정보검색기의 원리와 구조

황 호 정 *

〈Contents〉

- Information
- Information Retrieval System
- Automatic Indexing
- Query Operation
- Document Operation
- Information Gathering / Distribution
- Advanced / Case Study
- Conclusion

* 한글과 컴퓨터, Simmany Search Engine 개발자

☐ Information (1)

▸ Data vs Information

	Data Retrieval	Information Retrieval
Matching	Exact match	Partial match, best match
Inference	Deduction	Induction
Model	Deterministic	Probabilistic
Classification	Monothetic	Polythetic
Query language	Artificial	Natural
Query specification	Complete	Incomplete
Items wanted	Matching	Relevant
Error response	Sensitive	Insensitive

※ Contents : Text, Image, Sound, Data, Video

☐ Information (2)

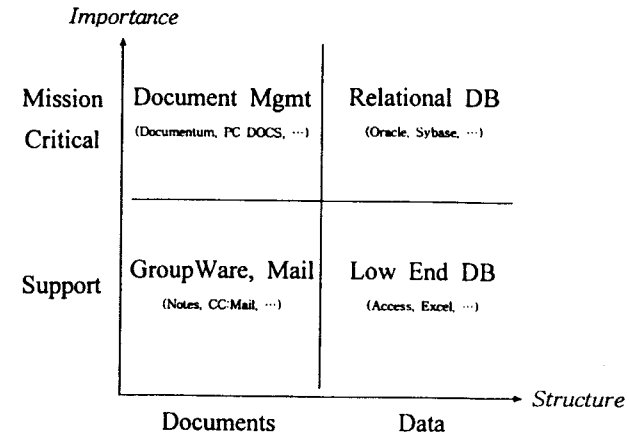
▸ Data vs Document

Issue	Data	Documents
Age of Technology	30 Yrs+	<10 Yrs
Relationships	Simple, Structured	Complex, Loose
Workflow	Transaction	Ad-Hoc
Datatypes	Numeric	Multimedia
Element Size	Small(Bytes)	Large(Megabytes)