에 대한 아이디어 이다.

"고빈도의 단어는 너무 일반적인 단어이므로 주제어로서의 가치가 없어 정확률이 떨어진다. 또한 저빈도의 단어도 주제어로서의 의미가 없어서 재현율을 떨어뜨린다. 따라서 중간빈도의 단어를 색인어로 선정해야 한다."

 # KWIC

KWIC이란 Key Word In Context의 두문자어이며, 용어색인(concordance) lines용 포맷으로 가장 일반적으로 사용되고 있다. 이 시스템은 Andrea Crestadoro에 이해 1864년에 맨테스터 도서관에서 가장 먼저 제한한 keyword in titles이라 부르는 개념에 근거하고 있다.

KWIC index은 학술기사의 서명에 있는 각 단어들(stop words 제외)을 대상으로 그 색인에서 알파벳 순으로 탐색할 수 있도록 분류하고 배열함으로써 만들어진다(form).
이것은 전산화된 풀 텍스트 탐색이 일반화되기 전까지는 기술 매뉴얼을 위한 유익한 색인방법이었다. 예를 들어, 지금 이 기사의 서명 진술(statement)과 위키피디아 슬로건은 KWIC 색인

에서는 다음과 같을 것이다. KWIC index은 대체로 다양한 모습으로 사용되어 '문맥' 에 있는 정보를 최대한 디스플레이하도록 한다:

KWIC is an **acronym** for Key Word In Context, ...	page 1
... Key Word In Context, the most **common** format for concordance lines.	page 1
... the most common format for **concordance** lines.	page 0
... In Context, the most common **format** for concordance lines.	page 1
Wikipedia, The **Free** Encyclopedia	page 0
KWIC is an acronym for **Key** Word In Context, the most ...	page 1
KWIC is an acronym for Key Word ...	page 1
common format for concordance **lines**.	page 1
... for Key Word In Context, the **most** common format for concordance ...	page 1
Wikipedia, The Free Encyclopedia	page 0
KWIC is an acronym for Key **Word** In Context, the most common ...	page 1

permuted index(순열색인)은 표목의 모든 순환적 순열(all cyclic permutations of the headings)을 색인한다는 의미에서 사실상 KWIC index의 또 다른 이름이다. 책은 표목을 갖춘 여러 개의 짧은 부분(sections)으로 구성되며, 유명한 매뉴얼 페이지들은 종종 순열색인 부분에서 끝난다. 따라서 독자들이 쉽게 그것의 표목에 있는 특정 단어를 통해 필요한 섹션을 찾을 수 있다. 이 같은 업무는 KWOC ("Key Word Out of Context")으로 알려져 있으며 더 이상은 일반적인(common) 것이 못되고 있다.

1) A **concordance**
이것은 책이나 저작에서 사용된 중요한 단어들의 문맥과 함께 알파벳으로 열거해 놓은 리스트이다. 컴퓨터 이전시기에는 이것을 만드는데 필요한 시간, 어려움, 비용으로 인하여, the Vedas, Bible, Qur'an or the works of Shakespeare and other classical Latin and Greek authors와 같은 특별히 중요한 저작들만이 자신의 용어색인을 갖고 있다.

2) **Text mining(**text data mining, roughly equivalent to text analytics
이것은 텍스트로부터 양질(high quality)의 정보를 추출해내는 방법이다. 양질의 정보는 전형적으로 statistical pattern learning과 같은 수단을 사용하여 patterns and trends를 devising함으로써 얻어진다. 이것은 일반적으로 입력 텍스트를 조직화하는 과정(DB에 추출한 언어적 특징을 추가, 제거, 삽입하는 parsing 과정), 조직화된 데이터로부터 패턴을 추출하는 과정, 그리고 마지막으로 결과를 평가하고 해석하는 과정을 갖고 있다.

여기서 'High quality'란 some combination of relevance, novelty, and interestingness을 말하며, 전형적으로 이것의 임무에는 text categorization, text clustering, concept/entity extraction, production of granular taxonomies, sentiment analysis, document summarization, 그리고 entity relation modeling (i.e., learning relations between named entities) 등이 포함되어 있다.

Text analysis involves information retrieval, lexical analysis to study word frequency distributions, pattern recognition, tagging/annotation, information extraction, data mining techniques including link and association analysis, visualization, and predictive analytics. The overarching goal is, essentially, to turn text into data for analysis, via application of natural language processing (NLP) and analytical methods.

A typical application is to scan a set of documents written in a natural language and either model the document set for predictive classification purposes or populate a database or search index with the information extracted.

*** Lotka's law,**

이것은 Alfred J. Lotka가 개발한 법칙이며, 지프 법칙의 여러 가지 특별한 응용 법칙들

중의 하나이다. 이것은 특정분야의 저자가 출판한 저작물의 빈도를 설명하고 있다. 다시 말해서, 특정분야에서 n개의 저작물을 출판한 저자의 수는 약 $1/n^a$ 이다. 이 공식에서 n은 1편의 저작물을 출판한 저자의 수를 나타내며, 거의 항상 a의 지수는 2 이다.

More plainly, the number of authors publishing a certain number of articles is a fixed ratio to the number of authors publishing a single article. As the number of articles published increases, authors producing that many publications become less frequent. There are 1/4 as many authors publishing two articles within a specified time period as there are single-publication authors, 1/9 as many publishing three articles, 1/16 as many publishing four articles, etc. Though the law itself covers many disciplines, the actual ratios involved (as a function of 'a') are very discipline-specific.

The general formula says:

$$X^nY = C \quad \text{or} \quad Y = C / X^n,$$

where X is the number of publications, Y the relative frequency of authors with X publications, and n and C are constants depending on the specific field ($n \approx 2$).

*** Moore's law**

이것은 관찰법으로, 컴퓨터 하드웨어의 역사적 관찰을 통해 집적회로의 트랜지스터의 수가 매 2년마다 약 2배가 증가한다는 법칙이다.

the 2010 update to the International Technology Roadmap for Semiconductors predicts that growth will slow at the end of 2013, when transistor counts and densities are to double only every three years.

*** 1% rule**

인터넷 문화에서, 1%의 규칙이란 인터넷 커뮤니티에 참가하는 것과 관련된 경험의 법칙이며, 웹사이트의 단지 1%의 이용자만이 활발하게 새로운 콘텐트를 만드는 반면에 나머지 99%는 단지 이용만 한다(lurk)는 것이다.

A variant is the "90-9-1 principle" (sometimes also presented as the 89:10:1 ratio), which states that in a collaborative website such as a wiki, 90% of the participants of a community only view content, 9% of the participants edit content, and 1% of the participants actively create new content.

Both can be compared with the similar rules known to information science, such as the 80/20 rule known as the Pareto principle, that 20 percent of a group

will produce 80 percent of the activity.

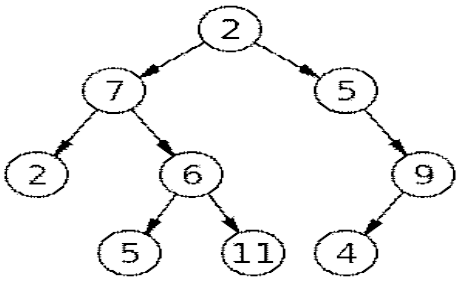
p. cxxvi

(이미지자료의 내용기반색인을 위한 대표적인 기법)

*** tree(data structure)**

트리(tree)란 abstract data type (ADT) 또는 한 무리의 linked nodes로 표현되며, 부모(root)인 value 와 자식인 subtrees의 계층적 트리 구조를 모방하고 있는 ADT와 같은 데이터 구조의 설정에 폭넓게 사용되고 있다. 트리 데이터 구조는 부분적으로 보면 하나의 노드 집단(루트 노드에서 출발하는)으로 귀납적으로 정의할 수 있으며, 또한 이것은 각 노드마다 자신의 값과 더불어 하위 노드들(the children)에 대한 레퍼런스의 리스트로 이루어진 데이터 구조이다. 그렇지만 이 노드들은 레퍼런스의 중복이 없다면 어떤 것도 루트로 연결되지 못한다는 문제점을 가지고 있다. 이 문제에 대한 대안으로 트리는 하나의 질서 있는 트리로 전체적인 맥락에서 추상적으로 정의될 수도 있는데, 이런 경우에는 각각의 노드는 할당된 값을 갖고 있어야 한다. 예를 들어, 전체적으로 트리를 살펴보면, 누구나 특정한 노드의 "parent node"를 말한 수 있지만, 일반적으로 데이터 구조로서 특정한 노드는 단지 자식의 리스트만을 포함하고 있지 부모에 대한 레퍼런스는 포함하지 않는다.

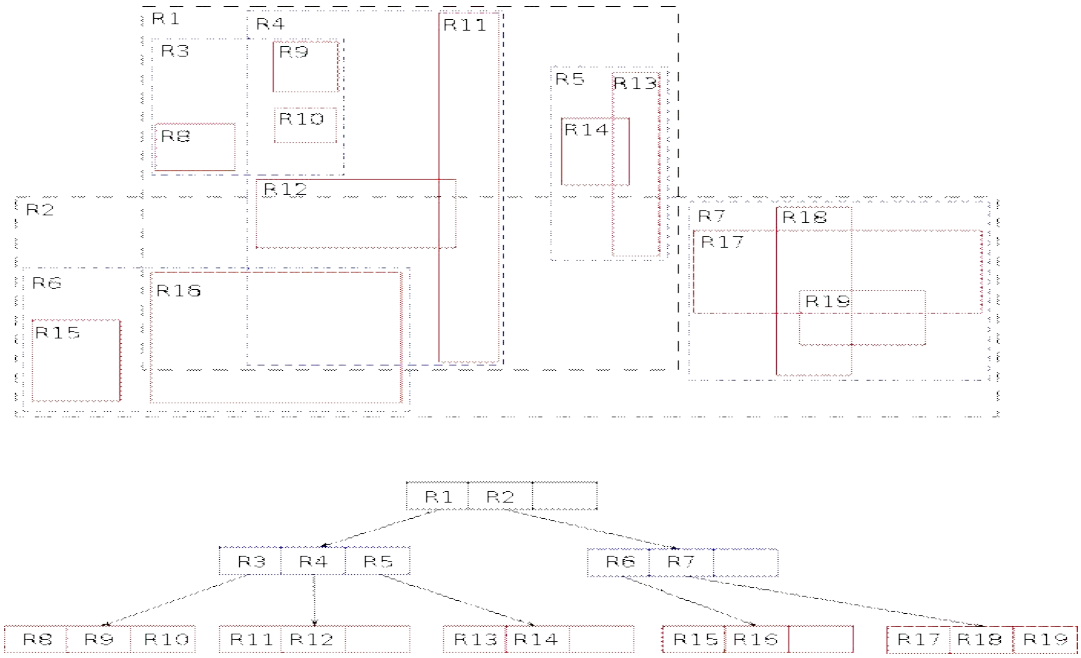
Binary tree



*** R-tree 기법:** 공간 데이터나 다차원 데이터의 효율적인 질의 처리를 위한 색인 구조.

R-trees는 공간적 접근을 위한 트리 데이터 구조이다. 다시 말해서, geographical coordinates, rectangles or polygons(다각형)과 같은 다차원 정보를 색인하는데 사용된다. R-tree의 일반적인 실생활 용도는 레스토랑 위치나 전형적인 지도로 이루어진 streets, buildings, lakes, coastlines, etc.의 다각형과 같은 공간적 객체를 저장하여, 예를 들어, "현 위치에서 2km내에 있는 모든 박물관을 차자라" 라는 질문에 신속하게 응답하는 것이다.

R-tree



*** B-tree**

B-tree는 분류된 데이터를 보관하고, logarithmic(대수적) time으로 탐색, 순차적 접근, 입력, 삭제가 가능한 트리 데이터 구조이다. B-tree는 하나의 노드가 2개 이상의 children을 가질 수 있는 이진탐색 트리를 일반화한 것이다. 스스로 균형을 유지하는 이진 탐색 트리와 달리, B-tree는 커다란 블록의 데이터를 읽고 쓰는 시스템을 최적화시킨다. 일반적으로 이것은 데이터베이스 및 파일 시스템에서 사용한다.

예를 들어, 2-3 B-tree (often simply referred to as a 2-3 tree)에서, 각각의 내적 노드는 단지 2 또는 3개의 자식 노드만을 갖는다. B-tree의 각각의 내적 노드는 많은 keys를 포함할 수 있다. 이 키들은 그것의 하위트리를 나누는 분리 값(separation values)으로 사용된다. 예를 들어, 만일 한 내적 노드가 3개의 자식 노드나 하위트리를 갖는다면 그것은 2 keys를 가져야만 한다: a_1과 a_2. 가장 왼쪽에 있는 모든 값들은 a_1보다 적을 것이며, 중간의 하위 트리의 모든 값은 a_1과 a_2의 사이에 있을 것이고, 가장 오른쪽에 있는 하위트리의 모든 값들은 a_2보다 클 것이다.

1) binary search tree (BST)

이것은 때때로 an ordered or sorted binary tree라고도 부르며, 다음과 같은 성질을 갖고 있는 a node-based binary tree data structure 이다:

The left subtree of a node contains only nodes with keys less than the node's key.
The right subtree of a node contains only nodes with keys greater than the node's key.
The left and right subtree each must also be a binary search tree.
There must be no duplicate nodes.

Generally, the information represented by each node is a record rather than a single data element.

일반적으로 말해서, 각 노드에 의해 표현된 정보는 하나의 단일 데이터 요소라기 보다는 하나의 레코드이다. 그렇지만, 순차적이라는 목적에 따라, 노드들은 관련된 레코드들보다는 자신들의 keys에 따라 비교된다.

다른 데이터 구조보다 2진 탐색 트리의 주요한 장점은 in-order traversal(횡단)과 같은 the related sorting algorithms and search algorithms이 매우 효율적이라는 것이다. 2진 탐색 트리는 sets, multisets, and associative arrays와 같은 보다 추상적인(abstract) 데이터 구조를 구축하기 위하여 사용되는 기본적인 데이터 구조이다.

*** TV(Telescopic Vector)-tree 기법**

이 기법은 매우 커다란 차원의 공간에 있는 데이터에 매우 효율적으로 접근하기 위한 방법이며, R-tree로부터 빌려온 데이터 구조를 사용한다. 또한 이것은 조사 대상인 데이터를 근거로 branch하는 방법을 역학적이고 융통성 있게 결정하도록 한다. 만일 모든 벡터가 어떤 값을 갖기로 동의한다면(예를 들어, 모든 다큐멘트가 많은 공동의 용어를 갖고 있다면), 우리는 벡터와 다큐멘트를 구별하는 이들 용어들(다시 말해서, 벡터의 필드들인)을 기준으로 branching하여 색인을 만들어야 한다.

*** SS-tree 기법**

SS-트리(Similarity Search tree)는 유사도 평가 척도를 사용하며, 질의와 이미지 데이터의 유사성을 비교하여 유사도가 높은 이미지 데이터를 검색하는데 적합하도록 설계된 동적인 색인 구조이다. R-트리 계열의 색인 구조가 데이터 공간을 MBR(Minimum Rectangle Region)로 분할하는 반면, SS-트리는 구 영역(spherical region)으로 데이터 공간을 분할한다. 이 기법은 유사성에 근거함으로 검색을 용이하게 하는 반면 중첩 영역이 많이 발생함으로써 검색성능을 저하시키다는 단점을 가지고 있다.

*** X-tree 기법**

X-tree는 여러 차원에 존재하는 데이터를 저장하기 위하여 사용된 R-tree를 근거로 작성된 색인 트리 구조이다. 이 기법과 R-trees, R^+-trees and R^*-trees와의 차이는 고차원으로 갈수록 점점 더 문제가 발생하여 서로 연결된 박스들 간의 overlap을 예방할 수 있다는 점이다. 따라서 이 기법의 노드들이 중복을 예방하기 위하여 더 이상 분할되지 못한다면, 그 노드는 결과적으로 슈퍼 노드가 된다. 극단적인 경우에, 다른 데이터 구조에서 관찰된 최악의 경우에 발생하는 행위를 방어하기 위하여 그 트리는 선형화되기도 한다.

1) **R^*-trees**

이것은 공간정보를 색인하기 위한 R-trees의 일종이며, 다른 R-trees보다 비용이 좀 더 들지만, 동시에 point와 spatial data를 지원하는 장점을 가지고 있다.

2) An **R^+ tree**

이것은 종종 (x, y) coordinates로 위치를 사용하는 데이터를 찾는데 사용되는 방법이며, 가끔 지상의 위치를 찾는 용도로 사용되기도 한다. 기본적으로, R^+ tree는 a tree data structure이며, 공간정보를 색인하는데 사용하는 R tree의 일종이다.

** 대표적인 검색엔진; **QBIC(Query By Image Content)** 프로젝트, IBM Almaden 연구소의 프로젝트

query by image content (QBIC)으로 알려진 Content-based image retrieval (CBIR)와 content-based visual information retrieval (CBVIR)는 대규모 데이터베이스에서 디지털 이미지를 찾는데 나타나는 이미지 검색 문제를 해결하기 위한 computer vision techniques의 어플 이다. 따라서 Content-based image retrieval과 concept-based approaches은 서로 반대적 이다.

여기서 "Content-based"란 의미는 이미지와 결합되어 있는 keywords, tags, or descriptions와 같은 메타데이터보다는 그것의 이미지의 콘텐트를 분석하는 탐색을 의미하며, 이런 맥락에서 "content"란 용어는 그 이미지의 colors, shapes, textures, or any other information를 말하기도 한다. CBIR이 바람직한 이유는 most web-based image search engines이 순전히 메타데이터에 의존하고 있으므로 결과적으로 많은 garbage를 생산하기 때문이다. 또한 수작업으로 데이터베이스에 있는 이미지용 키워드를 입력하는 것은 비효율적이고 비경제적이며, 또한 그 이미지가 묘사하고 있는 모든 키워드를 잡아내지 못할 수도 있다. 따라서 자신의 콘텐트들을 근거로 이미지를 filter할 수 있는 시스템은 보다 우수한 색인을 제공할 것이며, 그 결과는 더욱 정확할 것이다.

(오디오 자료의 내용기반색인)

* **Query by Humming (QbH)**: 1995년 코넬대학의 Query By Humming(QBH)

이것은 title, artist, composer, and genre에 대한 원 분류 시스템에서 파생된 음악 검색 시스템이다. 명확하게 구분되는 단일 테마나 멜로디를 갖춘 노래와 음악에 이것을 적용하는 것이 정상적이다. 이것은 a user-hummed melody (input query)를 잡아서 기존의 데이터베이스에 그것을 비교하는 것을 가능하게 한다. 그런 다음에 이것은 그 input query와 가장 밀접한 음악의 리스트를 서열화 시켜 보여준다.

* **MELDEX**: 1997년 뉴질랜드의 Waikato 대학의 MELody in Dex(MELDEX)

이것은 the New Zealand Digital Library's Web-based melody index 시스템 이며, 소수의 음표(a few notes)를 근거로 데이터베이스로부터 melodies를 검색하여 마이크로 들려주도록 디자인되었다.

*** (자체적으로는 인용색인을 생성하지 않고 "Cited By" 기능을 통해 인용색인 데이터베이스에 링크하는 데이터베이스의 예):**

> ScienceDirect

이것은 Anglo-Dutch publisher Elsevier에서 운영하는 웹사이트로, 2013년을 기준으로 about 11 million articles from 2,500 journals and over 25,000 e-books, reference works, book series and handbooks을 소장하고 있다.

이것의 학술기사들은 4개의 주요 분야로 집단화 되어 있다: Physical Sciences and Engineering, Life Sciences, Health Sciences, and Social Sciences and Humanities. 그리고 이것의 대부분의 웹 사이트 기사들은 무료로 초록을 이용할 수 있으며, 구독예약이나 pay-per-view purchase를 통해 PDF 또는 HTML로 전문에 접근할 수 있다.

> SAGE

이것은 journals, books, and electronic media for academic, educational, and professional markets 분야의 국제적이고 선도적인 출판사이다.

1965년부터, SAGE은 business, humanities, social sciences, and science, technology, and medicine와 같은 분야의 scholars, practitioners, researchers, and students의 집단에게 정보를 제공하고 있다.

> PMC

이것은 the U.S. National Institutes of Health's National Library of Medicine (NIH/NLM)에 있는 biomedical and life sciences journal literature의 a free full-text archive이다.

*** Bibliographic Database**

이 DB는 journal과 newspaper articles, conference proceedings, reports, government and legal publications, patents, books 등을 포함하여 기존에 출판된 문헌을 조직화해 놓은 디지털 컬렉션의 레퍼런스들이다. 도서관 목록 엔트리와는 대조적으로, 서지 데이터베이스에 있는 서지 레코드의 대부분은 완전한 단행본보다는 기사, 회의자료 등을 나타내며, 또한 일반적으로 이것들은 keywords, subject classification terms, or abstracts을 통하여 매우 풍부하게 관련 주제를 표현하고 있다.

서지 데이터베이스는 범위가 일반적인 것뿐만 아니라 특수한 학문 분야도 다루고 있지만, 대부분의 서지 데이터베이스는 현재에도 벤더로부터 또는 그것들을 만든 색인 및 초록 서비스로부터 직접적으로 라이센스를 얻어야만 사용할 수 있는 전매품들이다.

그리고 많은 서지 데이터베이스가 디지털 도서관으로 진화하여 색인된 콘텐트를 통하여

전문을 제공하고 있다. 또 한편으로 비-서지적 학술 데이터베이스로 전환한 것들은 Chemical Abstracts나 Entrez와 같은 보다 완벽한 학술탐색 엔진 시스템으로 발전하였다.

1) The **Entrez(**Global Query Cross-Database Search System)
이것은 강력한 federated search engine, or web portal이며, 이용자로 하여금 the National Center for Biotechnology Information (NCBI) website에 있는 많은 health sciences databases를 탐색할 수 있도록 한다.

The NCBI is a part of the National Library of Medicine (NLM), which is itself a department of the National Institutes of Health (NIH), which in turn is a part of the United States Department of Health and Human Services.

"Entrez" (a greeting meaning "Come in!" in French)이름은 대중이 NLM에서 이용 가능한 컨텐트를 탐색하는 것을 환영한다는 정신을 반영하고 있다.

*** Document-oriented database**

이 DB는 semi-structured data로 잘 알려진 다큐멘트-지향적 정보를 저장, 검색, 관리하도록 디자인된 컴퓨터 프로그램이다. 이것은 소위 NoSQL 데이터베이스의 주요 카테고리의 하나이며, "document-oriented database" (or "document store")라는 용어는 NoSQL을 사용하면서 그 인기가 커졌다. 그러나 이것은 관계형 데이터베이스의 “Relations” (또는 “Tables”)에 대한 개념과는 대조적인 “Document”라는 추상적 개념을 가지고 디자인한다는 특징을 가지고 있다.

> Documents?

다큐멘트-지향적 데이터베이스의 핵심 개념은 다큐멘트라는 개념이다. 각 다큐멘트-지향적 데이터베이스의 설치가 이것에 대한 정의의 내용에 따라 다르지만, 일반적으로 이것들 모두는 다큐멘트들이 어떤 표준 포맷이나 암호화기법에 의해 데이터(또는 정보)를 감싸서 암호화한다는 것을 가정하고 있다. 이때 사용하는 암호화기법으로는 XML, YAML, JSON, and BSON 뿐만 아니라, PDF and Microsoft Office documents (MS Word, Excel, and so on)와 같은 2진 forms도 있다.

1) **YAML** (/ˈjæməl/, rhymes with camel)
이것은 C, Perl, and Python과 같은 programming languages와 같은 프로그래밍 언어로부터 개념을, 그리고 XML and the data format of electronic mail (RFC 2822)로부터 아이디어를 가져와서 만든 a human-readable data serialization format 이다.

YAML is a recursive acronym for "YAML Ain't Markup Language". Early in its development, YAML was said to mean "Yet Another Markup Language", but it was then reinterpreted (backronyming the original acronym) to distinguish its purpose as data-oriented, rather than document markup.

2) **JSON** (/ˈdʒeɪsɒn/ JAY-soun, /ˈdʒeɪsən/ JAY-son), or JavaScript Object Notation,
이것은 사람이 읽을 수 있는 텍스트를 이용하여 attribute-value pairs로 된 데이터 objects를 전송하기 위한 an open standard format 이다. 따라서 이것은 XML의 대안으로 서버와 웹 어플간의 데이터 전송에 기본적으로 사용된다.

3) **BSON** (/ˈbiːsɒn/)
이것은 a computer data interchange format이며, 주로 the MongoDB database에서 a data storage and network transfer format으로 사용된다. 또한 이것은 simple data structures and associative arrays (called objects or documents in MongoDB)를 표현하기 위한 2진 form을이며, "BSON" 라는 이름은 JSON을 근거로 하

고 있으며, "Binary JSON"을 의미한다.

다큐멘트-지향적 데이터베이스에 있는 다큐멘트들은 관계형 데이터베이스의 레코드나 로우와는 몇 가지에서 비슷하지만, 엄격성이 좀 떨어진다는 차이가 있다. 다시 말해서, 이것들은 하나의 표준 스키마에 집착하지 않으며, 또한 이것들 모두가 동일한 sections, slots, parts, or keys를 갖는 것은 아니다.

다음의 다큐멘트 예를 살펴보자:

```
{
    FirstName: "Bob",
    Address: "5 Oak St.",
    Hobby: "sailing"
}
```

두 번째 다큐멘트는 다음과 같을 수 있다:

```
{
  FirstName: "Jonathan",
  Address: "15 Wanamassa Point Road",
  Children: [
        {Name: "Michael", Age: 10},
        {Name: "Jennifer", Age: 8},
        {Name: "Samantha", Age: 5},
        {Name: "Elena", Age: 2}
  ]
  }
```

위의 2 다큐멘트들은 서로서로 몇 가지의 구조적 요소를 공유하고 있지만, 각각은 또한 유일한 요소들을 가지고 있다. 모든 레코드가 사용하지 않은 필드를 빈칸으로 남겨 놓으면서 동일한 필드를 갖질 수 있는 관계형 데이터베이스와 달리, 위의 예에 있는 어떠한 다큐멘트(레코드)에서도 빈 필드는 존재하지 않는다. 이러한 방법은 새로운 정보가 그 데이터베이스에 있는 모든 다른 레코드와 동일한 구조를 공유할 것을 요구하지 않음으로써 어떠한 레코드도 추가될 수 있다는 장점을 가지고 있다.

> Keys

다큐멘트들은 자신을 대표하는 유일한 키를 통해 데이터베이스에 자리를 잡는다. 이 키는 종종 a simple string, a URI, or a path 이며, 데이터베이스로부터 다큐멘트를 검색하는데 사용될 수 있다. 전형적으로 데이터베이스는 다큐멘트 검색의 속도를 높이기 위하여 키로 된 색인을 가지고 있다.

> Retrieval

다큐멘트-지형적 데이터베이스의 또다른 분명한 특징은 다큐멘트를 검색하는데 사용될 수 있는 간단한 key-document (or key-value) lookup(검색) 이외에도, 데이터베이스가 이용자

로 하여금 컨텐트를 근거로 다큐멘트를 검색할 수 있도록 an API or query language를 제공하는 것이다. 예를 들어, 여러분은 어떤 값이 설정된 어떤 필드와 더불어 모든 다큐멘트를 검색하는 쿼리를 원할 수도 있지만, 이용 가능한 query APIs or query language의 기능들 뿐만 아니라 쿼리의 예상된 성능은 실행할 때마다 분명하게 차이가 난다.

*** Citation Index**

인용색인은 이용자로 하여금 나중의 생산된 문서가 앞서 생산된 어떤 문서를 인용했는지를 쉽게 알 수 있도록 만든 일종의 서지 데이터베이스이다.

인용색인의 형태는 12세기 히브리어 종교문헌에서 처음 발견되었으며, 법률적 인용색인은 18세기에 발견되었고, Shepard's Citations (1873)과 같은 인용자(citators)에 의해 인기를 끌었다. 1960년에, Eugene Garfield's Institute for Scientific Information (ISI)가 학술지 기사용으로 최초의 인용색인을 소개하였다.: first the **Science Citation Index (SCI)**, and later the **Social Sciences Citation Index (SSCI)** and **the Arts and Humanities Citation Index (AHCI)**. 그리고 첫 번째 자동화된 인용색인은 1997년에 CiteSeer에 의해 이루어졌으며, 이러한 데이터의 또 다른 정보원은 Google Scholar 이다.

> Major citation indexing services; 범용이면서 학술적인 인용색인 서비스

ISI (now part of Thomson Reuters)
인쇄판과 CD로 **the ISI citation indexes**를 출판하고 있으며, 이것들은 일반적으로 지금은 Web of Science라는 이름으로 웹을 통해 접근할 수 있다. 그리고 Web of Science는 the Web of Knowledge의 데이터베이스 그룹의 일부이다.

Elsevier
자연과학과 사회과학분야에서 주제탐색과 citation browsing and tracking을 유사하게 결합하여 온라인으로 이용할 수 있는 **Scopus**를 출판하고 있다.

 # Indian Citation Index
인도에서 출판된 peer reviewed journals을 취급하고 있는 an online citation data이다. 이것의 주요 주제는 scientific, technical, medical, and social sciences and arts and humanities이며, 인도 최초의 citation database 이다.

the ISI databases와 Scopus는 구독예약을 통해 이용 가능하지만, CiteSeer와 Google Scholar는 온라인으로 무료로 사용할 수 있다.

> Impact factor(IF)

이것은 학술지에 실린 최신의 기사들에 대한 평균인용의 수를 반영하는 척도이다. 이것은 종종 특정 학술지의 상대적 중요성을 가늠하는 proxy(대용물)로 사용되기도 하며, 높은 IF의 학술지는 낮은 IF의 학술지보다 더 중요한 학술지로 여겨진다. 이 IF는 Eugene Garfield(the founder of the Institute for Scientific Information)에 의해 만들어졌으며, the Journal

Citation Reports에 색인된 학술지를 대상으로 1975년부터 매년 계산되고 있다.

> Citation impac(CI)

CI는 여러 가지 방법으로 측정될 수 있지만, 한 가지 분명한 것은 인용된 저작물의 the usage and impact 둘 다를 계량화한 citation count이다. 이것을 citation analysis or bibliometrics라 부른다. 인용분석으로 얻은 여러 가지 척도 중에서 citation counts는 다음과 같은 것을 대상으로 한다:

an individual article (얼마나 자주 인용되는가?);
an author (기사별 전체 인용 수 또는 평균인용 수);
a journal (학술지의 기사별 평균 인용 수).

개인별 학자의 CI를 보다 잘 계량화하기 위하여 간단한 citation counts 이외에도 많은 척도들이 제안되었다. 가장 잘 알려진 척도는 the h-index와 the g-index 이며, 이것들은 편견(호감)에서부터 인용데이터 정보원의 discipline-dependence and limitations에 이르기까지 서로서로 장단점을 가지고 있다.

1) The h-index

이것은 학자의 출판물에 대한 생산성과 impact(영향력) 둘 다를 측정하려는 색인이며, 이론물리학자의 상대적 quality를 결정하는 도구로서 Jorge E. Hirsch(a physicist at UCSD)에 의해 제안되었다. 따라서 때때로 Hirsch index or Hirsch number라고 부르기도 한다.이 색인은 학자의 가장 많이 인용된 논문과 다른 출판물에서 인용된 수의 집합을 근거로 하고 있다. 또한 이 색인은 학과, 대학, 국가뿐만 아니라 학술지와 같은 학자 집단에 대한 생산성과 impact에 적용될 수 있다.

2) The g-index

이것은 출판 레코드를 근거로 과학적 생산성을 계량화하기 위한 색인이며, 2006년에 Leo Egghe에 의해 제안되었다 이 색인은 특정한 연구자의 출판물에 대한 인용의 distribution을 근거로 계산한다: 인용된 수에 따라 내림차순으로 서열화 된 한 무리의 기사가 주어졌을 때, 상위의 g 기사들은 적어도 g^2 citations을 갖게 되므로, g-index는 유일한 최대의 수이다.

h-index와 마찬가지로, g-index는 두 가지의 서로 다른 수량용으로 사용되지만 같은 number이다: g는 (1) 많이 인용된 기사의 수, 그리고 그것들 각 기사는 (2) 평균적으로 g citations을 갖는다.

> Eigenfactor score

이것은 University of Washington의 Jevin West와 Carl Bergstrom에 의해 개발되었으며, 과학지의 전체적 중요성을 나타내는 rating(등급) 이다. 학술지들은 낮은 등급의 학술지보다 eigenfactor에 더 커다란 영향을 끼치도록 가중치를 준 높은 등급의 학술지로부터 나온 citations과 함께 incoming citations의 수에 따라 등급이 결정된다. 중요도의 척도로서, Eigenfactor score는 어떤 학술지의 전체적인 impact를 측정하므로, 특정분야에서 보다 높은 impact를 생산하는 학술지는 보다 커다란 Eigenfactor scores를 갖는다.

Eigenfactor scores and Article Influence scores는 eigenfactor.org에서 계산되고 있으며, 무료로 볼 수 있다. The Eigenfactor score의 목적은 the origin of the incoming

citations을 고려하여 과학집단의 학술지의 중요성을 측정하는 것이며, 평균적 연구자가 얼마나 자주 그 학술지의 컨텐트에 접근할 수 있는지를 반영하는 것으로 여겨지고 있다. 그렇지만 the Eigenfactor score는 the size of the journal에 영향을 받으므로, 해당 저널의 규모(measured as published articles per year)가 두 배가 되면 그 점수도 두 배가 된다. The Article Influence score는 학술지의 기사에 대한 평균 influence를 측정하므로 the ISI impact factor에 필적한다.

또한 Eigenfactor scores는 개인별 학자의 저작을 평가하기 위하여 H-index와 결합하여 사용되기도 한다.

*** Journal Citation Reports**

이것은 연간출판물이며, the Science and Scholarly Research division of Thomson Reuters에서 생산하고 있고, the Web of Science와 통합되었으며, the Web of Science-Core Collections에서 접근할 수 있다. 이것은 또한 IF를 포함하고 있는 자연과학과 사회과학의 학술지에 대한 정보를 제공하고 있다. JCR은 원래 Science Citation Index의 일부로 출판되었으나, 현재에는 독립된 서비스이며, the Science Citation Index Expanded (SCIE) and the Social Science Citation Index (SSCI)로부터 편찬된 citations를 근거로 삼고 있다.

> Basic journal information

각 학술지에 대하여 다음과 같은 정보를 제공하고 있다:

1) publisher, title abbreviation, language, ISSN에 대한 기본적인 서지 정보.
2) 자연과학에 171개와 사회과학에 54개의 the subject categories.

> Citation information

기본적인 인용 데이터:

1) 해당연도에 출판된 기사의 수
2) 그 해 동안 후발 기사에 의해 자체적으로 또는 다른 학술지에 의해 그 학술지의 기사가 인용된 횟수

*** Coercive (강압적) citation**

이것은 학술지의 편집자가 학술지를 출판하기 전 기사에 대해 spurious(위조) citations을 추가하도록 강요하는 학술출판의 한 업무이다. 이렇게 함으로써 학술지의 IF를 인플레이시켜 인위적으로 그 학술지의 과학적 명성을 높이는 것이다.

Manipulation of impact factors and self-citation의 조작은 학술적 환경에서 오랫동안 눈살을 찌푸려 왔다: 그렇지만 2012년 survey 결과에서는 economics, sociology, psychology, and multiple business disciplines의 저작 중 약 20%에서 coercive citation을 경험한 것으로 나타났다.

*** SCImago Journal Rank**

SCImago Journal Rank (SJR indicator)는 학술지에 의해 인용된 인용문의 수와 그러한 인용문이 있던 학술지의 중요성 또는 prestige(명성) 둘 다를 계산하여 나타내는 학술지의 과학적 영향력에 대한 척도이다. The SJR indicator는 네트워크 이론에서 사용된 eigenvector (고유 벡터) centrality measure의 일종이다. 그 같은 척도는 높은 점수의 노드들로의 연결이 노드의 점수에 더 많이 공헌한다는 원칙을 근거로 네트워크에 있는 노드의 중요성을 수립한다.

또한 이 indicator는 the PageRank algorithm에 의해 영감을 받아서, 극단적으로 대규모이고 이질적인 학술지 인용 네트워크에서 사용되도록 개발되었다. 이것은 size-independent indicator이며, 그것의 값들은 학술지의 "average prestige per article"에 따라 학술지를 order 시킨다.

1) **PageRank**

이것은 Google Search에서 사용하는 algorithm이며, 탐색엔진의 결과를 가지고 웹사이트를 서열화 시킨다. PageRank는 was named after Larry Page(one of the founders of Google)에 의해 그 이름이 붙여졌으며, website pages의 중요성을 측정하는 한 방법이다.

Google에 따르면, "**PageRank는 웹 사이트의 중요도를 측정하기 위하여 링크의 수와 품질을 계산한다. 이것의 중요한 기본적 가정은 보다 중요한 웹사이트들은 다른 웹사이트로부터 더 많은 링크를 받는다는 것이다. – Facts about Google and Competition.**"

그렇지만 이것이 탐색엔진의 결과를 배열하기 위하여 사용하는 Google의 유일한 알고리즘이 아니라, 그 회사에서 사용한 첫 번째이면서 가장 잘 알려진 알고리즘이다. Google은 실재적으로 링크를 세고 웹 페이지에서 다른 정보를 수집하기 위하여 **Googlebot**라 부르는 an automated web spider를 사용하고 있다.

*** Acknowledgement index**

이 색인은 과학문헌의 acknowledgments(사사 , 승인)를 분석하고 색인하는 방법이다. 따라서 the impact of acknowledgments를 계량화 한다. 전형적으로 학술기사에는 저자가 자신들의 저작에 영향을 끼쳤거나 영감을 준 또는 자료나 지식에 도움을 준 funding, technical staff, colleagues, etc.과 같은 엔티티들에 사사(acknowledge)하는 섹션을 가지고 있다.

p. clxvii

*** 분류자질 선정의 대표적인 기법**

> 빈도기법(단어빈도, 문헌빈도, 역문헌빈도)
> 상호정보량(mutual information)
> 정보획득량(inf. gain)
> 카이제곱 통계량(X^2)

In probability theory and statistics, the chi-squared distribution (also

chi-square or χ^2-distribution) with k degrees of freedom(자유도) is the distribution of a sum of the squares of k independent standard normal random variables. It is one of the most widely used probability distributions in inferential statistics, e.g., in hypothesis testing or in construction of confidence intervals.

The chi-squared distribution is used in the common chi-squared tests for goodness of fit of an observed distribution to a theoretical one, the independence of two criteria of classification of qualitative data, and in confidence interval estimation for a population standard deviation of a normal distribution from a sample standard deviation.

p. clxviii

*** 문헌 범주화에 사용되는 분류기**

> 나이브 베이즈 분류기(naive Bayes classifier)

이 분류기는 단순한 확률적 분류기의 일종이며, "독립적인 특성의 모델"일 수 있는 강한(순수한) 모델에 베이즈 정리(theorem)를 기본적으로 적용하고 있다.

Naive Bayes는 1950년대 이후 활발하게 연구되어 왔다. 이것은 1960년대 초기에 텍스트 검색 집단에서 다른 이름으로 소개되었으며, the features로서 단어의 빈도을 가지고 spam or legitimate, sports or politics, etc.과 같은 하나의 카테고리나 다른 카테고리에 다큐멘트들이 소속되도록 하는 문제를 다루는 text categorization용으로 인기 있는 방법이 되었다. Naive Bayes classifiers는 learning problem에 있어서 변수(features/predictors)의 수에 선형적인 parameters의 수를 필요로 함으로써, 매우 scalable 하다.

1) Bayes' theorm
이것은 두 확률 변수의 사전 확률과 사후 확률 사이의 관계를 나타내는 정리다. 베이즈 확률론 해석에 따르면, 베이즈 정리는 새로운 근거가 제시될 때 사후 확률이 어떻게 갱신되는지를 구하는 것을 말한다.

> 의사결정나무 분류기 (Decision Tree classifier)

이것은 전산이나 통신의 모델이며, 그것의 알고리즘이나 통신절차가 기본적으로 의사결정나무 식으로 - 다시 말해서, 어떤 수치(quantities)의 비교(the unit computational cost을 근거로 이루어지는 비교)를 근거로 branching operations의 순서를 나타내는 방식으로 - 이루어진다.

> kNN(k-nearest neighbors) classifier

패턴 인식에서, the k-Nearest Neighbors algorithm (or k-NN for short)은 분류와 회귀분석(종속과 독립 변수들 간의 관련성을 평가하는 통계적 방법으로 prediction과 forecasting용으로 널리 사용되는 통계적 과정이다.)에서 사용되는 a non-parametric method 이다. 양쪽 경우 모두 그것의 입력은 feature space에 있는 the k closest training examples로 이루어지며, 출력은 k-NN가 분류를 이용하느냐 또는 회귀분석을 이용하느냐에 따라 결정된다.

1) Non-parametric regression

이것은 a predictor를 어떤 미리 결정된 형태로 취하는 것이 아니라 데이터에서 나온 정보에 따라서 만드는 일종의 회귀분석을 말한다. Nonparametric regression은 parametric models을 근거로한 회귀분석보다 더 커다란 샘플 사이즈를 필요로 하는데, 그 이유는 데이터가 그것의 모델 구조뿐만 아니라 모델 평가(the model structure as well as the model estimates)를 제공해야하기 때문이다.

2) Nearest neighbor search (NNS)

이것은 proximity search, similarity search 또는 closest point search로도 알려져 있으며, closest (or most similar) points를 찾는데 있어 최적화 문제를 다룬다. 그리고 closeness는 전형적으로 dissimilarity function으로 표현 된다: 유사성이 떨어질수록 그 사물은 더욱 커다란 function values를 갖는다.

> SVM(Support Vector Machine) classifier

machine learning 분야에서, **support vector machines (SVMs**, also support vector networks)는 데이터를 분석하고 패턴을 인식하는 학습 알고리즘과 결합된 학습모델을 관리감독(supervised) 하며, 분류와 회귀분석용으로 사용된다.

> 신경망(Neural Network) classifier

artificial neural networks는 기계 학습과 패턴 인식 능력을 갖고 있는 동물의 중추신경 시스템(특히 두뇌)에서 영감을 얻어서 만든 computational models 이다. 이것들은 대체로 네트워크를 사용하여 정보를 공급함으로써 이루어지는 입력으로부터 나온 값들을 계산할 수 있도록 상호 연결된 “neurons”의 시스템으로 표현되고 있다.

p. clxxii

*** 유사도 계수(유사계수)**

> 거리계수(distance coefficient)

>> Bhattacharyya distance

통계학에서 이것은 2 개의 이산적이거나 연속적인 확률분포의 유사성(similarity)을 측정하는 것이다. 이것은 두 개의 통계적 샘플이나 모집단 간의 중첩된 양의 척도인 the

Bhattacharyya coefficient와 밀접하게 관련되어 있다. 이 계수는 2개의 샘플에 대한 상대적 근접성을 결정하는데 사용될 수 있으며, 분류에 있어서 classes의 분리를 측정하는데도 사용될 수 있다.

>> 유클리드 거리(Euclidean distance),

수학에서, Euclidean distance or Euclidean metric는 누구나 자를 가지고 측정할 수 있는 두 포인트 간의 "ordinary" distance를 말하며, Pythagorean formula에 의해 얻을 수 있다. 거리와 관련해서 이 공식을 사용함으로써, Euclidean space (or even any inner product space)은 a metric space가 된다. 이것은 또한 Euclidean norm이라고도 부르며, 옛 문헌에서는 Pythagorean metric이라 부르고 있다.

>> 민코스키 매트릭스(Minincowski metrics)

이것은 the Euclidean distance and the Manhattan distance를 일반화한 것이라고 여겨질 수 있는 Euclidean space의 행렬(metric)이다.

>> 시티 블록 거리(city block distance)

19세기 독일에서 Hermann Mindowski가 생각한 Taxicab geometry는 기하의 일종이며, 일상적인 distance function or metric of Euclidean geometry를 두 개의 포인트 간의 거리가 그것들의 Cartesian coordinates의 절대적 차이의 합인 새 행렬로 대체시킨 것이다. The taxicab metric은 또한 rectilinear distance, L1 distance or norm (see Lp space), **city block distance**, Manhattan distance, or Manhattan length로 알려져 있다.

1) A Cartesian coordinate system

이것은 좌표 시스템으로, 한 쌍의 숫자 좌표로 평면 위에 각 포인트를 유일하게 나타내고 있으며, 특정 포인트로부터 동일한 길이의 단위로 측정된 두 개의 직각선까지를 표시한 거리이다. 각 reference line을 a coordinate axis or just axis of the system이라고 부르며, 이것들이 만나는 point가 그것의 origin이며, 대체로 이것은 ordered pair (0, 0)로 시작한다. 이 좌표들은 또한 두 축 상의 포인트에 대한 직각적(perpendicular projections) 위치로 정의될 수 있다.

> 연관계수(association coefficient)

>> 코사인 계수(Cosine coefficient):

문헌이나 용어 클러스터링에서 가장 많이 사용되는 유용한 척도이다.

Cosine similarity는 코사인 각도로 측정하는 inner product space의 두 벡터 간에 나타나는 유사도의 척도이다. The cosine of 0° 은 1 이며, 따라서 다른 각도는 1 보다 작다. 그러므로 이것은 orientation의 판단이지 magnitude는 아니다: 동일한 orientation을 갖고 있는 두 개의 vectors는 a Cosine similarity of 1를 가지며, 90°인 두 개의 벡터들은 a similarity of 0을 갖는다. 그리고 전혀 반대인 두 개의 벡터들은 a similarity of −1을 가지며, 자신들의 magnitude와는 독립적이다. 또한 Cosine similarity는 특히 [0,1] 영역(bound) 사이에서 산뜻하게 결과가 나타나는 positive space에서 사용되고 있다.

주목할 것은 이러한 영역들은 어떤 차원에 적용할 수 있으며, Cosine similarity가 high-dimensional positive spaces에서 가장 일반적으로 사용되고 있다는 것이다. 예를 들

어, Information Retrieval and text mining에서 각 용어는 개념적으로(notionally) 다른 차원에 배정되며 다큐멘트는 각 차원의 값이 그 다큐멘트에서 그 용어가 나타나는 횟수와 일치하는 벡터로 표시된다. 그런 다음에 Cosine similarity는 두 다큐멘트가 주제와 관련해서 얼마나 서로 유사한지를 나타내는 유익한 척도를 제공한다. 이 기법은 또한 data mining에서 클러스트의 응집력(cohesion)을 측정하는데도 사용된다.

또한 Cosine distance는 positive space, 즉: $D_C(A,B) = 1 - S_C(A,B)$에서 보완용으로 가끔 사용되는 용어이다. 그렇지만 이것을 주목해야 하는데, 이것은 triangle inequality property을 갖지 않음으로써 그리고 coincidence axiom을 위반함으로써 proper distance metric가 아니라는 것이다. coincidence axiom이란 to repair the triangle inequality property whilst maintaining 동일한 ordering을 유지하는 동안 triangle inequality property를 정정하는(repair)하는 것을 말한다.

Cosine similarity가 인기 있는 한 가지 이유는 단지 non-zero dimensions만을 고려하기 때문에, 특별하게 sparse vectors를 평가하는데 매우 효율적이기 때문이다.

>> 자카드 계수(Jaccard coefficient):
문헌이나 용어 클러스터링에서 가장 많이 사용되는 유용한 척도

The Jaccard index은 Jaccard similarity coefficient (originally coined coefficient de communauté by Paul Jaccard)로 알려져 있으며, 샘플 세트들의 유사성과 다양성을 비교하는데 사용되는 통계치이다. 또한 Jaccard coefficient는 한정된 sample sets 간에 유사성을 측정하며, 다음의 공식처럼 the size of the intersection을 the size of the union of the sample sets로 나눈 것이다:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}.$$

(만일 *A와 B가 둘 다 비어있다면*(empty), 우리는 *J(A,B)*=1로 정의한다.)
따라서 자카드 계수는 분명히 $0 \leq J(A,B) \leq 1$. 이다.

>> 다이스 계수(Dice coef.)

The Sørensen-Dice index라고도 하며, 두 개의 샘플 간에 유사도를 비교하는데 사용되는 통계치이다.

It was independently developed by the botanists Thorvald Sørensen and Lee Raymond Dice, who published in 1948 and 1945 respectively.

The index is known by several other names, usually Sørensen index or Dice's coefficient. Both names also see "similarity coefficient", "index", and other such variations. Common alternate spellings for Sørensen are Sorenson, Soerenson index and Sörenson index, and all three can also be seen with the -sen ending.

>> 해만 계수(Hamann coefficient)

유사도 측정을 위하여 주로 genetic research에서 사용되는 기법들 중의 하나이다.

>>Hamming distance

정보이론에서, 동일한 길이를 갖고 있는 두 개의 문자열(strings) 간의 Hamming distance는 상응하는 부호(corresponding symbols)가 다른 위치의 수이다. 이것은 하나의 문자열에서 다른 것으로 변경하려고 하는 대입(substitutions)의 최소의 수 또는 하나의 문자열에서 다른 문자열로 변형할 때 나타날 수 있는 최소한의 에러의 수를 측정한다.

<예>

The Hamming distance between:

"toned" and "roses" is 3.
1011101 and 1001001 is 2.
2173896 and 2233796 is 3.

*** 상관계수(correlation coefficient)**

> 피어슨 적률(Pearson product moment) 상관계수

통계학에서, Pearson product-moment correlation coefficient (/ˈpɪərsɨn/) (sometimes referred to as the PPMCC or PCC, or Pearson's r)는 두 개의 변수 X와 Y 간의 선형적 상관관계(dependence)에 대한 척도이다. 이것은 +1과 −1 사이의 값을 가지며, 1은 전체적으로 긍정적 상관관계를, 그리고 0은 어떠한 상관관계도 없음을, 그리고 -1은 전체적으로 부정적인 상관관계를 의미한다. 이것은 두 변수 간의 선형적 의존정도를 측정하는 도구로서 과학계에서 널리 사용되고 있으며 1880년대에 Franceis Galton에 의해 소개된 a related idea를 근거로 Karl Pearson에 의해 개발되었다.

*** 내적 계수(inner product coefficient)**

선형대수에서, an inner product space는 an inner product라고 부르는 추가적 구조를 갖고 있는 a vector space이다. 이 추가적 구조에서 각 쌍의 vectors와 the inner product of the vectors라고 알려진 a scalar quantity를 결합(associates)시킨다.

Inner products에서는 벡터의 길이 또는 두 벡터간의 각도와 같은 직관적인 기하학적 개념을 엄격하게 적용하고 있으며, 또한 벡터(zero inner product) 간의 직교(orthogonality)를 정의하는 수단을 제공하기도 하며, 어떤(possibly infinite) 차원의 벡터 스페이스를 다루는 Euclidean spaces (in which the inner product is the dot product, also known as the scalar product)을 일반화한 것으로, functional analysis에서 연구되고 있다.

또한 An inner product는 자연스럽게 결합된 개념(norm)을 유도하기 때문에 a normed vector space 이라고도 한다. 그리고 an inner product가 있는 완전한 스페이스를 Hilbert space라고 부르며, an inner product가 없는 불완전한 스페이스를 pre-Hilbert space라고 부르는데, 그 이유는 inner product에 의해 유도된 개념과 관련돼서 그것을 완성시킨 다음에야 Hilbert space가 되기 때문이다. 이것은 또 다른 분야에서는 때때로 unitary spaces라고 부른다.

p. clxxiv

*** Cluster analysis or clustering**

이것은 동일한 그룹(a cluster)에 있는 사물들이 다른 그룹(clusters)에 있는 사물들보다 보다 유사하게 사물들을 집단화하는 task이다. 이것의 주 업무는 machine learning, pattern recognition, image analysis, information retrieval, and bioinformatics 등에서 사용되는 통계적 데이터 분석의 일반적 기법들과 exploratory data mining 이다.

1) Cluster analysis

이것은 그 자체가 하나의 특별한 알고리즘이 아니라, 해결하기 위한 일반적인 업무이다. 이것은 무엇이 클러스트를 구성하는지 그리고 얼마나 효율적으로 그것들을 찾을 수 있는지에 대한 개념(notion)이 분명히 차이가 나는 다양한 algorithms에 의해 완성될 수 있다. 클러스터에서 많이 사용하는 개념은 클러스터 멤버들 간에 거리가 작은 그룹들 즉, data space, intervals or 특별한 통계적 분포가 밀집되어 있는 그룹들이다. 그러므로 Clustering은 a multi-objective optimization problem로 공식화될 수 있다. 적합한 clustering algorithm 과 parameter settings (사용할 거리 함수, density threshold 또는 예상된 클러스터의 수와 같은 값을 포함하여)은 각각의 data set 그리고 그 결과에 대한 의도적 용도에 따라 다를 수 있다. 그리고 이것은 자동적 업무가 아니며, 지식발견에서 이루어지는 시도와 실패를 포함하여 interactive multi-objective optimization를 반복하는 과정이며, 원하는 성질의 결과를 얻을 때까지 data preprocessing 그리고 model parameters를 종종 변경하기도 한다.

clustering이란 용어 이외에도 많은 비슷한 의미의 용어가 있다: automatic classification, numerical taxonomy, botryology (from Greek βότρυς "grape") and typological analysis. 미묘한 차이가 종종 결과의 용도에서 발생한다: data mining에서 최종결과 그룹은 관심의 대상(the matter of interest)인 반면에, automatic classification에서 최종결과를 차별하는 힘은 of interest 이다. 따라서 종종 data mining 과 machine learning의 연구자들 사이에 오해가 발생하는데, 그 이유는 그들이 동일한 용어와 동일한 알고리즘을 사용하지만 서로 다른 목표를 갖고 있기 때문이다.

그리고 Cluster analysis was originated in anthropology by Driver and Kroeber in 1932 and introduced to psychology by Zubin in 1938 and Robert Tryon in 1939 and famously used by Cattell beginning in 1943 for trait theory classification in personality psychology.

2) Clustering algorithms

이것은 클러스터 모델을 근거로 범주화될 수 있다. 다음은 클러스터 알고리즘의 가장 뛰어난 예들만을 열거한 것이다. 객관적으로 “correct”한 클러스터링 알고리즘이 없는 것은 아니지만 주목해야 하는 것은 "clustering is in the eye of the beholder"라는 것이다. 특별한 문제에 대한 가장 적합한 클러스터링 알고리즘은 만일 다른 것에 비하여 한 가지 클러스터 모델을 선호하는 수학적 이유가 존재하지 않는다면, 종종 경험적으로(experimentally)으로 선택되어야 한다. 주목해야 하는 것은 한 종류의 모델용으로 설계된 알고리즘은 급격하게 변하는

다양한 모델을 갖고 있는 데이터 세트에 어떠한 변화도 주지 못한다. 예를 들어, k-means(아래의 비계층적 기법 참조)에서는 non-convex(볼록한) clusters를 찾을 수 없다.

> 클러스트링 알고리즘의 계층적 기법: Connectivity based clustering (hierarchical clustering)

계층적 클러스터링으로 알려진 Connectivity based clustering은 멀리 떨어져 있는 사물(objects)보다는 근처에 있는 사물들이 더욱 관련이 있다는 사물에 대한 핵심 아이디어에 근거 한다. 이런 알고리즘은 사물의 거리를 근거로 연결하여 “클러스터즈”를 형성한다. 클러스터는 대부분이 그 클러스터의 부분들을 연결하기 위해 필요한 최대의 거리로 표현될 수 있다. 그리고 거리가 서로 다른 클러스트들은 어디서부터 공동의 이름인 "hierarchical clustering"이 시작되었는지를 묘사하는 dendrogram으로 표현될 수 있다: 이들 알고리즘들은 데이터 세트의 단독 분할(partitioning)을 제공하지 않지만, 그 대신에 특정 거리에서 서로가 통합되는 클러스터들의 광대한 계층을 보여주고 있다. 그리고 dendrogram에서, y 축은 클러스터들이 통합할 수 있는 거리를 표시하는 반면에, 사물들은 클러스터들이 섞이지 않도록 x 축을 따라 배치된다.

Connectivity based clustering은 거리를 계산하는 방식에서 차이가 나타나는 a whole family of methods 이다. 일반적인 distance functions의 선택과는 달리, 이용자는 또한 사용할 linkage criterion(하나의 클러스터가 다수의 사물로 이루어져 있으므로, 그것의 거리를 계산하기 위한 다수의 candidates가 존재한다.)을 결정하여야 한다. 인기 있는 선택으로는 single-linkage clustering (the minimum of object distances), complete linkage clustering (the maximum of object distances) or UPGMA ("Unweighted Pair Group Method with Arithmetic Mean", also known as average linkage clustering)가 있다. 또한 hierarchical clustering은 응집되거나(agglomerative - starting with single elements and aggregating them into clusters) 분산(divisive - starting with the complete data set and dividing it into partitions)될 수도 있다.

>> 단일 연결(single linkage)

이것은 agglomerative hierarchical clustering의 여러 가지 방법들 중의 하나이다. 이것을 시작하는 과정에서, 각 요소는 그 자신의 클러스터에 들어 있다. 그 클러스터들은 그런 다음에 순차적으로 보다 커다란 클러스터에 결합되며, 모든 요소가 동일한 클러스터에 포함될 때까지 계속된다. 각 단계에서, 가장 짧은 거리에 의해 분리된 두 개의 클러스터들이 결합된다. ‘shortest distance'에 대한 정의는 여러 agglomerative clustering methods 사이에 차이가 있다. single-linkage clustering에서, 두 클러스터 간의 link는 하나의 단일 요소 pair 즉, 서로에게 가장 가까이 있는 두 개의 요소들에 의해 만들어진다. 어떤 단계에서 이러한 가장 짧은 링크들은 그 같은 요소를 포함하고 있는 두 개의 클러스터들을 융합시키는 원인을 제공하며, 이러한 방법은 nearest neighbour clustering이라고도 한다. 이러한 클러스터링의 결과는 각각의 융합이 발생하는 거리와 클러스터의 융합 순서를 보여주는 dendrogram으로 시각화할 수 있다.

>> 완전 연결(complete linkage)

이것도 agglomerative hierarchical clustering의 여러 가지 방법들 중의 하나이다. 이

것도 단일연결과 마찬가지로 그 과정이 이루어지지만, complete-linkage clustering에서, 두 클러스터간의 링크는 모든 요소 pair를 포함하며, 클러스터간의 거리는 서로 아무리 멀리 떨어져 있다하더라도 이들 두 요소간의 거리는 똑 같다. 어떤 단계에서 이러한 가장 짧은 링크들은 그 같은 요소를 포함하고 있는 두 개의 클러스터들을 융합시키는 원인을 제공하며, 이러한 방법은 farthest neighbour clustering이라고도 한다. 이러한 클러스터링의 결과 역시 각각의 융합이 발생하는 거리와 클러스터의 융합 순서를 보여주는 dendrogram으로 시각화할 수 있다.

>> 그룹 평균 연결(group average agglomerative linkage)

Group-average agglomerative clustering or GAAC는 다큐멘트 간의 모든 유사성을 근거로 클러스터의 품질을 평가한다. 따라서 single-link and complete-link criteria의 단점을 피할 수 있다. GAAC는 group-average clustering 그리고 average-link clustering라고도 부르며, 동일한 클러스터의 pairs를 포함하여 다큐멘트의 모든 pairs에 대한 average similarity를 계산한다.

>> 와드 기법 연결(Ward's method linkage)

통계학에서, Ward's method은 계층적 클러스터 분석에 적용된 criterion 이다. Ward's minimum variance method은 최초로 Joe H. Ward, Jr.가 제시한 the objective function approach의 특별한 케이스 이다. Ward는 일반적인 agglomerative hierarchical clustering procedure를 제안했는데, 이것은 각 단계에서 통합되는 클러스터의 pair를 선택하기 위한 기준이 되었으며, objective function의 최적 값을 근거로 하고 있다. 이 objective function는 조사자의 목적을 반영하는 어떤 함수("any function that reflects the investigator's purpose.")일 수 있다. 많은 표준적 clustering procedures에는 이러한 매우 일반적인 유형을 포함하고 있으며, 이 절차를 설명하기 위하여, Ward는 “objective function이 error sum of squares”라는 예를 제시하였으며, 이 예는 Ward's method 또는 보다 정확하게 Ward's minimum variance method으로 알려졌다.

> 클러스트링 알고리즘의 비계층적 기법

>> single pass algorithm

싱글 패스 클러스터링은 특정 임계치를 설정하여, 시간 윈도우(Time window) 내에 발생한 과거 사건들과 새로 발생한 사건 사이의 유사성 정도를 계산한다. 만약 설정한 임계치 보다 낮은 유사도를 가지게 되면, 해당 사건은 새로운 이벤트로 판단한다. 단, 일반적으로 기존의 대부분의 경우 이러한 유사성 계산은 전통적 벡터 공간 모델에 기반한 Cosine 유사도 지표를 사용한다.

>> K-means algorithm

이것은 centroid-based clustering 방법이며, 클러스터들은 많은 데이터 세트에서 필요하지 않을 수도 있는 central vector에 의해 표현된다. 이것의 목적은 클러스터의 원형(prototype)으로 도움을 줄 수 있도록, 각각의 관찰을 가장 가까운 평균을 가지고 있는 클러스터에 포함시킴으로써 관찰들(observations)을 클러스터들로 쪼개는(partition) 것이다.

클러스터의 수가 k로 고정되어 있을 때, k-means clustering은 최적화 문제에 대한 공식적 정의를 제공 한다: 클러스터 센터를 찾은 다음에, 클러스터의 squared distances를 최소화하여 가장 가까운 클러스터 센터에 그 사물들을 배정한다.

optimization problem는 NP-hard (Non-deterministic Polynomial(다항식)-time hard: 라고 하며, 이것을 다루는 일반적인 방법은 단지 approximate solutions만을 탐색하는 것이다. 특히 잘 알려진 approximative method은 Lloyd's algorithm이며, 실재로는 "k-means

K-means는 재미있는 이론적 성질을 많이 가지고 있다. 하나는 cal properties. On the one hand, it partitions the data space into a structure known as a Voronoi diagram으로 알려진 구조로 데이터 스페이스를 분할하는 것이고, 또 하나는 개념적으로 . On the other hand, it is conceptually close to nearest neighbor classification과 밀접하다는 것이다. 또 하나 더 이것은 classification을 근거로 하는 모델의 변종일 수 있으며, as a variation of the Expectation-maximization algorithm의 변종인 Lloyd's algorithm으로 여겨질 수 있다.

1) Voronoi diagram

수학에서, 이것은 평면의 특별한 부분집합에 있는 포인트들의 거리를 근거로 평면을 분할한 것이다. That set of points (called seeds, sites, or generators)는 미리 지정되며, 각 seed용으로 다른 것보다 그 seed에 보다 가까운 모든 포인트들로 구성된 corresponding region가 존재한다.

>> EM(Expectation Maximization) 알고리즘

통계학에서, expectation-maximization (EM) algorithm은 관찰되지 않고 숨어 있는 변수(unobserved latent variables)에 의존하는 통계모델에서 파라미터의 maximum likelihood estimation(MLE: mean(평균)과 variance(분산)을 사용하여 알려지지 않은 모델 변수의 값을 가정한다.) 또는 maximum a posteriori (MAP) estimates(경험적 데이터를 근거로 관찰되지 않는 quantity의 point estimate를 얻는데 사용될 수 있다)를 찾기 위하여 시행되는 반복적 방법(iterative method) 이다. EM iteration은 파라미터의 최신 평가치를 사용하여 평가된 log-likelihood의 기대에 관한 함수를 생산하는 expectation (E) step, 그리고 E step에서 발견된 예상된 log-likelihood를 최대화하기 위하여 파라미터를 계산하는 a maximization (M) step을 수행하는 사이에 교대로 발생한다. 그런 다음에 이 파라미터 평가치들은 다음 번의 E step에 숨어있는 변수들의 분포를 결정하기 위하여 사용된다.

*** 웹 문헌 클러스터링**

> 단어기반(term-based) 클러스터링: 단어의 유사도에 기반, 텍스트가 많은 웹 문헌.
> 링크기반(link-based) 클러스터링: 이미지가 많은 웹 문헌.

>> intra-document link
>> inter-document link
>> out-link
>> in-link

> 혼합형(hybrid) 클러스터링

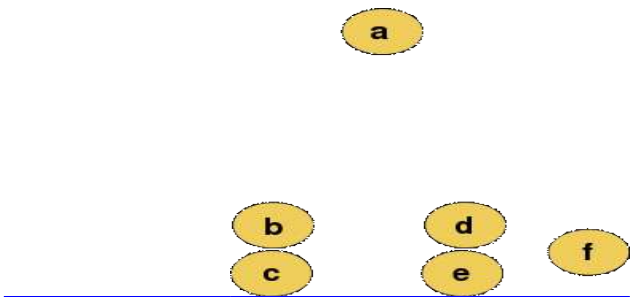
p. clxxx

*** 덴드로그램(dendrogram**: from Greek*dendron* "tree" and *gramma* "drawing"**):**

클러스터 생성 과정을 표현하는 한 방법이며, 계층적 클러스터링에 의해 생산된 클러스터들의 배열을 보여주기 위하여 종종 사용되는 나무형태의 도형이다.

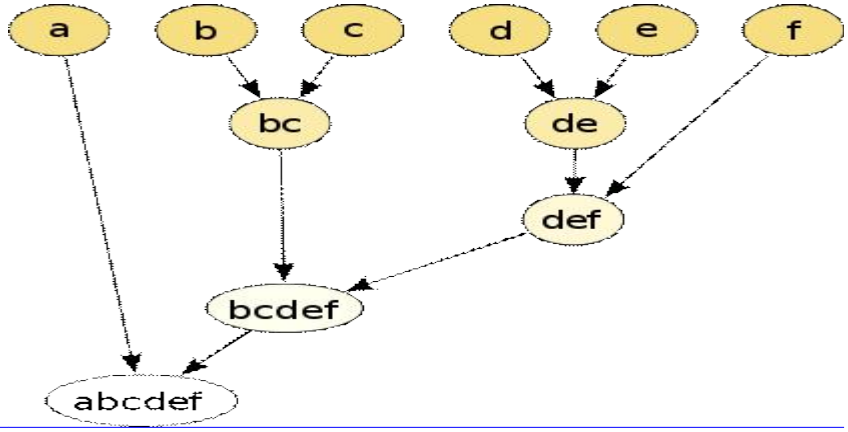
> Clustering Example

클러스터링의 예로서, 이 데이터들이 distance metric으로 Euclidean distance를 사용하여 클러스터 된다고 가정해 보자:



Raw data

이것의 계층적 클러스터링 덴더그램은 다음과 같을 것이다:



Traditional representation

The top row of nodes represent data (individual observations), and the remaining nodes represent the clusters to which the data belong, with the arrows representing the distance (dissimilarity).

맨 위에 있는 노드들을 row는 개별적으로 관찰된 데이터를 나타내며, 나머지 노드들은 그 데이터가 속해 있는 클러스터들을 나타내며, 이때에 화살표들은 노드들 간의 거리(dissimilarity)를 나타낸다.

<제 7장 정보검색 언어>

*** Web 1.0, 2.0, 3.0**

> Web 1.0

Geocities & Hotmail 시대에는 모두가 read-only content and static HTML websites 이었으며, 사람들은 Yahoo!의 link directories를 사용하여 인터넷을 항해하는 것을 선호하였다.

-the mostly read only web
-45million global users(1996)
-focused on companies
-home pages
-owning content
-Britannica Online
-HTML. ports
-web forms
-directories(taxonomy)
-Netscape
-page views

-advertising

> Web 2.0

이것은 user-generated content and the read-write web에 관한 것이다. 사람들은 Flickr, YouTube, Digg, etc.과 같은 블로그나 사이트를 통해 정보를 제공할 뿐만 아니라 소비하고 있다. The line dividing a consumer and content publisher를 구분하는 경계가 Web 2.0 시대에서는 점점 더 모호해 지고 있다.

-the wildly read-write web
-1 billion + golobal users(2006)
-focused on communities
-blogs
-sharing content
-Wikipedia
-XML, RSS
-web applications
-tagging("folksonomy")
-Google
-cost per click
-word of mouth

> Web 3.0

이것은 semantic web (or the meaning of data), personalization (e.g. iGoogle), 사물간의 intelligent search and behavioral advertising에 관한 것이다.

-the portable personal web
-focused on the individual
-lifestream
-consolidating dynamic content
-the semantic web
-widgets, drag & drop mashups

1) mashup

이것은 새로운 서비스를 창조하기 위하여 두 개 이상의 소스 들에서 나온 data, presentation or functionality을 결합하여 사용하는 Web page or application을 말한다. 이 용어는 raw source data를 생산하기 위하여 반드시 본래의 이유가 아니더라도 풍부한 결과를 얻기 위하여 open APIs and data sources를 사용하여 편하고 신속하게 통합시키는 것을 의미한다.

이것의 주요한 특징들은 combination, visualization, and aggregation 이며, 기존의 데이터를 개인적 또는 직업적 목적으로 보다 더 유용하게 만드는데 중요하다. 또 다른 서비스들의 데이터에 항구적으로 접근하기 위하여, 이것들은 일반적으로 client applications or hosted online들이다.

-user behavior("me-onomy")
-iGoogle, NetVibes
-user engagement
-advertainment

p. clxxxvi

*** 토픽맵(Topic Maps) : See Chptr 11.**

*** 시맨틱 웹(Semantic Web) : See Chptr 11.**

*** 온톨로지(Ontology): See Chptr 11**

*** RDF(Resource Description Framework)**

RDF는 W3C 스펙의 한 종류이며, 원래는 메타데이터 데이터 모델로 디자인되었다. 이것은 다양한 syntax notations과 data serialization formats을 사용하여 웹 자원에서 사용되고 있는 정보의 개념적 기술이나 모델에 관한 일반적 방법이다.

1) serialization

데이터 저장분야에서, 이것은 예를 들어, a file or memory buffer에 있거나, network connection link를 통해 전송되어온 data structures or object state를 저장하여 나중에 동일한 환경에서 재조직할 수 있는 포맷으로 번역해 주는 과정을 말한다. serialization format에 따라 최종적인 일련의 비트들이 다시 읽혀질 때, 이것은 원래의 사물과 어의적으로 동일한 clone을 만드는데 사용된다. 많은 references를 사용하는 복잡한 사물에서, 이런 과정은 직선적으로 이루어지지 않는다. object-oriented objects의 serialization은 전에 서로 복잡하게 연관되어 링크되어 있던 방법은 어떠한 것도 여기에 포함되지 않는다.

사물을 연속화시키는 이러한 절차를 또한 marshalling an object라고도 부르며, 일련의 바이트들로부터 데이터 구조를 발췌하는 반대 기능을 deserialization (which is also called unmarshalling)라고 부른다.

> Overview

RDF 데이터 모델은 객체-관계 또는 클래스 다이어그램과 같은 고전적 개념적 모델링 기법과 비슷하다. 그 이유는 주체-술어-객체(subject-predicate-object) 표현의 형태로 자원(특히 웹자원에서)을 표현한다는 아이디어에서 비롯되었기 때문이다. 이러한 표현식을 triples이라 하며, 주체는 자원을 의미하며, 술어는 자원의 속성이나 모습, 그리고 주체와 객체 사이의 관계를 나타낸다.

예를 들어, RDF로 “The sky has the color blue”라는 개념(notion)을 표현하는 한 가지 방법은 다음과 같은 트리플(주체: “the sky”, 술어: “has”, 그리고 객체: “the color blue”)로 나타내는 것이다. 그러므로 RDF는 object를 객체-지향형 디자인에 있는 엔티티-속성-값(entity-attribute-value)의 고전적 개념에 사용될 수 있는 subject로 바꿀 수 있다; 객체(sky), 속성(color), 값(blue). RDF는 여러 가지 연속 포맷(다시 말해서 파일 포맷)으로 된 하나의 추상적 모델이므로, 특정한 하나의 자원이나 트리플을 코드화하는 방법은 포맷마다 차이가 난다. 이 같은 메카니즘으로 자원을 기술하는 것은 웹을 통해 보급되는 소프트웨어를 통하여 기계가독형 정보를 저장, 교환, 이용할 수 있도록 함으로써, 이용자는 보다 높은 효율성과 확실성을 가지고 정보를 다룰 수 있으며, 이것은 시멘틱 웹의 활동에서 중요한 요소가 되었다.

RDF 서술문(statements)에는 본질적으로 표식과 통제된 멀티 그래프를 포함하고 있다. 마찬가지로, RDF-의존형 데이터 모델은 관계형 모델이나 기타 온톨로지 모델보다도 특정한 종류의 지식을 표현하는데 있어서 보다 자연스럽다. 그렇지만 실재로 RDF 데이터가 만일 context(다시 말해서 the named graph) 역시 각 RDF 트리플용으로 존속(persist)한다면, 종종 관계형 데이터베이스나 Triplestore라고 부르는 native representation, 또는 Quad stores 에도 존속하고 있다.

1) A **triplestore**
이것은 "Bob is 35" or "Bob knows Fred"처럼, subject-predicate-object로 구성된 데이터의 엔티티인 트리플들의 저장과 검색을 위한 a purpose-built database 이다. 이것의 관계형 데이터베이스와 매우 유사한 점은 누구나 트리플스토어에 정보를 저장한 다음에 쿼리 언어를 사용하여 그것을 검색할 수 있다. 관계형 데이터베이스와의 차이로 이것은 트리플들을 저장하고 검색하는데 최적화되어 있다는 것이다. 쿼리에 따라서, 트리플들은 일반적으로 RDF나 기타 포맷을 사용하여 수출입될 수 있다.

2) **Named graphs**
이것은 한 세트의 RDF statements(a graph)를 context, provenance information 또는 기타 metadata와 같은 statements를 descriptions하는 URI를 사용하여 식별하도록 하는 Semantic Web architecture의 핵심 개념이다.

Named graphs는 그래프를 만들 수 있는 RDF data model의 간단한 확장형이지만, 일단 웹에 출판된 그래프들을 구별하는 효과적 수단은 되지 못하고 있다.

또한 ShEX(Shape Expression)은 RDF 그래프의 constraints를 표현하기 위한 언어이며, OSLC Resource Shapes와 Dublin Core Description Set Profiles 뿐만 아니라 분리와 다형성을 위한 논리적 관계에 발생하는 cardinality constraints도 다루고 있다. RDFS와 OWL에서 보여주듯이, 누구나 RDF를 근거로 추가적인 ontology language를 작성할 수 있다.

1) RDF Schema (Resource Description Framework Schema, variously abbreviated as RDFS, RDF(S), RDF-S, or RDF/S)
이것은 RDF 자원을 구조화하기 위하여 온톨로지의 기술용인 기본요소(다른 말로해서, RDF vocabularies라고도 부르는)를 제공하는 RDF extensible knowledge representation language를 사용하여 특정 성질을 갖고 있는 a set of classes이다. 이러한 자원들은 쿼리언어 SPARQL을 사용하여 접근할 수 있는 a triplestore에 저장될 수 있다.

2) Web Ontology Language (OWL)
이것은 ontologies or knowledge bases를 authoring하기 위한 일종의 knowledge representation languages or ontology languages 이며, Semantic Web을 위한 공식적 semantics and RDF/XML-based serializations라는 특

성을 가지고 있다. OWL는 World Wide Web Consortium (W3C)에 의해 공인되었으며, academic, medical and commercial interest 분야에서 관심을 끌고 있다..

OWL family에는 비슷한 이름의 많은 species, serializations, syntaxes and specifications가 있으며, OWL and OWL2가 각각 2004 and 2009 specifications으로 사용되었다.

> RDF topics

>> RDF vocabulary: RDF specification에 의해 정의된 어휘는 다음과 같다:

1) Classes

<rdf>

◉ rdf:XMLLiteral - the class of XML literal values
◉ rdf:Property - the class of properties
◉ rdf:Statement - the class of RDF statements
◉ rdf:Alt, rdf:Bag, rdf:Seq - containers of alternatives, unordered containers, and ordered containers (rdfs:Container is a super-class of the three)
◉ rdf:List - the class of RDF Lists
◉ rdf:nil - an instance of rdf:List representing the empty list

<rdfs>

◉ rdfs:Resource - the class resource, everything
◉ rdfs:Literal - the class of literal values, e.g. strings and integers
◉ rdfs:Class - the class of classes
◉ rdfs:Datatype - the class of RDF datatypes
◉ rdfs:Container - the class of RDF containers
◉ rdfs:ContainerMembershipProperty - the class of container membership properties, rdf:_1, rdf:_2, ..., all of which are sub-properties of rdfs:member

2) Properties

<rdf>

◉ rdf:type - an instance of rdf:Property used to state that a resource is an instance of a class
◉ rdf:first - the first item in the subject RDF list
◉ rdf:rest - the rest of the subject RDF list after rdf:first
◉ rdf:value - idiomatic property used for structured values
◉ rdf:subject - the subject of the subject RDF statement
◉ rdf:predicate - the predicate of the subject RDF statement
◉ rdf:object - the object of the subject RDF statement

rdf:Statement, rdf:subject, rdf:predicate, rdf:object는 reification(관념의 구체화)용으로

사용된다. (see below).

<rdfs>
◉ rdfs:subClassOf - the subject is a subclass of a class
◉ rdfs:subPropertyOf - the subject is a subproperty of a property
◉ rdfs:domain - a domain of the subject property
◉ rdfs:range - a range of the subject property
◉ rdfs:label - a human-readable name for the subject
◉ rdfs:comment - a description of the subject resource
◉ rdfs:member - a member of the subject resource
◉ rdfs:seeAlso - further information about the subject resource
◉ rdfs:isDefinedBy - the definition of the subject resource

이 어휘는 확장용 RDF Schema의 기초로 사용될 수 있다.

3) Serialization formats
다음과 같은 여러 가지 공동의 serialization formats이 사용되고 있다:

◉ Turtle; a compact, human-friendly format.
◉ N-Triples; a very simple, easy-to-parse, line-based format that is not as compact as Turtle.
◉ N-Quads; a superset of N-Triples, for serializing multiple RDF graphs.
◉ JSON-LD; a JSON-based serialization.
◉ N3 or Notation 3; a non-standard serialization that is very similar to Turtle, but has some additional features, such as the ability to define inference rules.
◉ RDF/XML; an XML-based syntax that was the first standard format for serializing RDF.

RDF/XML을 때때로 간단하게 RDF라 부르는데 이것은 잘못된 것이다. 왜냐하면 이것은 RDF를 정의하고 있는 다른 W3C 스펙들 간에 소개되었고, 역사적으로도 첫 번째 W3C 표준 RDF serializtion format이기 때문이다. 그렇지만, RDF/XML format과 추상적인 RDF 모델 그 자체를 구분하는 것이 중요한데, 왜냐하면 비록 RDF/XML 포맷이 아직까지 사용 중이라 하더라도, 다른 RDF serializations를 이제 많은 RDF 사용자에 의해 선호되고 있기 때문이며, 또한 이것들은 인간 친화적이고, XML QNames의 구분법에 있는 restriction으로 인하여 어떤 RDF graph는 RDF/XML에서는 표현될 수 없기 때문이다.

1) QNames
이것은 URI references처럼 사용하기 위하여 XML Namespaces에 의해 소개되었으며, QName이란 "qualified name"을 말한다. 그리고 이것은 elements and attributes용으로 타당한 식별자를 정의하며, 일반적으로는 XML documents에 있는 특별한 elements or attribute를 참조하는데 사용된다.

4) Resource identification

RDF statements(문장)의 subject는 URI 또는 blank node이며, 둘 다 자원을 나타낸다. blank node가 가르키는(indicate) 자원은 anonymous resources 라 부르며, 이것들은 RDF statments에서 직접적으로 식별할 수는 없다. 또한 그것의 술어는 관계를 나타내는 자원을 의미하는 URI이고, object는 URI, blank node 또는 Unicode string literal 이다.

1) A **string literal(문자상수)**
이것은 컴퓨터 프로그램의 소스 코드에 있는 string value를 대표한다. 현대 언어에서 대부분 이것은 “foo”가 foo라는 값을 가진 문자 상수에서 x = "foo"처럼 a quoted sequence of characters (formally "bracketed delimiters")를 말한다. 여기서 인용부호들은 값의 일부분이 아니며 누구나 delimiters를 사용하여 escape characters를 이용할 수 있다.

그렇지만 문자열 상수를 특정화하는 수많은 대안적 표기법들이 있으며, 특히 보다 복잡한 경우에 정확한 표기가 개개의 프로그램 언어별로 존재해야 하는지에 대해서는 아직 의문스럽다. 그럼에도 불구하고 이것에는 대부분의 현대 프로그래밍 언어가 따라야 하는 몇 가지 일반적인 가이드라인이 존재한다.

시멘틱 웹 어플에서, 그리고 RSS와 FOAF 같이 RDF의 비교적 인기있는 어플들에서, 자원들은 웹의 실재적인 데이터에 접근하는데 사용하기 위하여 고의적으로 URIs로 표현되는 경향이 있다. 그러나 RDF에서는 일반적으로 인터넷-의존형 자원에 대한 표기용으로만 제한하고 있진 않다. 사실상, 자원의 이름인 URI는 결코 탈참고용(derefernceable)이 되어져서는 안된다. 예를 들어, “http:”로 시작하여 RDF 문장의 주제로 사용되는 URI는 반드시 http를 통해 접근할 수 있는 자원만을 표현할 필요는 없으며, 또한 실체적이고 네트워크로 접근 가능한 자원임을 표현할 필요도 없다 - URI는 무조건 어떤 것이든 표현할 수 있어야 한다. 그렇지만, 널리 인정받고 있는 것은 HTTP GET request에서 사용될 때 300가지의 암호화된 응답을 리턴하는 a bare URI(# symbol이 없는)는 접근에 성공한 인터넷 자원을 의미하는 것으로 처리되어야 한다는 것이다.

1) **RSS (Rich Site Summary)**; originally RDF Site Summary; often dubbed Really Simple Syndication
이것은 blog entries, news headlines, audio, video 처럼 자주 갱신되는 정보를 출판하기 위한 standard web feed formats의 일종이다. RSS document (called "feed", "web feed", or "channel")에는 출판날짜와 저자명과 같은 메타데이터와 full or summarized text가 포함되어 있다.

RSS feeds란 자동으로 출판사로 하여금 데이터를 배급(syndicate)하는 것을 말한다. A standard XML file format에서는 서로 다른 machines/programs 간의 호환성을 보장하며, 또한 이것은 좋아하는 웹사이트로부터 적시에 갱신된 정보를 받기 원하거나 많은 사이트로부터 데이터를 수집하고자 하는 이용자에게 도움을 준다.

Subscribing to a website RSS는 이용자가 새 내용을 위하여 웹사이트를 수작업으로 체크하여야 하는 필요성을 제거시키며, 이 브라우저는 지속적으로 웹 사이트를 모니터하여 어떤 갱신된 정보를 알려주기도 하고 자동으로 다운로드해 주기고 한다. web-based, desktop-based, or mobile-device-based한 "RSS reader", "aggregator", or "feed reader"는 이용자에게 RSS feed data를 제공하며, 이용자는 그 feeds에 가입하여야 한다. RSS reader는 새 정보와 관련해서 정규적으로 이용자의 feeds를 체크하여 자동적으로 그것을 다운로드 시켜준다. 이 리더는 또한 이용자 인터페이스를 제공하기도 한다.

2) **FOAF** (an acronym of Friend of a friend)
이것은 사람, 그들의 행동, 그리고 다른 사람과 사물과의 관계를 기술하는 a machine-readable ontology 이다. 누구나 FOAF를 이용하여 남녀를 기술할 수 있다. FOAF는 사람의 그룹으로 하여금 중앙식 데이터베이스에 대한 필요없이 사회적 네트워크를 묘사할 수 있도록 한다. FOAF는 Resource Description Framework (RDF) and Web Ontology Language (OWL)를 사용하여 표현된 descriptive vocabulary이다. 컴퓨터들은 이러한 FOAF profiles을 사용하여 예를 들어 모든 유럽 사람을 찾을 수 있도록 한다. 이것은 사람들 간의 관계를 정의함으로써 완성되며, 각

프로파일은 유일한 식별자(person's e-mail addresses, a Jabber ID, or a URI of the homepage or weblog of the person와 같은)를 가지며, 이것은 사람들간의 관계를 정의할 때 사용된다.

The FOAF project는 2000년에 Libby Miller and Dan Brickley에 의해 시작되었으며, RDF technology을 'Social Web'의 관심사에 연결시킨 최초의 Social Semantic Web application으로 여겨지고 있다.

또한 2007년에 Tim Berners-Lee는 Semantic web concept을 relationships가 networks and documents를 능가하는 Giant Global Graph로 재정의하였다. 그는 "I express my network in a FOAF file, and that is a start of the revolution."이라고 말하면서, GGG를 Internet and World Wide Web과 동일 토대 위에 있다고 생각하였다.

따라서 RDF 문장의 생산자와 소비자들은 자원 식별자의 어의에 일치하여야 한다. 그러한 일치는 비록 RDF에서 사용하기 위하여 URI space에 부분적으로 포함되는 Dublin Core Metadat처럼 일반적 용도의 몇 가지 통제어휘가 있다하더라도, RDF 그 자체에 필수적인 것은 아니다. 웹에서 RDF-의존형 온톨로지를 출판하는 목적은 RDF에서 데이터를 표현하기 위하여 사용된 자원 식별자에 대한 계획된 의미를 종종 제한하거나 확립함이다.

예를 들어, the URI:

http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#Merlot

위의 URI는 vintner에 의해 생산된 모든 Merlot 붉은 포도주의 등급을 언급하기 위하여 이것의 소유자가 의도적으로 만든 것이다. 다시 말해서, 위의 예에서는 양조인 한명이 생산한 모든 포도주의 등급을 표현하고 있으며, 이 정의는 스스로가 RDF 다큐멘트인 OWL 온톨로지에 의해 표현되고 있다. 따라서 이러한 정의에 대한 주의 깊은 분석이 없다면, 누구나 위의 URI의 예는 포도주의 종류 대신에 다른 물리적 사물이라는 잘못된 결론을 내릴 수도 있다.

이것은 'bare' resource identifier가 아니며, 그것보다는 '#' 문자를 포함하면서 fragment identifier로 마감하는 URI reference 라는 것에 주목하여야 한다.

1) A URI reference

이것은 a full URI의 형태이거나 빈문자열의 하나 또는 복수의 후속 구성요소로 된 scheme-specific portion일 수 있다. #로 시작되는 optional fragment identifier는 URI reference의 끝에 나타날 수 있으며 # 에 있는 references의 일부는 간접적으로 자원을 식별할 수 있고 그 fragment identifier는 그 자원의 특정 부분을 식별할 수 있다. 예를 들어, HTML에서, <img> 요소의 <src> 속성의 값은 <a> or <link> 요소의 <href> 속성의 값과 마찬가지로, URI reference를 제공하고 있다.

URI reference로부터 URI를 추출하기 위하여, 소프트웨어는 그 URI reference를 고정된 알고리즘에 따라 absolute 'base' URI와 통합시킴으로써 'absolute' form로 변환시킨다. 이런 시스템에서는 비록 절대적 레퍼런스의 경우에 그 base가 어떠한 relevance도 갖고 있지 않더라도, URI reference를 base URI의 상대적인 것으로 취급한다. 비록 이것이 그 다큐멘트 내에서 이루어진 선언에 의해서 또는 외부 데이터 전송 프로토콜의 일부로서 무시될 지라도, 전형적으로 base URI는 URI reference를 포함하고 있는 다큐멘트를 식별한다. 만일 base URI가 fragment identifie를 포함하고 있다면, 통합과정동안 그것은 무시된다. 만일 fragment identifier가 URI reference에 존재한다면, 그것은 merging process 동안에 보존된다.

Web document markup languages는 종종 external documents or specific portions of the same logical document와 같은 다른 자원을 지정하기 위하여 URI references를 사용하기도 한다.

2) fragment identifier

이것은 또 다른 primary resource에 종속되어 있는 자원을 말하는 a short string of characters 이다. primary resource는 Uniform Resource Identifier (URI)에 의해 식별되며, fragment identifier는 종속된 자원을 지정한다(point).

hash mark #로 시작되는 fragment identifier 다큐멘트의 URL에 있어서 선택적으로 이루어지며, 이것의 일반적인 syntax는 RFC 3986에서 밝히고 있다. 그러나 URIs에서 hash mark separator는 fragment identifier에 속하진 않는다.

5) Statement reification and context

한 다발의 문장에 의해 모델화된 지식의 body는 각 문장(즉 각 트리플인 주체-술어-객체 모두 함께인)이 URI를 할당 받아서, 예를 들어 “Jane says that John is the author of document X”에서처럼, 추가적 문장을 작성할 수 있는 하나의 자원으로 취급받는 reification(구체화)에 따라야 할 것이다. Reification은 각 문장의 신뢰 수준이나 유용 정도를 파악하기 위하여 때때로 중요하다.

1) Reification

지식표현에 있어서 이것은 다른 주장에 의해서도 언급된 사실적 주장을 다루는 것이다. 또한 이것은 예를 들어 신뢰도를 결정하기 위하여 서로 다른 목격(witnesses)에서 나온 논리적 주장을 비교하기 위하여 어떤 방법으로 조작될 수도 있다.

메시지 "John is six feet tall"은 진실을 포함하고 있는 주장이며, speaker는 그것의 사실성을 말하고 있다. 반명에 the reified statement, "Mary reports that John is six feet tall"는 Mary에게 그러한 책임을 미루는 것이다. 이러한 방식으로, 문장들은 이성적으로 반대를 만들지 않고서는 양립될 수 없다. 예를 들어, 문장 "John is six feet tall" and "John is five feet tall" 은 서로 배타적이므로 양립할 수 없지만, 문장 "Mary reports that John is six feet tall," and "Paul reports that John is five feet tall"은 둘 다 Mary나 Paul이 사실상 부정확하다는 결과적 근거(conclusive rationale)에 의지함으로써, 양립할 수 없는 것은 아니다.

또한 이것은 컴퓨터 프로그램에 대한 추상적 아이디어가 분명한 data model or other object 로 바뀌어가는 과정을 말한다. reification에 의해, 과거에는 implicit, unexpressed, and possibly inexpressible한 어떤 것이 분명하게 공식화되어 conceptual (logical or computational) manipulation에서 이용할 수 있게 된다.

6) Query and inference languages

RDF 그래프용으로 우수한 쿼리 언어는 SPARQL이다 SPARQL은 SQL-유형의 언어이며, W3C에서 추천하고 있다.

> 가상의 온톨로지를 사용하여 아프리카에 있는 국가수도를 나타내는 SPARQL 쿼리의 예:

```
PREFIX abc: <nul://sparql/exampleOntology#> .
SELECT ?capital ?country
WHERE {
   ?x abc:cityname ?capital ;
      abc:isCapitalOf ?y.
   ?y abc:countryname ?country ;
      abc:isInContinent abc:Africa.
}
```

> RDF graphs에 쿼리하는 또다른 비-표준적 방법은 다음과 같다:

◉ RDQL, precursor to SPARQL, SQL-like
◉ Versa, compact syntax (non-SQL-like), solely implemented in 4Suite (Python)
◉ RQL, one of the first declarative languages for uniformly querying RDF schemas

and resource descriptions, implemented in RDFSuite.

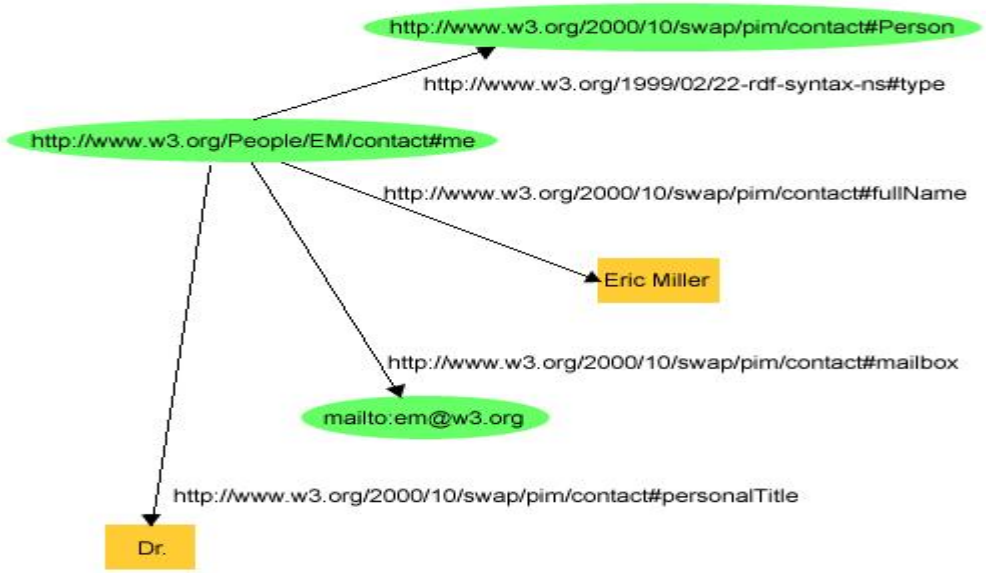
◉ SeRQL, part of Sesame

◉ XUL has a template element in which to declare rules for matching data in RDF. XUL uses RDF extensively for databinding.

> Examples

Example 1: Eric Miller라는 사람의 RDF Description

The following example is taken from the W3C website describing a resource with statements "there is a **Person** identified by **http://www.w3.org/People/EM/contact#me**, whose name is **Eric Miller**, whose email address is **em@w3.org**, and whose title is **Dr.**



Eric Miller를 Describing하고 있는 RDF Graph.

The resource "**http://www.w3.org/People/EM/contact#me**" 는 subject이다.

The objects는 다음과 같다:

"Eric Miller" (with a predicate "whose name is"),
mailto:em@w3.org (with a predicate "whose email address is"), and
"Dr." (with a predicate "whose title is").

The subject is a URI.

The predicates also have URIs. For example, the URI for each predicate:

"whose name is"은 http://www.w3.org/2000/10/swap/pim/contact#fullName,
"whose email address is"은
http://www.w3.org/2000/10/swap/pim/contact#mailbox,
"whose title is"은 http://www.w3.org/2000/10/swap/pim/contact#personalTitle.

추가로, subject는 a type (with URI http://www.w3.org/1999/02/22-rdf-syntax-ns#type)을 가지며, 이것은 person (with URI http://www.w3.org/2000/10/swap/pim/contact#Person)이다.

Therefore, the following "subject, predicate, object" RDF triples can be expressed:

http://www.w3.org/People/EM/contact#me, http://www.w3.org/2000/10/swap/pim/contact#fullName,
"Eric Miller"

http://www.w3.org/People/EM/contact#me, http://www.w3.org/2000/10/swap/pim/contact#mailbox,
mailto:em@w3.org

http://www.w3.org/People/EM/contact#me,
http://www.w3.org/2000/10/swap/pim/contact#personalTitle, "Dr."

http://www.w3.org/People/EM/contact#me, http://www.w3.org/1999/02/22-rdf-syntax-ns#type,
http://www.w3.org/2000/10/swap/pim/contact#Person

In standard N-Triples format, this RDF can be written as:

<http://www.w3.org/People/EM/contact#me>
<http://www.w3.org/2000/10/swap/pim/contact#fullName> "Eric Miller" .

<http://www.w3.org/People/EM/contact#me>
<http://www.w3.org/2000/10/swap/pim/contact#mailbox> <mailto:e.miller123(at)example> .

<http://www.w3.org/People/EM/contact#me>
<http://www.w3.org/2000/10/swap/pim/contact#personalTitle> "Dr." .

<http://www.w3.org/People/EM/contact#me>
<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
<http://www.w3.org/2000/10/swap/pim/contact#Person> .

Equivalently, it can be written in standard Turtle (syntax) format as:

@prefix eric: <http://www.w3.org/People/EM/contact#> .
@prefix contact: <http://www.w3.org/2000/10/swap/pim/contact#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

```
eric:me contact:fullName "Eric Miller" .
eric:me contact:mailbox  <mailto:e.miller123(at)example> .
eric:me contact:personalTitle "Dr." .
eric:me rdf:type contact:Person .
```

Or, it can be written in RDF/XML format as:

```
<?xml version="1.0" encoding="utf-8"?>
<rdf:RDF xmlns:contact="http://www.w3.org/2000/10/swap/pim/contact#“
         xmlns:eric="http://www.w3.org/People/EM/contact#“
         xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <rdf:Description rdf:about="http://www.w3.org/People/EM/contact#me">
    <contact:fullName>Eric Miller</contact:fullName>
  </rdf:Description>
  <rdf:Description rdf:about="http://www.w3.org/People/EM/contact#me">
    <contact:mailbox rdf:resource="mailto:e.miller123(at)example"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://www.w3.org/People/EM/contact#me">
    <contact:personalTitle>Dr.</contact:personalTitle>
  </rdf:Description>
  <rdf:Description rdf:about="http://www.w3.org/People/EM/contact#me">
    <rdf:type rdf:resource="http://www.w3.org/2000/10/swap/pim/contact#Person"/>
  </rdf:Description>
</rdf:RDF>
```

p. ccvii

<제 8장 정보검색의 실제>

*** Marcia J. Bates의 Berrypicking Model**

Bates는 "berrypicking"이 정보검색의 이전 모델보다 이용자가 정보를 어떻게 탐색하는지를 보다 잘 반영하고 있다고 주장하였다. 이러한 주장은 이전의 모델이 엄격한 선형적 모델이어서 인지적 질문을 반영하지 못했기 때문에 아마도 나올 수 있었다. 예를 들어, 한가지의 전형적 모델은 쿼리와 다큐멘트 간의 단순한 선형적 match로 이루어졌다. 그렇지만 Bates는 이러한 과정에 간단한 변화가 필요하다는 것을 지적하였다. 예를 들어, Salton은 user feedback이 search results를 개선하는데 도움을 줄 수도 있다고 주장한 것처럼.

Bates는 탐색이 진행하면 bit by bit로 이루어진다고 주장하였다. 즉, 사람은 정보검색시스템으로부터 얻은 결과에 반응하여 지속적으로 자신의 탐색어를 변화시킨다는 것이다. 그러므로 간단한 선형적 모델은 정보검색의 본질을 수용하지 못하는데, 왜냐하면 탐색의 실재 행동은 찾고 있는 정보의 인지적 모델을 이용자가 변경하도록 하는 feedback의 원인이 되기 때문이다. 게다가 정보검색은 bit by bit로 이루어진다. 그녀는 많은 예를 제시하였으며, 예를 들어 이용자는 footnotes를 살펴본 다음에 그 정보원을 쫓아간다. 그렇지 않다면 이용자는 그

주제에 대한 최신 학술기사를 살펴볼 수도 있다. 각각의 경우에 이용자의 질문은 변할 수 있으며 이렇게 탐색은 진화한다.

고전적인 정보검색모델과는 대조적으로, 딸기줍기 모델은 이용자가 문서를 탐색할 때(아마도 수많은 서로 다른 쿼리로 서로 다른 탐색할 때), 그들은 맛있는 것(유용한 참고자료와 현실적 정보)만 수집하여 저장할 것이라는 가정을 근거로 한다. 이 모델의 중요한 특징은 그것의 목표가 기존의 탐색모델처럼 검색된 최종문헌을 생산하는 것이 아니라, 그것보다는 하나씩 하나씩 딸기(이삭)줍기를 통하여 "딸기"란 정보를 수집하는 것이다.

딸기줍기모델을 소개한 논문에서, Marcia Bates는 또한 진화적 탐색의 개념을 소개하였다. 정보검색의 고전 모델에서, 이용자는 불변의 정보요구를 쿼리로 표현하고자 하지만, 진화적 탐색에서는 이용자가 쿼리를 수행함으로써 최종결과로 어떤 다큐멘트들을 얻어가면서, 동시에 자신들의 정보요구를 실재로는 좀 변경해야한다는 것을 깨닫게 된다. 그런 다음에 이용자는 자신들의 정보요구를 보다 잘 표현하고 그뿐만 아니라 자신들의 정보요구 자체가 이젠 변했음으로 자신들의 쿼리를 재조정 한다. 이러한 상황을 통하여 이용자는 "다큐멘트 또는 정보"의 딸기들이 보다 확대된 연속적인 탐색을 통해서 서로 다른 시기에 수집된다는 것을 알게 된다.

따라서 이 모델은 누구나 최종 쿼리에 의해 탐색절차의 말미에 얻게 되는 다큐멘트들의 집단에 자신이 원하는 모든 것이 포함될 때까지 스스로 탐색과정을 간단하게 재조정하면서 진행시킨는 것이다.

기존의 정보검색 모델과의 두 가지 차이점은 다음과 같다:

a) 정보요구는 단편적이고 불변적인 것이 아니라 탐색과정을 통해 진화한다.
b) 탐색의 결과는 최종 쿼리에 의해 검색된 다큐멘트의 세트가 아니라, 그 과정을 거치면서 딸기줍기식으로 검색된 다큐멘트, 참고자료, 정보 이다.

1) Exploratory(조사) Search

human-computer interaction and cognitive science의 연구자들은 WWW와 상호작용할 때 사람들이 어떻게 정보를 탐험(explore)하는가에 초점을 맞추고 있다. 이런 종류의 탐색은 때때로 exploratory search라고 부르며, 사람들이 어떻게 자신들의 탐색행동을 반복적으로 세련(refine)시켜서 탐색문제에 대한 자신들의 내적 표현을 갱신시키는지에 초점을 맞추고 있다. 기존의 탐색엔진들은 인터페이스를 통하여 기본적인 사실과 간단한 정보를 검색하는 것과 관련된 전통적인 도서관학에 근거하여 디자인되었다. 그렇지만 exploratory information retrieval에서는 종종 잘못 정의된 탐색목적과 적합성 평가를 위한 발전된 기준 등이 포함하기도 한다. 인간과 정보시스템 간의 상호작용은 그러므로 더 많은 인지적 행동을 포함시켜야할 것이며, exploratory search을 지원하는 시스템은 역동적인 정보검색과정이 진행되는 동안 관련된 인지적 복잡성을 고려해야할 것이다.

2) Natural language searching

정보검색을 도울 수 있는 또 다른 인지적 정보모델은 natural language searching 이다. 예를 들어, American commercial edutainment website인 How Stuff Works에서는 영화를 찾아서 비평을 보고, 그 다음엔 멕시코 식당을 찾아서 비평을 읽는 것보다는 단지 브라우저에 ""I want to see a funny movie and then eat at a good Mexican restaurant. What are my options?"를 타이핑함으로써 유익하고 적절한 해답을 얻는 세계를 상상할 수 있다. 비록 그러한 일이 오늘날 가능하지는 않지만, 이것은 정보검색의 인지모델로서는 holy grail과 같은 것이다. 이것의 목표는 다소 정보검색 프로그램으로 하여금 자연어탐색에 응답하도록 프로그램을 짜는 것이며, 이를 위해서는 사람들이 쿼리를 구조화하는 방법에 대하여 더 많이 이해하여야 할 것이다.

*** JISC IE(Joint Information Systems Commitee Information Environment**

Jisc(formerly the Joint Information Systems Committee, and still commonly

referred to as JISC)는 learning, teaching, research 그리고 administration 분야에서 정보통신기술의 사용에 대한 leadership을 제공함으로써, post-16 and higher education, 그리고 research를 지원하는 역할을 담당하고 있는 a United Kingdom non-departmental public body 이다.

사람들이 요구하는 정보자원 - books, journals, research papers, teaching resources, videos, maps 등 - 은 매우 다양하며, 그것들이 어떤 포맷으로 되어 있더라도 점차적으로 디지털화되고 있다.

Information Environment (IE)는 사람들로 하여금 자신들의 학습, 교수, 또는 연구와 관련된 정보를 효율적이고 효과적으로 발견하고 관리할 수 있는 서비스를 개발하고 제공하는 JISC의 업무를 말하는 용어이다.

> What does the Information Environment mean in practice?

- ◉ national resource discovery tools such as the Archives Hub which provides convenient access to information about unique research collections distributed across the UK
- ◉ software protocols such as SWORD (Simple Web Service Offering Repository Deposit) which enables files to be easily deposited in digital repositories from within other applications
- ◉ 'technical' infrastructure such as the OpenURL router service at EDINA which enables linking between bibliographic records and the electronic or other copy of the item referenced to which a user's home institution has access
- ◉ centres of expertise such as the Digital Curation Centre
- ◉ practical guidance such as a methodology for the analysis and costing of the lifecycle of digital objects

1) EDINA

인터넷에서 전달된 data applications를 제공하는 UK-based data centre (funded by the Joint Information Systems Committee - JISC)이며, 이것의 목표는 주로 영국의 Higher Education staff and students 이다.
(이 글에서 "data centre"란 인터넷에서 직접적으로 다운로드하거나 접근하여 조작할 수 있는 특별한 datasets를 제공하는 기관을 말한다. 이것 이외에 영국에서 또 다른 주요한 데이터센터는 MIMAS 와 UKDA 이다. 이것은 또한 네트워크에서 데이터의 전달에 관한 research and development (R&D) projects를 수행하고 있으며, 비록 국가자금을 지원받더라도, (그것의 a division of Information Services인) the University of Edinburgh에 의해 운영되고 있다.

2) Mimas

이것은 영국의 University of Manchester에 근거를 둔 a nationally designated academic data centre 이다. 이것의 임무는 knowledge, research, and teaching의 발전을 지원하는 것이며, 수 많은 연국의 연구정보자산을 관리하고 있으며, 사람들이 이러한 자원엥 접근하는데 도움을 주는 어플을 만들고 있다. 이 기관은 Jisc와 오랫동안 관계를 유지하고 있으며 특히 the Economic and Social Research Council과 같은 연구 위원회와 강한 유대를 가지고 있다.

3) UK Data Archive

이것은 영국에서 data archiving분야에 대한 전문기술을 갖춘 a national centre 이다. 영국의 사회과학과 인문학에 관한 가장 커다란 디지털 데이터 콜렉션을 보관하고 있으며, 신뢰할 수 있는 디지털 저장고로서 the Data Seal of

Approval의 인증을 받았으며, 또한 정보보안을 위한 국제 ISO 27001 기준에 따른 인증을 받았다.

>> Current programme activities of the Information Environment
◉ exploring how digital repositories of research outputs can be made easier to use
◉ putting digital preservation into practice
◉ increasing understanding of how metadata for digital resources can be created automatically.

<제 9장 검색 인터페이스>

*** ORBIT : Questel-Orbit**

Questel-Orbit는 지적재산권에 관련된 특별한 공급자이다. 이것은 특허 데이터베이스 장서, 상표 데이터베이스, 그리고 CAS, COMPENDEX(engineering), INSPEC(scientific and technical literature의 주요색인 데이터베이스), PASCAL(과학서지데이터베이스이며 유럽의 science, technology and medicine분야에 있는 핵심적 과학문헌을 취급한다)에 포함되어 있는 비-특허 자료도 제공한다. 이것의 콘텐트는 예를 들어, Dialog에 의해 데이터 장서와 함께 부분적으로 중복되어 제공된다.

*** Chemical Abstracts Service (CAS)**

이것은 a division of the American Chemical Society이며, 화학정보의 원천이고, 미국의 오하이오 주의 Columbus에 있다.

> 인쇄 정간물

Chemical Abstracts는 최근에 출판된 과학 다큐멘트에 발표(disclosures)된 summaries and indexes를 제공하는 a periodical index 이며, 27개 국가와 두 개의 국제기관에서 나온 특허내용과 더불어 약 50개의 언어로 된 약 8,000 journals, technical reports, dissertations, conference proceedings, and new books을 매년 모니터하고 있다. 그러나 Chemical Abstracts는 ceased print publication on January 1, 2010년 1월 1일자로 인쇄출판을 중지하였다.

>Databases

서로 다른 products를 지원하는 두 가지 중요한 데이터베이스가 있다: CAplus and

Registry.

◉ CAplus

CAplus는 전세계의 화학지에 있는 모든 기사에 대한 bibliographic information and abstracts, 그리고 모든 과학지, 특허, 및 기타 과학출판물에서 나온 화학-관련 기사들로 구성되어 있다.

◉Registry

Registry는 more than 71 million organic and inorganic substances, and more than 64 million protein and DNA sequences에 관한 정보를 포함하고 있다. DNA 배열정보는 National Institutes of Health에서 생산되며, CAS and GenBank에서 가져온다. 화학정보는 CAS에서 생산되며, 화학구조에 대한 특수한 CAS registry number, index name, and graphic representation으로 이루어진 각각의 복합물을 식별할 수 있도록 the CAS Registry System에 의해 준비된다. 화학적 이름의 배정은 chemical nomenclature rules for CA index names에 따라 이루어지지만, CA index names는 IUPAC(International Union of Pure and Applied Chemistry) 의 규칙에 따라 국제적으로 표준화된 IUPAC names와는 약간 차이가 난다.

> Products

CAS databases는 두 가지의 중요한 데이터베이스 시스템인 STN 과 SciFinder에서 이용할 수 있다.

◉ STN (Scientific & Technical Information Network) International

이것은 CAS and FIZ Karlsruhe에서 공동으로 운영하고 있으며, 명령어 방식의 인터페이스를 사용함으로써 기본적으로 정보전문가를 목표로 하고 있다. CAS databases와 더불어, STN 또한 Dialog와 같은 다른 많은 데이터베이스에 대한 접근을 제공하고 있다.

◉ SciFinder

SciFinder는 chemical and bibliographic information의 데이터베이스 이며, client application의 web version은 graphics interface을 사용하여 chemical structures and reactions을 탐색할 수 있도록 하였다.

◉CASSI

CASSI는 Chemical Abstracts Service Source Index의 두문자어 이다. CASSI는 선택된 학술지의 titles and abbreviations, CODEN, ISSN, publisher, and date of first issue (history)를 제공하고 있으며, 또한 its language of text and language of summaries를 포함하고 있다.

*** Inspec**

Inspec은 과학 기술 문헌의 중요한 색인 데이터베이스이며, the Institution of Engineering and Technology (IET)와 전에는 IET'의 forerunners 중의 하나인 the Institution of Electrical Engineers (IEE)에서 출판하고 있다.

Inspec에서는 coverage is extensive in the fields of physics, computing, control, and engineering과 같은 포괄적 분야를 다루고 있으며, 그것의 주제는 astronomy, electronics, communications, computers & computing, computer science, control engineering, electrical engineering, information technology, physics, manufacturing, production and mechanical engineering 등이다.

*** PASCAL**

이것은 INIST (CNRS)에서 관리하고 있는 과학적 서지 데이터베이스 이다. PASCAL은 특별히 유럽 자료에 초점을 맞추어 science, technology and medicine 분야의 핵심 문헌을 취급하고 있다.

As of 2012, PASCAL maintains a database of more than 17 million records, 90% of these are author abstracts. Its coverage is from 1973 to present. Its source documents are composed of journal articles at 88% (3,085 international titles), proceedings at 9%, and dissertations, books, patents, and reports account combined for 3%.

*** Dialog and the invention of online information services**

이것은 1972년 설립된 세계 최초의 상업적 온라인 탐색 서비스이다. Dialog는 최고급 콘텐트 정보원의 선도적 제공자로 오랫동안 자랑스러운 역사를 가지고 있다.

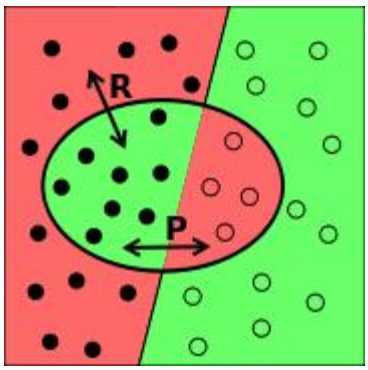
현재는 ProQuest의 일부로서, 이것은 전문가의 정보요구를 충족시키기 위한 콘텐트와 탐

색기능을 제공하는데 초점을 맞추고 있다. 전세계에 있는 corporate, business and government settings의 연구자를 위하여, ProQuest Dialog는 중요한 의사결정을 지원하고 경쟁력 있는 장점을 마련하고 혁신을 이끌기 위한 권위있는 대답을 전달하고 있다.

STN International은 an online database service로서 that provides global access to published research, journal literature, patents, structures, sequences, properties, and other data에 대한 범세계적인 접근을 제공하고 있지만, Questel-Orbit의 분명하고도 절대적인 장점은 patent and trademark information 이다. Questel-Orbit''s retrieval language의 힘은 Dialog와 STN International의 그것과 견줄만하다. patent full-texts를 포함하여 Questel-Orbit에서는 무료로 특허정보를 제공하며, 국가별 그리고 국제 특허 사무소(예, the European Patent Office)에서 이용할 수 있다.

<제 10장 정보검색 시스템의 평가>

*** Precision and recall**



Recall-precision

위의 그림에서 적합한 아이템들은 직선의 왼쪽에 있는 반면에, 검색된 아이템들은 타원형 안에 포함되어 있으며, 붉은 지역은 에러를 나타낸다. 왼쪽 편에는 검색되지 않은 적합한 아이템들(false negatives)이 있으며, 반면에 오른쪽에는 적합하지 않은 검색된 아이템들(false positives)이 있다. 패턴 인식과 정보검색 분야에서, 정확률(또는 positive predictive value라고도 함)은 적합해서 검색된 경우의 단편(fraction)인 반면에, 재현율(또는 sensitivity로 알려져 있음)은 검색된 적합한 경우의 단편이다. 그러므로 정확률과 재현율 둘 다 적합성의 이해와 척도의 근거이 된다.

여러 화면(scenes)에서 개들을 인식하는 프로그램이 개 9마리와 고양이 몇 마리가 있는 하나의 화면에서 개 7마리를 찾았다고 가정해 보자. 찾아낸 것 중에서 4마리는 정확하지만 3

마리가 실재로는 고양이라면, 이 프로그램의 정확률은 4/7인 반면에 재현율은 4/9이다. 탐색 엔진이 단지 20페이지만 적합한 30페이지를 제공하고 추가적으로 적절한 40페이지를 제공하는데 실패한다면, 그것의 정확률은 20/30 = 2/3 이지만, 그것의 재현율은 20/60 = 1/3 이 된다.

간단히 말해서, 높은 재현율에서는 그 알고리즘이 대부분이 적합한 결과를 제공하지만, 높은 정확률에서는 그 알고리즘이 본질적으로 부적합한 것보다는 더 많은 적합한 결과를 제공한다는 것을 알 수 있다.

*** Relevence**

적합성의 개념은 인지과학, 논리학, 문헌정보학과 같은 많은 분야에서 연구되고 있으나, 가장 기본적으로 이것은 인식론(지식의 이론)에서 주로 연구되고 있다. 지식에 대한 다양한 이론들은 적합한 것이 무엇인가에 대하여 여러 가지 주장을 펴고 있으며, 이러한 기본적인 주장들은 나머지 분야와도 연관성을 갖고 있다.

> 정의

"만일 어떤 것이 T를 암시하는 목표(G)를 완성할 수 있는 가능성을 높인다면, 그 어떤 것(A)은 임무(T)에 적절하다." (Hjørland & Sejer Christensen,2002).

> 문헌정보학

이 학문에서는 데이터베이스로부터 다큐멘트를 검색할 때 적절한지 또는 적절치 않은지를 고려하며, relevance의 개념을 정의할 때, 두 가지의 척도를 사용한다: Precision and recall:

Recall = a : (a + c) X 100%,
where a = number of retrieved, relevant documents,
c = number of non-retrieved, relevant documents (sometimes termed "silence").

재현율은 그러므로 다큐멘트의 탐색이 얼마나 망라적으로 이루어졌는가를 표현한 것이다.

Precision = a : (a + b) X 100%,
where a = number of retrieved, relevant documents,
b = number of retrieved, non-relevant documents (often termed "noise").

정확율은 그러므로 다큐멘트-검색에서 노이즈의 양에 관한 척도이다.

*** 확률모델**

특정한 쿼리로 근거로 각 다큐멘트가 해당 쿼리에 적합할 확률을 베이지언 룰을 활용하여 계산될 수 있다는 가정을 전제로, 비-연관 다큐멘트들이 쿼리에 포함될 확률과 연관 다큐멘트들이 쿼리에 포함될 확률을 계산하여 필요한 다큐멘트를 찾는 모델링이다.

> 장점 : 다큐멘트들이 쿼리에 대하여 적합한 확률의 순서에 따라 내림차순으로 랭크된다.
> 단점 : 비-연관 다큐멘트와 연관 다큐멘트 집단의 초기 결과 집단을 가정해야만 한다.
불리안 모델과 같이 가중치가 없어서 색인어의 빈도수에 대한 가중치를 부여할 수가 없다.
색인어들에 대한 상호 독립 가정을 전제로 한다.

1) **Bayesian probability**
이것은 확률의 개념에 대한 서로 다른 해석 중의 하나이며, evidential probabilities의 범주에 속한다. The Bayesian interpretation of probability은 참과 거짓의 불확실한 제안을 가지고 증명(reseasoning)할 수 있는 propositional logic을 확장한 것으로 여겨질 수 있다. hypothesis의 확률을 평가하기 위하여, the Bayesian probabilist는 어떤 이전의 확률을 특정화 한 다음에, 새롭고 적절한 데이터의 모습으로 갱신한다.
한다.

The Bayesian interpretation은 이러한 계산을 하기 위하여 표준적인 procedures and formulae를 제공한다. 어떤 현상에 대한 he "frequency" or "propensity(경향)"으로 확률을 이석하는 것과는 대조적으로, Bayesian probability은 “우리가 지식의 상태 또는 믿음의 상태를 표현할 목적으로 이론적으로 할당한 quantity”이다. In the Bayesian view에서, a probability는 a hypothesis으로 설정되는 반면에, the frequentist view에서, a hypothesis 은 전형적으로 a probability로 설정됨이 없이 테스트 된다.

*** Markov model**

확률이론에서, Markov 모델은 무작위로 변하는 시스템을 모델화하기 위하여 사용되는 stochastic(추측통계적) model - 미래의 상태가 현재의 상태에만 의존하지 그것을 앞서가는 사건의 sequence에는 의존하지 않는다고 가정하는 모델 - 이다(즉, 이것은 Markov property를 가정한다). 일반적으로 이러한 가정은 다른 방법으로는 다루기 힘든 모델에 대한 reasoning and computation을 가능하게 한다.

1) Markov property
이것은 추측통계 과정의 memoryless property의 성질이다. 예를 들면, 자동차 엔진의 수명을 X 라고 가정해 보자. 만일 그 엔진이 20만 마일의 수명을 갖고 있다면, 우리는 직관적으로 처음의 10만마일의 엔진과 그 다음의 10만마일의 엔진이 똑같지 않다는 것을 알게 된다. 그렇지만, memorylessness는 두 개의 확률이 동일하다고 말한다. 본질적으로, 우리는 자동차의 현 상태에 대해서는 ‘forget’하며, 바꿔 말해서 그 확률들은 얼마나 많은 시간이 경과했는지에 대하여 영향을 받지 않는다

2) a stochastic (/stoʊˈkæstɪk/) process, or sometimes random process (widely used)
이것은 시간이 지나면서 어떤 시스템의 진화를 표현하는 a collection of random variables 이다.

> Introduction
모든 연속적 상태가 관찰되든지 말든지, 또는 시스템이 관찰결과를 근거로 조정을 받든지

말든지에 따라 서로 다른 상황에서 이용하는 4 가지의 일반적인 Markov models이 있다:

	observable	System state is partially observable
System is autonomous	Markov chain	hidden Markov model
System is controlled	Markov decision process	partially observable Markov decision process

> Markov chain

이것은 가장 간단한 Markov model이며, 시간이 변하면서 랜덤 변수들이 있는 시스템의 상태를 모델화한다. 이런 맥락에서, Markov property는 이러한 변수의 분산이 이전 상태의 분산에만 의존한다고 여겨진다.

> Hidden Markov model

이것은 그 상태가 단지 부분적으로만 관찰 가능한 Markov chain 이다. 다른 말로해서, 관찰들은 그 시스템의 상태와 관련이 있지만, 전형적으로 그 상태를 정확하게 결정하는 데는 충분치 않다.

여러 가지 잘 알려진 hidden Markov models이 있다: For example, given a sequence of observations, the Viterbi algorithm will compute the most-likely corresponding sequence of states, the forward algorithm will compute the probability of the sequence of observations, and the Baum-Welch algorithm will estimate the starting probabilities, the transition function, and the observation function of a hidden Markov model.

One common use is for speech recognition, where the observed data is the speech audio waveform and the hidden state is the spoken text. In this example, the Viterbi algorithm finds the most likely sequence of spoken words given the speech audio.

> Markov decision process

A Markov decision process는 state transitions가 현 상태와 시스템에 적용된 action vector에 의존하는 Markov chain 이다. 전형적으로 이것은 기대되는 보상과 관련해서 어떤 utility를 최대화할 수 있는 행동들의 정책을 계산하는데 사용된다.

> Partially observable Markov decision process

A partially observable Markov decision process (POMDP)는 시스템의 상태가 단지 부분적으로만 관찰되는 Markov decision process 이다. 최근에는 approximation techniques이 만들어져서 controlling simple agents or robots와 같은 다양한 어플에 유용하게 사용되고 있다.

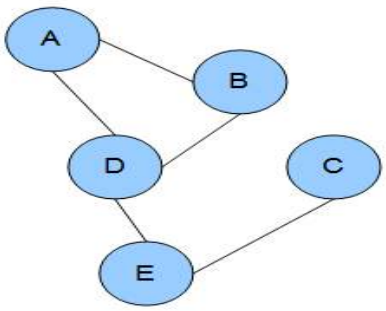
> Markov random field

A Markov random field는 복수의 차원에 존재하는 일반화된 Markov chain 이라고

여겨지고 있다. Markov chain의 state는 시기적으로 이전의 상태에만 의존하고 있는 반면에, Markov random field에서 각 상태는 다수의 방향 중에서 어떤 방향에 있는 그것의 이웃에 의존한다.

Markov random field (often abbreviated as MRF), Markov network or undirected graphical model는 undirected(목표가 불분명한) graph로 묘사된 Markov property를 가지고 있는 랜덤 변수의 세트이다. 이것은 의존성을 표현하는데 있어서 Bayesian network과 비슷하다; 이것들 간의 차이는 Bayesian networks이 directed and acyclic(비-주기적, 비-순환적)한 것인 반면에, Markov networks는 undirected 하고 cyclic할 수도 있다는 것이다. 그러므로 Markov network는 cyclic dependencies처럼 Bayesian network에서 할 수 없는 어떤 의존성을 표현할 수 있다. 반면에, 이것은 induced dependencies처럼 Bayesian network에서는 할 수 있는 어떤 의존성을 표현할 수 없다.

Markov random field example

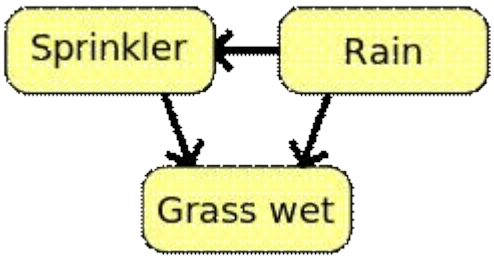


<<An example of a Markov random field. Each edge represents dependency. In this example: A depends on B and D. B depends on A and D. D depends on A, B, and E. E depends on D and C. C depends on E.>>

* **A Bayesian network**, Bayes network, belief network, Bayes(ian) model or probabilistic directed acyclic graphical model

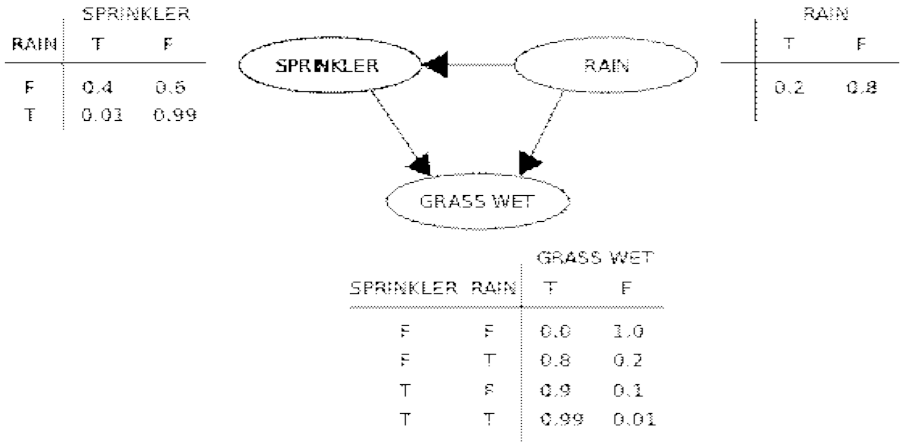
이것은 directed acyclic(비주기적) graph (DAG)를 통해 자신들의 조건적 의존성과 랜덤 변수의 세트를 표현하는 통계 모델의 일종인 probabilistic graphical model 이다. 예를 들어, Bayesian network는 질병과 증상 간의 확률적 연관성을 표현할 수 있다. 증상이 제공되면, 그 네트워크는 다양한 질병의 출현 확률을 계산하는데 사용될 수 있다.

또 다른 예로, 잔디를 적실 원인인 두 가지 사건이 존재한다고 가정해 보자: 하나는 sprinkler를 킨 것이고, 또 하나는 비가 오는 것이다. 또한 비가 스프링클러의 사용에 직접적인 영향을 끼친다고 가정해 보자(즉 비가 올 때, 스프링클러는 대체로 꺼져 있다). 그런 다음에 이 상황을 아래처럼 Bayesian network로 모델화할 수 있다. 모두 3개의 변수가 두 개의 가능한 값, T (for true) and F (for false)을 갖는다.



<<A simple Bayesian network>>

Rain influences whether the sprinkler is activated, and both rain and the sprinkler influence whether the grass is wet.



The joint probability function is:

$$\mathrm{P}(G, S, R) = \mathrm{P}(G|S, R)\mathrm{P}(S|R)\mathrm{P}(R)$$

where the names of the variables have been abbreviated to *G = Grass wet (yes/no)*, *S = Sprinkler turned on (yes/no)*, and *R = Raining (yes/no)*.

The model can answer questions like "What is the probability that it is raining, given the grass is wet?" by using the conditional probability formula and summing over all nuisance variables(nuisance variables는 확률모델에 기본적인 랜덤 변수이지만, 그 자체로는 특별한 관심을 받지 못하거나 더 이상 관심의 대상이 아닌 변수 이다) :

$$\mathrm{P}(R = T \mid G = T) = \frac{\mathrm{P}(G = T, R = T)}{\mathrm{P}(G = T)} = \frac{\sum_{S \in \{T,F\}} \mathrm{P}(G = T, S, R = T)}{\sum_{S,R \in \{T,F\}} \mathrm{P}(G = T, S, R)}$$

Using the expansion for the joint probability function $\mathrm{P}(G, S, R)$ and the conditional probabilities from the conditional probability tables (CPTs) stated in the diagram, one can evaluate each term in the sums in the numerator and denominator. For example,

$$\begin{aligned}
&\mathrm{P}(G = T, S = T, R = T) \\
&\quad = \mathrm{P}(G = T | S = T, R = T)\mathrm{P}(S = T | R = T)\mathrm{P}(R = T) \\
&\quad = 0.99 \times 0.01 \times 0.2 \\
&\quad = 0.00198.
\end{aligned}$$

Then the numerical results (subscripted by the associated variable values) are

$$\begin{aligned}
\mathrm{P}(R = T \mid G = T) &= \frac{0.00198_{TTT} + 0.1584_{TFT}}{0.00198_{TTT} + 0.288_{TTF} + 0.1584_{TFT} + 0.0_{TFF}} \\
&= \frac{891}{2491} \approx 35.77\%.
\end{aligned}$$

<제 11장 차세대 정보검색>

*** 시멘틱 웹(Semantic Web)**

시멘틱 웹은 W3C와 같은 국제 표준 기구들이 이끌어 가는 협업적 활동(movement) 이며, 이것의 표준은 웹에서 공동의 데이터 포맷을 장려하고 있다. 웹페이지에 어의적 콘텐트를 포함시키도록 함으로써, 시멘틱 웹은 비-구조적이고 유사-정형화된 다큐멘트가 지배적인 현재의 웹을 "web of data"로 변경시키는 것을 목표로 하고 있다. 또한 이것의 stack(서고, 산더미)은 W3C의 RDF로 작성된다.

W3C에 따르면, "시멘틱 웹은 application, enterprise, and community boundaries 간에 데이터를 공유하고 재사용하도록 하는 공동의 프레임워크를 제공"하는 것이며, 이 용어는 컴퓨터로 처리하는 a web of data 용으로 Tim Berners-Lee에 의해 만들어졌다.

> Purpose

시멘틱 웹의 주요 목적은 이용자로 하여금 보다 더 쉽게 정보를 찾고, 공유하고, 결합할 수 있도록 함으로써 현재의 웹을 발전시키는 것이다. 인간은 웹을 사용하여 "twelve months"의 에스토니아 번역본을 찾고, 도서관 책을 예약하고, 가장 값싼 DVD를 찾는 것과 같은 업무를 수행할 수 있다. 그렇지만, 컴퓨터는 인간의 지시가 없다면 모든 이러한 일을 할 수가 없다. 왜냐하면 웹페이지들은 사람이 읽을 수 있도록 디자인 되어야지 컴퓨터를 위한 것이 아니기 때문이다. 그러나 컴퓨터는 웹에서 정보를 찾고, 결합하고, 관련된 행동을 하는 것을 포함하여 보다 많은 지루한 일을 할 수 있다.

1) The **Twelve Months** is a Greek fairy tale collected by Georgios A. Megas in Folktales of Greece.
A young and beautiful girl is sent into the cold forest in the winter to perform impossible tasks. She must get violets and apples in midwinter. She meets the 12 months personified who help her. The step mother and sister take the items, without a word of thanks. When the evil stepsister comes and is rude, they disappear, taking their fire, and leaving the stepsister cold and hungry.....

시멘틱 웹은 컴퓨터가 의미를 근거로 복잡한 인간의 리퀘스트들을 "이해"하고 반응하는 하나의 시스템이며, 이 같은 "이해"는 어의적으로 구조화된 적합한 정보원을 필요로 한다.

Tim Berners-Lee가 시멘틱 웹에 대한 전망:

I have a dream for the Web become capable of analyzing all the data on the Web – the content, links, and transactions between people and computers. A "Semantic Web", which makes this possible, has yet to emerge, but when it does, the day-to-day mechanisms of trade, bureaucracy and our daily lives will be handled by machines talking to machines. The "intelligent agents" people have touted(손님을 끌다) for ages will finally materialize.

> Limitations of HTML

전형적으로 컴퓨터의 많은 파일들은 크게 봐서 human readable documents와 machine readable data로 나눌 수 있다. mail messages, reports, and brochures 같은 다큐멘트는 인간에 의해 읽혀지며, calendars, addressbooks, playlists, and spreadsheets 같은 데이터는 그것들을 볼 수 있고, 탐색할 수 있고, 연결할 수 있는 응용 프로그램을 사용하도록 제공된다.

현재, 웹은 이미지와 쌍방형 폼과 같은 멀티미디어 사물이 산재해 있는 텍스트를 코딩하는 markup convention인 HTML로 작성된 다큐멘트를 주요한 근거로 삼고 있다. 메타데이터 태그들은 예를 들어 다음과 같이 컴퓨터가 웹 페이지의 콘텐트를 범주화할 수 있는 방법을 제공한다:

```
<meta name="keywords" content="computing, computer studies, computer" />
<meta name="description" content="Cheap widgets for sale" />
<meta name="author" content="John Doe" />
```

HTML과 이것을 제공하는 도구(웹브라우저 소프트웨어, 기타 유저 에이전트)를 가지고, 누구나 판매용 아이템을 리스트하고 있는 페이지를 만들 수 있다. 이 카탈로그 페이지의 HTML으로 “this document's title is 'Widget Superstore'"와 같은 간단한 document-level assertions(주장)을 만들 수 있지만, 그 HTML 자체는 예를 들어, 아이템 번호 X586172가 도매가격 199 유로인 책 서명 ”Acme Gizmo“라고 명백하게 언급하거나 그것이 소비자용 제품이라고 밝힐 능력을 가지고 있지는 않다. 그 보다는 HTML은 단지 텍스트 ”X586712“의 spam(범위)가 "Acme Gizmo" 그리고 "€199", etc. 근방에 있는 어떤 것이라고만 표현할 수 있다. 따라서 이것은 ”this is a catalog“라고 표현하거나 심지어 ”Acme Gizmo“가 타이틀의 일종이라거나, 또는 ”€199“는 가격이라는 것을 표현할 방법이 전혀 없다. 이것뿐만 아니라 그 밖의 여러 가지 정보를 함께 묶어서 그 페이지에 리스트 되어 있는 다른 아이템들과 확실하게 구분할 수 있도록 설명할 방법도 전혀 없다.

어의적으로 HTML은 직접적으로 layouts의 details를 세밀하게 지정하는 것보다는 의미를 수반하는 마크업의 전통적인 업무를 수행한다. 예를 들어, 의 사용은 이탤릭체를 지정하는 <i>보다 “emphasis”를 나타낸다. 이러한 layouts의 details는 Cascading Style Sheets와 결합함으로써 브라우저에 적용되지만, 이러한 업무는 판매용 아이템이나 가격과 같은 사물에 대한 어의를 지정하는 데 있어서는 어려움을 겪는다.

마이크로포맷은 HTML 구문식을 확대하여 사람, 조직, 사건, 제품과 같은 것에 대하여 기계가독형 어의적 마크업을 만들 수 있도록 하고 있다. 유사한 계획에는 RDFa, Microdata and Schema.org.가 포함되어 있다.

1) A **microformat** (sometimes abbreviated μF)

이것은 RSS와 같은 (X)HTML을 지원하는 웹 페이지나 기타 콘텍스트에 있는 메타데이터와 기타 속성을 전달하기 위하여 HTML/XHTML tags를 재사용하려는 semantic markup의 a web-based approach 이다. 이 시도에서는 소프트웨어로 하여금 최종 이용자가 의도한 정보(contact information, geographic coordinates, calendar events, and similar information)를 자동적으로 처리하도록 한다.

As of 2010, microformats allow the encoding and extraction of events, contact information, social relationships and so on. Established microformats such as hCard are published on the web more than alternatives like schema (microdata) and RDFa.

2) **RDFa** (or Resource Description Framework in Attributes)
이것은 웹 다큐멘트에 들어 있는 풍부한 메타데이터에 대한 HTML, XHTML 그리고 여러 가지 XML-based document types에 한 세트의 확장된 attribute을 추가시킨 W3C Recommendation 이다. The RDF data-model mapping은 XHTML documents에서 embedding RDF subject-predicate-object expressions용으로 사용할 수 있으며, 또한 compliant user agents에 의해 RDF model triples을 발췌하는 것도 가능케 한다.

The RDFa community에서는 tools, examples, and tutorials을 host하기 위한 wiki website를 운영하고 있다.

3) **Microdata**
이것은 웹 페이지의 기존 콘텐트에서 메타데이터를 nest하는데 사용되는 WHATWG HTML specification 이다. 탐색엔진, web crawlers, and browsers는 웹 페이지로부터 Microdata를 발췌하여 처리할 수 있으며, 이용자를 위하여 a richer browsing experience을 제공하는데 이것을 사용할 수 있다. Search engines은 이러한 정형화된 구조의 데이터에 직접 접근하는 것으로부터 커다란 도움을 받는데, 그 이유는 서치엔진으로 하여금 웹 페이지의 정보를 이해하고 이용자에게 보다 적절한 정보를 제공할 수 있도록 하기 때문이다.

Microdata는 그것의 속성에 값을 할당하기 위하여 an item and name-value pairs을 묘사하도록 a supporting vocabulary를 사용한다. Microdata는 RDFa 와 microformats을 사용하는 유사한 시도보다 machine-readable tags가 있는 annotating HTML elements를 보다 간편하게 사용하려는 방법을 제공하려는 시도이다.

4) **Schema.org**
이것은 "웹 페이지에서 정형화된 data markup을 위한 공동의 schema set를 만들어 지원하고자 하는" Bing, Google and Yahoo! (the operators of the then world's largest search engines)에 의해 2011년에 시작된 an initiative 이며, 그 해 11월에 Yandex (whose search engine is the largest one in Russia)가 참여하였다. 이들은 메타데이터를 갖고 있는 웹사이트의 콘텐트를 mark up하기 위하여 자신들의 ontology와 HTML5에 있는 Microdata의 사용을 제안하고 있다. 그 같은 markup은 search engine spiders and other parsers에 의해 인지될 수 있으므로, 그 사이트의 meaning에 접근할 수 있다는 것이다.

> Semantic Web solutions

더욱이 시멘틱 웹은 솔루션이 있는데, 여기에는 Resource Description Framework (RDF), Web Ontology Language (OWL), and Extensible Markup Language (XML)처럼 데이터용으로 특별하게 설계된 언어로 출판하는 것이 포함 된다. HTML은 이들 간의 다큐멘트와 링크들을 기술하는데 사용되지만, 대조적으로 RDF, OWL 그리고 XML은 사람, 회의 또는 비행기 부품과 같은 임의의 사물을 묘사하는데 사용된다.

이런 technologies는 웹 다큐멘트의 콘텐트를 보완하거나 대체하는 descriptions를 제공하기 위하여 결합되기도 한다. 따라서 콘텐트는 웹으로 접근이 가능한 데이터베이스에 저장된 descriptive data처럼, 또는 (또는 특히 XML이 점재되어 있는 XTHML로 된, 또는 별도로 저장된 cues를 제공하거나 레이아우트을 갖춘 XML로 된) 다큐멘트 내의 markup 처럼, 스스로 manifest 될 수 있다. 기계가독형 descriptions을 통하여 content managers는 콘텐트의 의미를 우리가 알고 있는 지식의 구조를 추가로 표현할 수 있도록 한다. 이런 방식으로, 컴퓨터

는 텍스트 대신에 지식을 처리하는데 있어서 인간의 연역법과 귀납법과 비슷한 처리과정을 사용하여 의미 있는 결과를 얻을 수 있고, 그럼으로써 컴퓨터를 사용하여 자동정보수집과 연구조사를 수행할 수 있는 것이다.

a non-semantic web page에서 사용될 수 있는 tag의 예:

```
<item>blog</item>
```

semantic web page에서 유사한 정보를 암호화하는 것은 다음과 같을 것이다:

```
<item rdf:about="http://example.org/semantic-web/">Semantic Web</item>
```

Tim Berners-Lee는 Linked Data의 결과로 발생한 네트워크를 HTML-based World Wide Web과 대비하여 Giant Global Graph라고 불렀다. Berners-Lee는 만일 과거가 document sharing이었다면, 미래는 data sharing이라고 주장하면서, “how”라는 질문에 대하여 다음과 같은 3가지의 답을 제시하였다;

1) a URL should point to the data.
2) anyone accessing the URL should get data back.
3) relationships in the data should point to additional URLs with data.

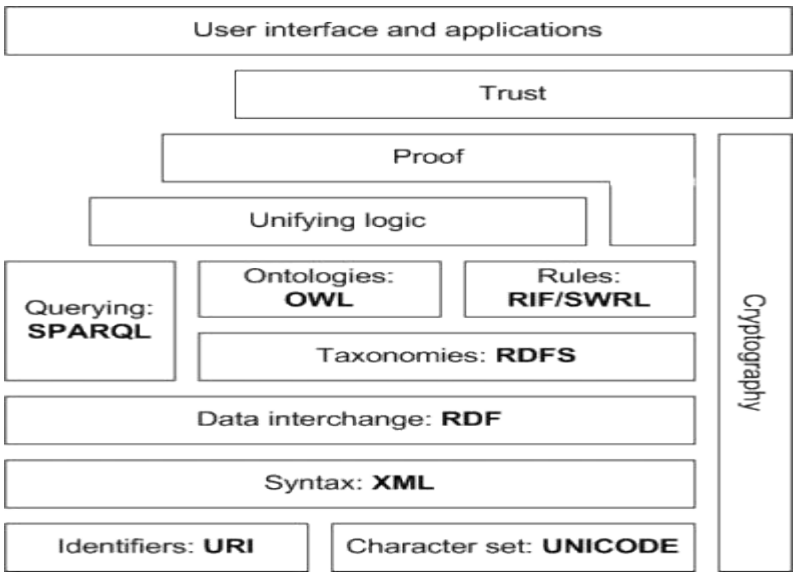
"Semantic Web"은 비록 두 용어간의 정의에 차이가 있지만, 때때로 "Web 3.0"과 동의어로 사용된다.

> Components

"Semantic Web"은 그것이 할 수 있는 포맷과 테크놀로지를 언급할 때 특별하게 자주 사용된다. linked data의 collection, structuring and recovery은 특정한 지식 도메인에서 concepts, terms, and relationships의 공식적 묘사를 제공하는 테크놀로지에 의해 가능하다. 이런 테크놀로지는 W3C standards으로 정해져 있으며, 다음과 같다:

- **Resource Description Framework (RDF)**, a general method for describing information
- **RDF Schema (RDFS)**
- **Simple Knowledge Organization System (SKOS)**
- **SPARQL**, an RDF query language
- **Notation3 (N3),** designed with human-readability in mind
- **N-Triples**, a format for storing and transmitting data
- **Turtle** (간단한 RDF Triple Language)
- **Web Ontology Language (OWL)**, a family of knowledge representation languages
- **Rule Interchange Format (RIF)**, a framework of web rule language dialects

supporting rule interchange on the Web



Semantic-web-stack

The Semantic Web Stack에서는 Semantic Web의 구조를 보여주고 있다. 구성요소의 기능과 연관성은 다음과 같이 요약될 수 있다:

> XML

이것은 해당 다큐멘트에 포함된 콘텐트의 meaning에 대하여 어떠한 semantics도 아직은 결합하지 않은 다큐멘트들의 콘텐트 구조를 위한 기본적 문장규칙을 제공한다. XML은 Turtle과 같은 대안적 규칙들이 존재하게 됨으로써, 오늘날 대부분의 경우에 Semantic Web technologies에서 꼭 필요한 구성요소는 아니다. Turtle은 사실상 표준(de facto standard)이지만, 공식적인 표준화절차를 밟지는 않았다.

> XML Schema

이것은 XML documents에 포함되어 있는 요소들의 구조와 콘텐트를 제공하기 위한 언어이다.

> RDF

이것은 사물("web resources")과 그것들 간의 연관성을 다루는 데이터 모델을 표현하기 위한 간단한 언어이다. RDF-based model은 RDF/XML, N3, Turtle, and RDFa와 같은 다양한 syntax로 표현될 수 있다. RD는 Semantic Web의 기본적인 기준이다.

> RDF Schema

이것은 RDF를 확장한 것이며, properties와 classes로 된 일반적 계층(generalized-hierarchies)의 semantics를 갖고 있는 RDF-based resources의 properties

와 classes를 묘사하기 위한 vocabulary 이다.

> OWL

이것은 properties와 classes를 묘사하기 위하여 더 많은 어휘를 추가한 것이다.: among others, relations between classes (e.g. disjointness), cardinality (e.g. "exactly one"), equality, richer typing of properties, characteristics of properties (e.g. symmetry), and enumerated classes.

> SPARQL

이것은 semantic web data sources용의 protocol and query language 이다.

> RIF

이것은 the W3C Rule Interchange Format 이며, 컴퓨터로 처리할 수 있는 웹 규칙을 표현하는 XML 언어 이다. RIF는 다수의 변종(dialects)가 존재하며, 대표적인 것으로는 RIF Basic Logic Dialect (RIF-BLD)와 RIF Production Rules Dialect (RIF PRD)가 있다.

최신의 표준들:

- ◉ Unicode
- ◉ Uniform Resource Identifier
- ◉ XML
- ◉ RDF
- ◉ RDFS
- ◉ SPARQL
- ◉ Web Ontology Language (OWL)
- ◉ Rule Interchange Format (RIF)

아직 충분하게 현실화되지 못한 분야:

- ◉ Unifying Logic and Proof layers

*** Ontology(information science)**

온톨로지란 실재로 또는 기본적으로 특별한 discourses용으로 존재하는 entities의 유형, 성질, 상호연관성에 대한 공식적인 naming이고, 정의이며, 이것은 철학적 존재론을 taxonomy에 실재적으로 적용시킨 것이다. 또한 온톨로지는 computation이 필요한 변수들을 구획화(compartmentalize)하여 이것들 간의 관계를 설정함으로써, 정보를 조직하는 정형화된 프레임워크이다. 그리고 artificial intelligence, Semantic Web, systems engineering, software engineering, biomedical informatics, library science, enterprise bookmarking, and information architecture에서 자신들의 정보를 조직하고 그것의 복잡성을 제한하기 위하여 온톨로지를 만든 다음에 자신들의 문제해결에 이것을 적용시키고 있다.

> Components

현대의 온톨로지는 많은 구조적 유사성을 공유하고 있다, 그것을 표현하는 언어와 상관없이. 대부분의 온톨로지는 individuals (instances), classes (concepts), attributes, and relations로 묘사된다.

>> 온톨로지의 공동 요소:

- ◉ Individuals: instances or objects (the basic or "ground level" objects)
- ◉ Classes: objects의 sets, collections, concepts, classes in programming, types of objects, 또는 kinds of things
- ◉ Attributes: objects (and classes)가 가질 수 있는 aspects, properties, features, characteristics, or parameters.
- ◉ Relations: classes and individuals가 서로 연결되는 방식.
- ◉ Function terms: complex structures formed from certain relations that can be used in place of an individual term in a statement
- ◉ Restrictions: 어떤 주장을 input으로 접수하기 위하여 사실이어야 한다고 공식적으로 언급된 묘사.
- ◉ Rules: 특별한 형태의 주장에서 나올 수 있는 logical inferences을 묘사하는 if-then (antecedent-consequent) sentence 의 형태로 된 statement.
- ◉ Axioms: 온톨로지에 대한 전반적 이론으로 구성된 논리형태의 assertions (including rules).
- ◉ Events: attributes 또는 relations의 변화.

온톨로지는 일반적으로 온톨로지 언어(예: OWL)을 사용하여 코드화된다.

> Types of ontologies

◉ Domain ontology

도메인 온톨로지 또는 domain-specific ontology는 해당 분야의 일부분을 대표하는 특수한 도메인을 모델화 한다. 그 같은 도메인에 적용된 용어들의 특별한 의미들은 도메인 온톨로지에 의해 제공된다. 예를 들어, “card” 단어는 많은 서로 다른 의미를 가지고 있다. poker 도메인의 온톨로지에서 그 단어에서 “playing card”라는 의미로 모델화될 수 있다. 반면에 컴퓨터 하드웨어 도메인의 온톨로지에서는 “punched card” 그리고 “video card” 의미로 모델화 될 수 있다.

도메인 온톨로지가 매우 특별하고 종종 절충적인(eclectic) 방법으로 개념을 표현하므로, 이것들은 종종 호환성이 없기도 한다. 그렇지만, 도메인 온톨로지에 의존하는 시스템들이 늘어남으로써, 이러한 도메인 온톨로지를을 더 많은 일반적인 표현으로도 사용할 수 있도록 통합되어야 하며, 이것이 온톨로지 디자이너에게는 하나의 도전이다. 동일한 도메인에서 서로 차별화된 온톨로지들은 언어, 의도, 그리고 인식(문화적 배경, 교육, 이데올로기, 등을 근거로)의 차이에서 비롯한다.

현재에, a common foundation ontology로부터 발전하지 못하고 있는 온톨로지들의 통합은 주로 수작업으로 이루어지며, 시간 소모적이고 비용이 많이 든다. 도메인 온톨로지의 요소들에 대한 의미를 지정하는 기본적인 요소들을 제공하기 위하여 똑같은 foundation ontology를 사용하는 도메인 온톨로지들은 자동적으로 통합될 수 있다. 그러나 온톨로지 통합에 대한 일반화된 테크닉에 대한 연구가 이루어지고는 있지만, 아직까지 이 분야의 연구는 주로 이론적 단계에 머물고 있다.

1) **upper ontology (also known as a top-level ontology or foundation ontology)**
이것은 모든 지식 도메인에 걸쳐서 똑 같은 매우 범용적인 개념을 묘사하는 온톨로지 이다. 이것의 중요한 기능은 이 온톨로지의 "under"에 서열화 되어 있는 접근 가능한 수많은 온톨로지들 사이에 매우 포괄적인 어의적 상호운영성을 지원하는 것이다.

it is usually a hierarchy of entities and associated rules (both theorems and regulations) that attempts to describe those general entities that do not belong to a specific problem domain.

Library classification systems predate these upper ontology systems. Though library classifications organize and categorize knowledge using general concepts that are the same across all knowledge domains, neither system is a replacement for the other.

◉ Upper ontology

upper ontology (or foundation ontology)는 다양한 범위의 도메인 온톨로지 간에 적용할 수 있는 common objects의 모델이다. 이것은 다양하고 적합한 도메인 세트들에서 사용되는 용어들 그리고 associated object descriptions를 포함하고 있는 core glossary를 채택하고 있다.

사용이 가능한 여러 가지의 표준화된 upper ontologies가 존재 한다: 예; BFO, Dublin Core, GFO, OpenCyc/ResearchCyc, SUMO, and DOLCE. 어떤 사람들에게는 upper ontology라고 여겨지는 WordNet는 엄격하게 말해서 온톨로지가 아니라, 이것은 도메인 온톨로지를 배우기 위한 언어적 도구 이다.

1) The **Basic Formal Ontology (BFO)** is a formal ontological framework developed by Barry Smith and his associates that consists in a series of sub-ontologies at different levels of granularity. The ontologies are divided into two varieties: continuant (or snapshot) ontologies, comprehending continuant entities such as three-dimensional enduring objects, and occurrent ontologies, comprehending processes conceived as extended through (or as spanning) time. BFO thus incorporates both three-dimensionalist and four-dimensionalist perspectives on reality within a single framework. Interrelations are defined between the two types of ontologies in a way which gives BFO the facility to deal with both static/spatial and dynamic/temporal features of reality. Each continuant ontology is an inventory of all entities existing at a time. Each occurrent ontology is an inventory (processory) of all the processes unfolding through a given interval of time. Both types of ontology serve as basis for a series of sub-ontologies, each of which can be conceived as a window on a certain portion of reality at a given level of granularity.

2) The **general formal ontology (GFO)** is an upper ontology integrating processes and objects. GFO has been developed by Heinrich Herre, Barbara Heller and collaborators (research group Onto-Med) in Leipzig. Although GFO provides one taxonomic tree, different axiom systems may be chosen for its modules. In this sense, GFO provides a framework for building custom, domain-specific ontologies. GFO exhibits a three-layered meta-ontological architecture consisting of an abstract top level, an abstract core level, and a basic level.

3) **Cyc** is an artificial intelligence project that attempts to assemble a comprehensive ontology and knowledge base of everyday common sense knowledge, with the goal of enabling AI applications to perform human-like reasoning.

#OpenCyc

The latest version of OpenCyc, 4.0, was released in June 2012. OpenCyc 4.0 includes the entire Cyc ontology containing hundreds of thousands of terms, along with millions of assertions relating the terms to each other; however, these are mainly taxonomic assertions, not the complex rules available in Cyc. The knowledge base contains 239,000 concepts and 2,093,000 facts and can be browsed on the OpenCyc website.

ResearchCyc

In addition to the taxonomic information contained in OpenCyc, ResearchCyc includes significantly more semantic knowledge (i.e., additional facts) about the concepts in its knowledge base, and includes a large lexicon, English parsing and generation tools, and Java based interfaces for knowledge editing and querying.

4) The **Suggested Upper Merged Ontology or SUMO** is an upper ontology intended as a foundation ontology for a variety of computer information processing systems.

SUMO originally concerned itself with meta-level concepts (general entities that do not belong to a specific problem domain), and thereby would lead naturally to a categorization scheme for encyclopedias. It has now been considerably expanded to include a mid-level ontology and dozens of domain ontologies.

5) **DOLCE and DnS**

Developed by Nicola Guarino and his associates at the Laboratory for Applied Ontology (LOA), the Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) is the first module of the WonderWeb foundational ontologies library. As implied by its acronym, DOLCE has a clear cognitive bias, in that it aims at capturing the ontological categories underlying natural language and human common sense.

DnS (Descriptions and Situations), developed by Aldo Gangemi (STLab, Rome), is a constructivist ontology that pushes DOLCE's descriptive stance even further. DnS does not put restrictions on the type of entities and relations that one may want to postulate, either as a domain specification, or as an upper ontology, and it allows for context-sensitive 'redescriptions' of the types and relations postulated by other given ontologies (or 'ground' vocabularies). The current OWL encoding of DnS assumes DOLCE as a ground top-level vocabulary. DnS and related modules also exploit 'CPs' (Content ontology design Patterns), which provide a framework to annotate 'focused' fragments of a reference ontology (i.e., the parts of an ontology containing the types and relations that underlie 'expert reasoning' in given fields or communities). The combination of DOLCE and DnS has been used to build a planning ontology known as DDPO (DOLCE+DnS Plan Ontology).

Both DOLCE and DnS are particularly devoted to the treatment of social entities, such as e.g. organizations, collectives, plans, norms, and information objects. It has also been used to study and create domain ontologies for sovereign states, geopolitical boundaries, and the agentivity of social entities. The DOLCE-2.1-Lite-Plus OWL version, including a number of DnS-based modules, has been and is being applied to several ontology projects.

6) **WordNet**, a freely available database originally designed as a semantic network based on psycholinguistic principles, was expanded by addition of definitions and is now also viewed as a dictionary. It qualifies as an upper ontology by including the most general concepts as well as more specialized concepts, related to each other not only by the subsumption relations, but by other semantic relations as well, such as part-of and cause. However, unlike Cyc, it has not been formally

axiomatized so as to make the logical relations between the concepts precise. It has been widely used in Natural language processing research.

7) **Unified Foundation Ontology (UFO)**

The Unified Foundational Ontology (UFO), developed by Giancarlo Guizzardi and associates, incorporating developments from GFO, DOLCE and the Ontology of Universals underlying OntoClean in a single coherent foundational ontology. The core categories of UFO (UFO-A) have been completely formally characterized in Giancarlo Guizzardis Ph.D. thesis and further extended at the Ontology and Conceptual Modelling Research Group (NEMO) in Brazil with cooperators from Brandenburg University of Technology (Gerd Wagner) and Laboratory for Applied Ontology (LOA). UFO-A has been employed to analyze structural conceptual modeling constructs such as object types and taxonomic relations, associations and relations between associations, roles, properties, datatypes and weak entities, and parthood relations among objects.

8) **IDEAS**

The upper ontology developed by the IDEAS Group is higher-order, extensional and 4D. It was developed using the BORO Method. The IDEAS ontology is not intended for reasoning and inference purposes; its purpose is to be a precise model of business.

9) **UMBEL**

Upper Mapping and Binding Exchange Layer (UMBEL) is an ontology of 28,000 reference concepts that maps to a simplified subset of the OpenCyc ontology, that is intended to provide a way of linking the precise OpenCyc ontology with less formal ontologies. It also has formal mappings to Wikipedia, DBpedia, PROTON and GeoNames. It has been developed an maintained as open source by Structured Dynamics.

◉ Hybrid ontology

Gellish ontology는 upper and a domain ontology가 결합된 좋은 예이다.

1) **Gellish**

이것은 비록 그것의 개념이 다양한 자연어로 ‘names’와 정의를 갖고 있다하더라도, 자연어에서 독립한 공식적 언어이다. Any natural language variant, such as Gellish Formal English is a controlled natural language. Information and knowledge can be expressed in such a way that it is computer-interpretable, as well as system-independent and natural language independent. Each natural language variant is a structured subset of that natural language and is suitable for information modeling and knowledge representation in that particular language. All expressions, concepts and individual things are represented in Gellish by (numeric) Unique Identifiers (Gellish UID's). This enables a software to automatically generate expressions that are created in one formal natural language into any other formal natural language. From a data modeling perspective, Gellish is a universal and extendable conceptual data model that also includes domain-specific terminology and definitions. Therefore, it can also be called a semantic data model. The accompanying Gellish modeling methodology thus belongs to the family of semantic modeling methodologies.

> Visualization

두 가지 가장 잘 알려진 온톨로지 시각화 평가 테크닉은 indented tree and graph 이며, OWL에서 제공하고 있는 온톨로지용 visual language는 Visual Notation for OWL Ontologies (VOWL)에 상세히 나와 있다.

1) commonsense knowledge

인공지능분야에서, 이것은 보통사람들이 알기 바라는 사실과 정보의 collection 이다. commonsense knowledge problem는 대부분의 사람들이 가지고 있으며, 자연어를 사용하여 인공지능 프로그램을 이용하거나 일반 세상에 대해 추론하도록 만드는 방식으로 표현된 모든 일반적인 지식을 담고 있는 데이터베이스인 commonsense knowledge base를 만들기 위한 knowledge representation (a sub-field of artificial intelligence)에서 계속되고 있는 프로젝트이다. 이러한 데이터베이스는 일반적으로 upper ontologies라 부르는 온톨로지의 한 형태이다.

2) folksonomy

이것은 content를 해제하고 범주화하기 위하여 tags를 공동으로 만들어 번역하는 업무이자 방법에서 유래된 분류시스템이다. 이러한 업무는 또한 collaborative tagging, social classification, social indexing, and social tagging 으로도 알려져 있다.

Folksonomy는 모든 웹페이지가 그것의 콘텐트를 기술하고 있는 machine-readable metadata를 포함하고 있는 Semantic Web을 개발하는데 있어 중요한 키이다. 그러한 metadata는 탐색엔진 검색 리스트에서 정확성을 놀라울 정도로 개선시킬 수 있다. 그렇지만 대규모이고 다양한 웹페이지 저자들의 커뮤니티를 어떻게 설득시키서 자신들의 웹페이지에 일관성 있고 신뢰할 수 있는 방식으로 메타데이터를 추가하도록 설득하는 것은 정말로 어려운 일이다: 그렇게 한다고 하더라도 웹 저자들은 높은 entry costs을 경험하게 되는데, 왜냐하면 메타데이터 시스템은 배우고 사용하는데 시간-소모적이기 때문이다. 이런 이유 때문에, 극소수의 Web authors만이 심지어 비록 Dublin Core meta-tags의 사용이 탐색엔진 검색 리스트에서 자신들의 페이지에 대한 탁월성(prominence)을 높인다하더라도 간단한 Dublin Core metadata standard만을 사용하고 있다. 통제어휘를 사용하는 보다 공식적이고 top-down 방식인 분류표와 대조적으로, folksonomy는 entry costs가 낮은 a distributed classification system 이다.

몇몇 도서관에서 자신들의 목록에 더욱 더 사회적이고 참여적인 web 2.0의 성질을 갖도록 하기 위하여 표준화된 주제표목의 사용과 더불어, 자신들의 online public access catalog, or OPACs에 tagging features를 추가하고 있다. 이러한 것이 이용자에게 도움을 줘서 다른 closed cataloging system을 사용하는데 힘을 보태주고 있지만, 이것은 단지 전통적인 편목을 완전하게 대체하는 것이 아니라 단지 보완만 해주는 것이다.

folksonomy의 조직화나 분류에 대한 연구를 folksontology라고 부른다. 이것은 온톨로지의 한 branch이며, 분류 시스템용으로 무엇이 최상의 features인지를 이들 두 가지에 물어보기 위하여 고도로 조직화된 taxonomies or hierarchies 그리고 loosely structured folksonomy 간의 상호 교차를 다룬다.

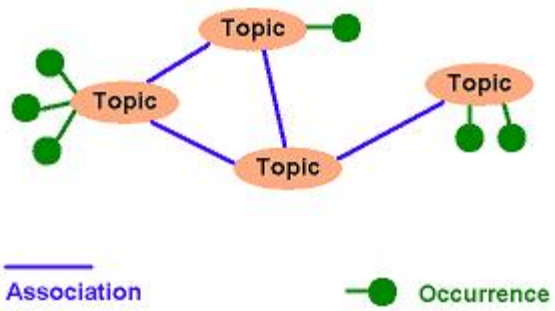
*** 토픽맵(Topic Maps)**

토픽 맵은 지식의 표현과 상호교환을 위한 표준이며, 정보의 발견성(findability)에 초점을 맞추고 있다. 이것은 , back-of-the-book index structures를 모방하는 방식을 사용하여 서로 다른 정보원의 복수 색인들을 통합할 목적으로 1990년대 말에 최초로 개발되었다. 이것은 공식적으로 ISO/IEC 13250:2003 표준이 되었다.

> 토픽 맥에서는 다음과 같은 것을 사용하여 정보를 표현 한다:

◉ **topics:** software modules, individual files, and events과 관련된 사람, 국가, 기관에서 나온 어떤 개념의 표현 이다.

◉ **associations:** topics 간의 hypergraph relationships을 표현 한다.
◉ **occurrences:** 특별한 토픽에 적절한 정보자원을 표현 한다.



Topic Map Key Concepts

토픽 맵은 많은 부분에서 concept maps이나 mind maps과 비슷하지만, 토픽 맵만이 표준이다. 토픽 맵은 시멘틱 웹 기술의 한 형태이므로, 어떤 작업은 semantic web standards의 W3C's RDF/OWL/SPARQL family와 Topic Maps standards의 ISO's family 간에 상호 교류에 의해 진행된다.

1) A **concept map**
이것은 개념들 간에 제안된 연관성을 설명하는 diagram 이다. 이것은 또한 지식을 구조화하여 조직하려고 하는 designers, engineers, technical writers, and others use 의 graphical tool 이다. 또한 이것은 전형적으로 boxes or circles로 아이디어와 정보를 표현하며, 이것들은 a downward-branching hierarchical structure에서 labeled arrows로 연결된다. 개념들 간의 연관성은 causes, requires, or contributes to와 같은 linking phrases 에서 분명하게 표현될 수 있다.

2) A **mind map**
이것은 시각적으로 정보를 outline하기 위하여 사용되는 diagram 이다. A mind map은 종종 중앙에 자리잡고 있는 a single word or text 둘레에서 associated ideas, words and concepts를 추가하기 위하여 만들어진다. 중요한 categories가 중앙 노드에서부터 방사되며 그렇지 못한 categories는 larger branches의 sub-branches 이다. Categories는 a central key word or idea와 관련된 words, ideas, tasks, or other items로 표현될 수 있다.

3) **Topincs**
는 신속하게 발전하는 web databases and web applications이며, LAMP와 semantic technology Topic Maps를 근거로 삼고 있다. A Topincs web database는 Wiki처럼 브라우징을 통하여 정보에 접근할 수 있다. 주제에 관한 페이지를 편집하는 것은 markup 편집보다는 forms를 통해 이루어진다. 이 web database는 특별한 이용자 지단에게 필요한 데이터로 맞춤식 어프로치를 제공할 수도 있다.

많은 방식에서 토픽 맵의 어의적 표현성은 RDF의 그것과 비슷하지만, 중요한 차이점들은 첫째, 토픽 맵은 topics, associations and occurrences의 a template를 제공하는데 있어서 보다 높은 차원의 어의적 추론을 제공하지만, RDF는 한 가지 관계로 링크된 두 가지의 arguments에 대한 a template만을 제공하는 것이고, 둘째, 토픽 맵은 어떠한 수의 노드 간에도 n-ary 관계(hypergraphs)를 허용하지만, RDF는 triplets로 제한한다는 것이다.

> Ontology and merging

Topics, associations, and occurrences는 모두가 type를 가질 수 있는데, 이 때 type는 하나 이상의 토픽 맵의 제작자에 의해 정의되어야 한다. 허용된 type의 정의를 그 토픽 맵의 온톨로지라고 부른다. 토픽 맵은 분명하게 말해서, 복수의 토픽이나 토픽 맵들 간의 identities를 통합하는 개념을 지원하며, 더구나 온톨로지가 토픽 맵 그 자체이기 때문에, 그것들은 또한 다양한 sources에서 나온 정보들을 그것과 연관된 새로운 토픽 맵에서 자동적으로 통합할 수 있다. subject identifiers (URIs given to topics) 와 PSIs (Published Subject Indicators)와 같은 features는 서로 다른 분류방법들 간에 이루어지는 통합을 조정하기 위하여 사용되며, Scoping on names(이름에 대한 범위지정) 과정에서 서로 다른 정보원에 의해 만들어진 특별한 토픽의 다양한 이름을 조직화하는 방법을 제공한다.

> Current standard

최신의 작업 표준 Topic Maps (ISO/IEC 13250)은 the ISO/IEC JTC1/SC34/WG3 committee (ISO/IEC Joint Technical Committee 1, Subcommittee 34, Working Group 3 - Document description and processing languages - Information Association)에서 관리되고 있다.

The Topic Maps (ISO/IEC 13250) reference model 그리고 data model standards은 어떤 특별한 serialization or syntax와는 독립된 방식으로 정의되고 있다.

◉ TMDM: Topic Maps - Data Model

ISO/IEC 13250-2 (TMDM, Topic Maps Data Model)에서는 ISO/IEC 13250-3 Topic Maps XML Syntax and ISO/IEC 13250-4 Topic Maps Canonicalization에서 정의한 것과 같은 syntaxes and notations의 기초를 제공한다. 당연히, TMDM는 특별한 주제(topics, associations, occurrences)를 어떻게 식별하는가, 어떠한 성질이 요구되는가, 두 개 이상의 proxies(대리)가 동일한 주제를 표현하는지를 결정하기 위하여 사용되는 테스트, 등등과 관련된 ontological commitments을 결정한다.

◉ TMRM: Topic Maps - Reference Model

TMDM은 보다 추상적이고 온톨로지적 책임은 거의 없는 TMRM (Topic Maps Reference Model)을 정의한다. 이것의 목적은 TMDM과 같은 subject-centric data models을 위한 a minimal, conceptual foundation으로 봉사하는 것이며, 이러한 모델을 밝히기 위한 온톨로지적으로 중립적 용어를 제공하는 것이다. 또한 이것은 the Topic Maps standards의 전반적인 목적을 충족시키기 위하여, 서로 다른 subject-centric data models을 다 함께 mapping하는데 필요한 것들을 정의하고 있다. 이 때에 각 주제는 그것에 관한 모든 정보와 관련된 a single location을 갖는다.

TMRM은 또한 ISO/IEC 18048 Topic Maps Query Language (TMQL) and ISO/IEC 19756 Topic Maps Constraint Language(TMCL)처럼 관련된 Topic Maps standards를 위한 공식적 기초를 제공하고 있다.

*** 토픽맵의 정의: <국립중앙도서관 도서관연구소 웹진 15호 도서관용어해설>**

토픽맵(Topic Maps)은 차세대 웹인 시맨틱 웹 구현을 위한 등장한 개념체계 인 온톨로지를 표현하기 위한 전용 언어 중 하나이다. 온톨로지를 구축하려 면 개념화 구조를 정확하게 표현해야 하는데, XML은 개념의 특성이나 상호관계를 표현하는 데는 미흡하므로 이를 대신하여 RDF/RDFS, DAML, OWL, 토픽맵 등 온톨로지 전용 언어가 개발되었다.

그 중 토픽맵은 국제표준화기구(ISO)에서 제정된 온톨로지 생성 언어로 W3C의 OWL과 상호 보완 및 경쟁 관계에 있으며 원래 용어집, 시소러스, 색인집등 용어의 의미적 구조를 다루는 목적으로 되었다. 그러나 현재는 정보자원을 의미적 관계를 표현하고 의미적 검색이 가능하도록 하는 시맨틱 웹의 핵심기술로 인정받고 있다.

토픽맵 관련 표준으로 ISO/IEC 13250 국제 표준이 있다. 처음에는 토픽맵 표준 규격으로 SGML(Standard Generalized Markup Language)구조와 HyTM 언어였으나 2001년 TopicMaps.org에서 개발한 XTM(XML TopicMaps)으로 통합되면서 현재는 XTM 1.0이 표준 규격이 되었다.

>SUMMARY: Conceptual<

***Data versus Information**

Data collections:

►well structured collections of related items

►items are usually atomic with a well-defined interpretation

Information repositories:

►Information, on the other hand is usually semi-structured or unstructured.

►Potential for many interpretations. Potential vagueness.

***Data retrieval versus Information retrieval**

►Data retrieval involves the selection of a fixed set of data based on a well-defined query (e.g SQL, OQL, and many many more).

►Information retrieval (IR) involves the retrieval of documents of natural language. Not as structured and may be semantically ambiguous.

***Retrieval and Filtering**

The main differences are:

►the nature of the information need

►the nature of the document set

Despite these two differences, the same models tend to be used. Documents and queries are represented using the same set of techniques and similar comparison algorithms are used.

***User Role**

In traditional IR, the user role was pretty well-defined in that a user:

►formulated a query

►viewed the results

►possibly offered feedback

►possibly reformulated query and repeated steps

In more recent systems, with the increasing popularity of more complex interaction spaces the user usually intersperses *browsing* with the traditional *querying*.

***Information Retrieval System Architecture**

The main components include:

►A document collection/set and queries
►Pre-processing approaches
►Representation of both documents and queries
(dependent on information retrieval model chosen)
►Comparison algorithm
►Feedback module

***Rough Taxonomy of models**

►Boolean
- **Classical Boolean**
- Fuzzy Set approach
- Extended Boolean

►Vector
- **Classical Vector Space Model**
- Latent Semantic Indexing
- Neural Networks

►Probabilistic
- Inference Network
- Belief Network

***Information Retrieval Model**

We can view any IR model as comprising:

►D is the set of logical representations of the documents
►Q is the the set of logical representations of the user information needs (queries)
►F is the mathematical framework adopted

► R is a ranking function which defines an ordering among the documents with regard to a query qi .

∗Basics ..

► Typically, we have a set of index terms $t1 \ . \ . \ . \ tn$.
► We can assign a weight wi,j to each term ti occurring in document dj .
► We can view a document or query as a vector of weights.

∗Boolean Model

► Based on set theory and Boolean algebra.
► model also assumes terms are present or absent, hence term weights wi,j are discrete, i.e., wi,j 2 {0, 1}.
► A query is viewed a Boolean expression.
► For example, $q = t1 \wedge (t2 \vee \neg t3)$. This can be mapped to what is termed disjunctive normal form, where we have a series of disjunctions, e.g., q = 100 _ 110 _ 111

► Advantages

■ clean formalism
■ popular, widespread
■ relatively simple

► Disadvantages

■ not very good performance. Suffers badly from natural language effects of synonymy etc.
■ no ranking of results.
■ harbours some difficulty in use.
■ terms in a documents are considered independent of each other

∗Vector Space Model

Attempts to improve upon the Boolean model by removing the limitation of binary weights for index terms.
Terms can have a non-binary value both in queries and documents.
Hence we can represent documents and query as
n-dimensional vectors.

$$\vec{dj} = (w1,j \ , w2,j \ , \ . \ . \ . \ wn,j \ )$$

$$\vec{q} = (w1,q, w2,q \ldots wn,q)$$

We can calculate the similarity between a document and a query by calculating the similarity between the vector representations.

We can measure this similarity by measuring the cosine of the angle between the two vectors.

Inner product:

$$a \cdot b = |a||b| \cos(a, b)$$

$$\Rightarrow \cos(a, b) = \frac{a \cdot b}{|a||b|}$$

We can therefore calculate similarity between document and query as:

$$sim(\vec{d_j}, \vec{q}) = \frac{d_j \cdot q}{|d_j||q|}$$

$$\Rightarrow \text{sim}(\vec{d_j}, \vec{q}) = \frac{\sum_{i=1}^{n} Wi,j \, X \, Wi,q}{\sqrt{\sum_{i=1}^{n} w_{i,j}^2} X \sqrt{\sum_{i=1}^{n} w_{i,q}^2}}$$

***Vector Space - *tf-idf* weighting**

►We need means to calculate the term weights in the document and query vector representations.

►A term's frequency within a document quantifies how well a term describes a document. The more frequent a term occurs in a document, the better it is at describing that document and vice-versa.

►This frequency is known as the term frequency or *tf factor*.

***idf**

►Also, if a term occurs frequently across all the documents that term does little to distinguish one document from another.

►This factor is known as the inverse document frequency(idf-frequency).

► The most commonly used weighting schemes are known as *tf-idf* weighting schemes.

∗A tf-idf weighting scheme

For all terms in a document, the can be weight assigned is calculated by:

$$wi,j = fi,j \times \log\frac{N}{ni}$$

where

► fi,j is the normalised frequency of term ti in document dj
► N is the number of documents in the collection
► ni is the number of documents that contain term ti

A similar weighting scheme can be used for queries. The main difference is that the tf and idf are given less credence and all terms have an initial value of 0.5 which is increased or decreased based on tf-idf across the document collection. Weighting schemes due to Salton (1983).

∗Vector Space Model - summary

Advantages:

► improved performance over the Boolean model due to weighting schemes
► partial matching allowed which gives a natural ranking

Disadvantages of the Vector Space Model:

► terms are considered to be mutually independent

∗Metrics - Introduction

► *functional requirements:* -standard testing techniques
► *performance:*

■ response time
■ space requirements
■ measure by empirical analysis, efficiency of algorithms and data structures for compression, indexing ..

► *retrieval performance: — how useful is the system. Not really an issue in data*

retrieval systems where perfect matching is possible (as there exists a correct answer).

Evaluation of IR systems is usually based on a test reference collection and some human evaluations.
Test collection usually comprises:

►a collection of documents
►a set of information requests (queries)
►a list of relevant documents for each request

Given some IR system, the evaluation quantifies similarity between the set of documents retrieved by the IR system and those retrieved by the expert.

Interaction with the system may be:

►one-off batch query
►interactive session

For the former, "quality" of the returned set is the important metric.
For interactive systems, other issues have to be considered—duration of session, user-effort required etc. These issues make evaluation of interactive sessions more difficult.

***Precision and Recall**

►The most commonly used metrics are: *precision* and *recall*
►Given a set D and a query Q:
►Let R be the set of documents relevant to Q
►Let A be the set actually returned by the system
►Let RA be the intersection of R and A
►Precision is defined as $\frac{|RA|}{|A|}$
►Recall is defined as $\frac{|RA|}{|R|}$

Returned documents are usually ranked.
Typically plot precision against recall.
In an ideal system, for a recall value of 1, we would have a precision value of 1. i.e., all relevant documents have been returned and no irrelevant documents have been returned.

Usually plot for recall values = 0%, 10% . . . 100%.
Typically calculate precision for these recall values over a set of queries:

$$P(r) = \sum_{i=1}^{n} \frac{\Pi(r)}{N}$$

►Single value measures are often used (Evaluate precision when first relevant document retrieved, R-precision:
Calculate precision when the final relevant document has been retrieved etc.)
►Precision Histograms
►the harmonic mean, which combines precision and recall into a single value

$$F(j) = \frac{2}{\frac{1}{r(j)} + \frac{1}{p(j)}}$$

►a variation on the harmonic mean is the *E measure* which allows the user to specify the importance of precision and recall

$$E(j) = 1 - \frac{1 + b^2}{\frac{b^2}{r(j)} + \frac{1}{p(j)}}$$

*** Precision-Recall ??**
Precision-Recall values are useful
- ►widespread use
- ►give definable measure
- ►summarise behaviour of IR system

Disadvantages
- ►Not always possible to calculate recall
- ►measure effective of queries in batch mode only
- ►Precision and recall can only be calculated when we have ranking
- ►Not necessarily of interest to user.

***User-Oriented Measures**
►Other approaches try to take into account the user's knowledge of the document collection.
►Let $U \subset R$ which is known to user. Let AU be set of returned documents known to the user. Let the relevant ones in this set be Rel_AU

► Coverage = $\frac{|Rel\_AU|}{|AU|}$

► Let NEW be the set of relevant documents returned to user previously unknown to the user.

► Novelty = $\frac{New}{New+|AU|}$

► These metrics are often used in interactive sessions to gauge the usefulness of successive iterations on user interaction.

*Other user measures include:
- ► recall effort
- ► Expected Search Length
- ► Satisfaction and Frustration

***Test Collections**

Many test collections exist for evaluating IR systems. TREC, CACM, CISI and Medline are among the most commonly used.
TREC provided and provides means to empirically test the performance of systems in different domains.

- ► interactive
- ► expert search
- ► natural language processing techniques
- ► cross language
- ► high precision
- ► spoken language retrieval
- ► very large corpus experiments

***Text Properties**

► Not all words are equally important for capturing meaning of a document
► Text documents comprise symbols from a finite alphabet
► What is the distribution of the frequency of different words?
► How fast does vocabulary size grow with the size of a document collection?
► Such factors affect the performance of information retrieval

►Can be used to select appropriate term weights and other aspects of an IR system.

***Word frequencies**

►A few words are very common. e.g. the two most frequent words (e.g. “the", “of") can account for about 10% of word occurrences.

►Most words are very rare. Half the words in a corpus appear only once i.e. a “heavy tailed"/Zipfian distribution

Frequent Word	Number of Occurrences	Percentage of Total
the	7,398,934	5.9
of	3,893,790	3.1
to	3,364,653	2.7
and	3,320,687	2.6
in	2,311,785	1.8
is	1,559,147	1.2
for	1,313,561	1.0
The	1,144,860	0.9
that	1,066,503	0.8
said	1,027,713	0.8

Frequencies from 336,310 documents in the 1GB TREC Volume 3 Corpus
125,720,891 total word occurrences; 508,209 unique words

***Zipf**

►Gives an approximate model for the distribution of different words in a document

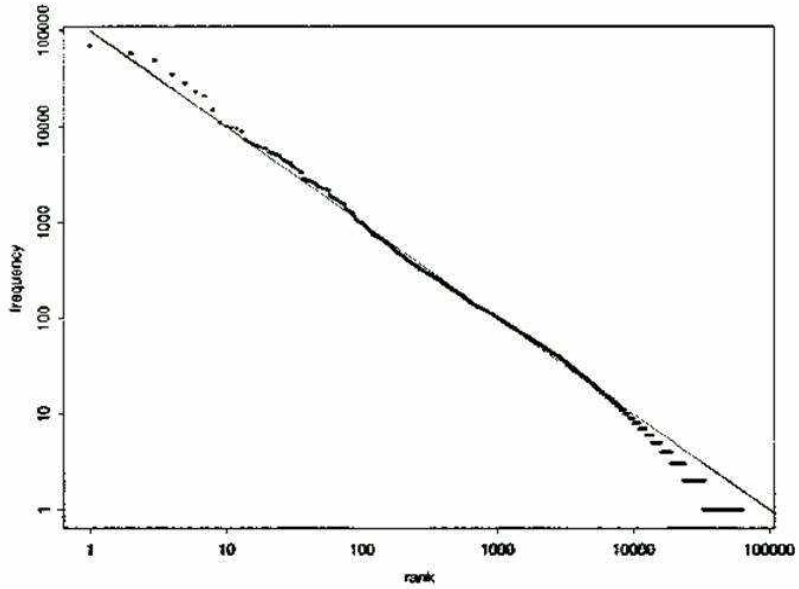
►Rank(r): The numerical position of a word in a list sorted by decreasing frequency (f).

►Zipf’s law: $f \times r$ = constant

►Represents a power law; straight line on a log-log plot

►Accurate model except for extremes

►Model on the Brown corpus



Luhn - resolving power

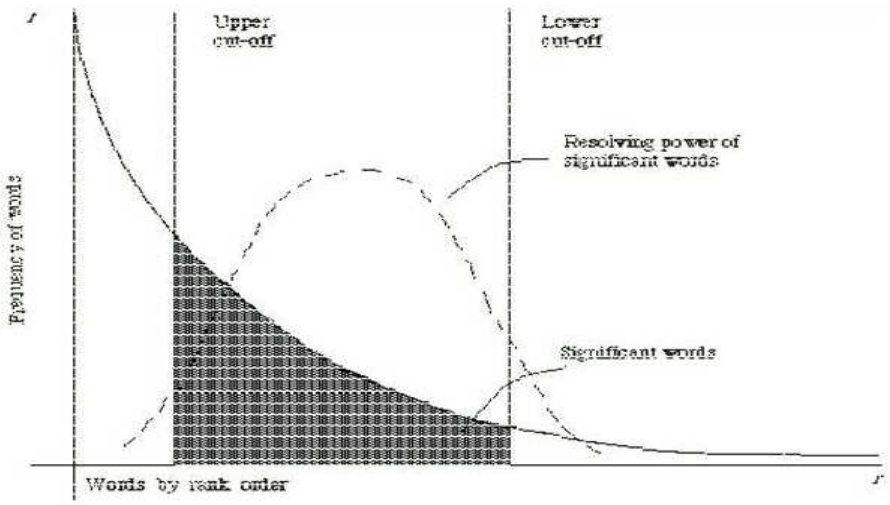


Figure 2.1. A plot of the hyperbolic curve relating f, the frequency of occurrence and r, the rank order (Adapted from Schultz[58] page 120)

***Weighting schemes**

Quality of performance of information retrieval system depends on the quality of the weighting scheme

Want to assign high weights to those terms with a high resolving power.

tf*idf is one such approach where weight is increased for frequently occurring terms but decreased again for those that are frequent across the collection

***BM25/Okapi**

$$BM25(Q,D) = \sum_{t \in Q \cap D} \left(\frac{tf_t^D \cdot \log\left(\frac{N - df_t + 0.5}{df_t + 0.5}\right) \cdot tf_t^Q}{tf_t^D + k_1 \cdot ((1-b) + b \cdot \frac{dl}{dl_{avg}})} \right) \quad (1)$$

- ►Standard benchmark
- ►Relatively good performance
- ►Needs to be tuned for collection

*** Axiomatic approaches**

►*Constraint 1*

Adding a query term to a document must always increase the score of a document

►*Constraint 2*

Adding a non-query term to a document must always decrease the score of a document

►*Constraint 3*

Adding successive occurrences of a term to a document must increase the score of a document less with successive occurrences. Essentially any term-frequency factor should be sub-linear.

►*Constraint 4*

- ■ Thus, vector length should be a better normalisation factor for retrieval. However, using the vector length will violate one of the existing constraints.
- ■ Thus, ensuring that the document length factor is used in a sub-linear function will ensure that repeated appearances of non-query terms are weighted less.

► New weighting schemes which adhere to all these constraints outperform best known benchmarks

***Extensions**

There exist many extensions to models. This can be broadly categorised as:

- ► Extensions to the underlying mathematical model
- ► Ad-hoc extensions to model to incorporate extra sources of evidence.

***Extended Boolean Model**

- ► Attempts to improve upon Boolean model by adopting partial weighting on terms.
- ► Allows Boolean queries within a vector space model.
- ► Euclidean distances from document points to other points used to calculate similarity. For example, for a query *t*1 _ *t*2, the distance from the origin can be used to measure similarity.

***Latent Semantic Indexing**

- ► Attempts to overcome term independance assumption of the classical vector space model.
- ► Approach involves mapping the term document matrix to a reduced space using singular value decomposition.
- ► The resulting matrix is then used for in comparisons to queries.
- ► Better performance obtained but difficulty exists in determining the optimal reduction in matrix size.

***Relevance Feedback**

- ► Attempt to improve performance by modifying the user query; new modified query is then resubmitted to the system.
- ► New query is usually created via:
  - ■ incorporating new terms
  - ■ re-weighting existing terms

***Possible approaches**

- ► feedback from user to recalculate weights

► analysis of document set:
  ■ local analysis (returned set)
  ■ global analysis (whole document set)

***User feedback**

► User examines returned list of documents and marks those which are relevant.
► This feedback allows reformulation of query.
► Advantage: User is shielded from task of query reformulation and from the inner details of the comparison algorithm.
► Potential problems in *query drift*.

***Relevance Feedback for the VS Model**

► Assumes relevant documents have similarly weighted term vectors.

Let *Dr* be the set of relevant documents returned
Let *Dn* be the set of non-relevant documents returned
Let *Cr* be relevant documents in whole collection

► Assume *Cr* is known for a query *q*
► The best vector for a query to distinguish relevant from non-relevant is:

$$\vec{q} = \frac{1}{|C_r|} \sum_{d_j \in C_r} d_j - \frac{1}{N - |C_r|} \sum_{d_j \notin C_r} d_j$$

► impossible to generate this query as we do not know *Cr*

► Can make estimate though as we know *Dr* (*CR*)
  ■ One well known approach – Rocchio:

$$\vec{q} = \alpha \vec{q} + \frac{\beta}{|D_r|} \sum_{d_j \in D_r} d_j - \frac{\gamma}{|D_r|} \sum_{d_j \in D_n} d_j$$

α, β, and γ are constants which determine
  ▫ importance of feedback
  ▫ relative importance of positive feedback over negative feedback

***Local Analysis**

► Many scenarios exist where people give short queries and little feedback.

- ►Documents retrieved are examined at query time to determine terms for query expansion.
- ►Typically develop some form of term-term correlation matrix to quantify correlation between two terms.
- ►Expand query to include terms correlated to the query terms.
- ►Main approaches include - associated clusters, scalar clusters and metric clusters.

*** Association cluster**

Create matrix M of terms × terms

$$M_{i_j} = \frac{freq_{i,j}}{freq_i + freq_j - freq_{i,j}}$$

Each cell *Mi,j* indicates correlation between i and j.

Can develop an association cluster for each term *ti* :

choose *ith* row

select top N values from row

For query q, select a cluster for each query term ⇒ |*q*| clusters.

N is usually small to prevent generation of very large query.

***Metric Clusters**

Association clusters do not take into account position within documents.

Metric clusters attempt to overcome this.

Let *dis*(*ti* , *tj*) be the distance between two terms *ti* and *tj* in the same document.

if *ti* and *tj* are not in same document, then *dis*(*di* , *dj*) = ∝

Can define term-term correlation matrix by:

$$M_{i,j} = \sum_{t_i, t_j \in D_i} \frac{1}{dis(t_i, t_j)}$$

Can define clusters as before.

***Scalar Clusters**

Based on comparing sets of words.

If two terms have similar neighbourhoods there is a high correlation between terms.

Similarity can be defined by comparing the two vectors representing the neighbourhoods.

This measure can be used to define the term-term correlation
matrix and procedure continues as before.

***Pre-processing involves**

- ►Stop-word removal
- ►Stemming
- ►Indexing

***Stop word removal**

- ►Stop words are words with poor resolving power.
- ►Can maintain list of stop words - articles, prepositions etc.
- ►Can count frequency of terms across the corpus and ignore those with too high a frequency
- ►Reduces size of index
- ►In reality, any good weighting scheme should weight these terms with value close to zero in case so no influence on weighting schemes should be detected.

***Stemming**

- ►Goal is to reduce all morphological variations to a common root form.
- ►Improves performance of the IR system by overcoming problems arising from user variations in term usage.
- ►Increases recall; can damage precision.
- ►Secondary beneficial effect in terms of reducing lexicon and hence side of resulting index

***N-Gram Stemmers**

- ►Based on counting the number of diagrams (or N-grams)
    - ■computer: co om mp pu ut te er
    - ■computing: co om mp pu ut ti in ng

- ►Can calculate likelihood of the two being variations of the same word as ratio of common N-grams to the number of unique N-grams

***Suffix Removers**

- ►Based on a set of rules to remove endings. For example, if word ends in 's', but not 'us' and not 'es', then remove s.

  Well known approaches:
  - ■ Lovin
  - ■ Porter
  - ■ Krovetz

***Indexes**

- ►Having removed stop words and stemmed remaining terms, we have a list of terms (stems) and occurrence.
- ►Need to be able to calculate similarity quickly.
- ►Need an index on words with information on occurrences
- ►Type of index required depends on type of queries supported
- ►Fast lookup on terms - trie, hash, B+-tree
- ►Need to store occurrences - inverted list approach usually adopted
- ►Much work on compression of indexes
- ►Many parallel variations of indexes researched

***Quick recap**

- ►Main mathematical models
- ►Main ideas behind weighting schemes
- ►Feedback mechanisms
- ►Pre processing
- ►Metrics

***Many subfields in IR**

- ►Collaborative Filtering
- ►NLP in IR
- ►Distributed IR
- ►Visualisation
- ►Web Search
- ►Learning

***Collaborative Filtering**

- ►collect human judgments and match people who share same information needs and tastes users share their judgments and opinions
- ►echoes "word of mouth" principle
- ►offers support for filtering/retrieval of items where content cannot be easily analysed in an automated manner
- ►ability to filter based on quality/taste

***General approach**

Three general steps:

- ► Find how similar each user is to every other user (Calculate user correlation)
- ► Form groups or neighbourhoods of users who are similar (Select Neighbourhood)
- ► In each group, make recommendations based on what other users in the group have rated (Generate Prediction)

►Metrics of coverage and accuracy often used to measure performance
►Issues arise with new items and new users
►Issues also exist with temporal aspect
►Robustness to noise current area of research Many systems combining collaborative with content
►approaches (mainly adhoc)

***Natural Language Processing in IR**

- ►Most of the techniques discussed here have been statistics-based, e.g. in deriving weighting schemes.
- ►Large body of work in applying NLP approaches
  - ■ obviously in areas like stemming
  - ■ also in parts of speech recognition to inform weighting schemes
  - ■ in query analysis and disambiguation
  - ■ Techniques often more computationally expensive and hence often restricted to applications on query side.

***Distributed IR**

- ►Often repository does not reside at single location
- ►Different scenarios exist for distributed case

- Collection distributed for efficiency purposes (easy case)
- Set of sites with repositories that we can access freely
- Set of sites with repositories only accessible via interface provided by owner of collection. We may be dealing with different IR models, weighting schemes, types of answer sets

► Final case poses some difficult questions

***Main areas**

► Site description - how do we best describe a site?

► Source selection - how do we know which site to select for a particular query (e.g query clustering, query probing)

► Results fusion - how do we merge answer sets?

***Visualisation**

► There have been many approaches to interface design and information visualisation.

► Difficult task as there are many aspect of the information retrieval process that could be represented:

- Representing queries
- Representing relationship between query and returned documents
- Representing relationship between returned documents and the rest of the collection
- Overall views on the document collection
- Representing user's interaction

***Many prototype systems**

► There have been many proposed approaches/systems

- Representing queries (e.g VQuery)
- Representing relationship between answer set and query (e.g. Tilebars)
- Representing overall collection (e.g. self organising maps)
- Representing answer set and relationship to query (e.g. kartoo (web setting))

► Still remain many issues, particulary in the area of evaluation.

***Web search**

► Most notable application of information retrieval ideas

► Given massive quantities of data necessitated much emphasis on indexing and

efficiency

► New interesting problems (vague queries, graph structure over collection, volatile data, diversity in language, adversarial search)

***Evolution**

► 1st generation: use only on-page information – word frequency etc. (classical IR)

► 2nd generation: link analysis (HITS, Pagerank), click through data (in conjunction with classical IR)

► 3rd generation: integrate multiple sources of information, context analysis (spatial, query stream, personal profiling), aiding the user (re-spelling, query refinement, query suggestion), representation of results/query/collection

► Many interesting questions and areas under research - collaborative search, query log mining, complex information-rich spaces.

***Learning in IR**

► Learning has been used to:
- model users behaviours
- learn means to combine sources of evidence
- improve representations
- classify data

► Earlier examples
- relevance feedback
- SOMs for clustering

► Lots of hard problems in IR with many potential sources of evidence.

► Excellent domain for learning approaches

► Several notable successes; growing domain (Learning to rank); growing domain

*So ..

► Looked at main areas (models, evaluation, weighting schemes etc.)

► Looked briefly at some subfields (all worthy of full tutorials in their own right)

>SUMMARY FIN<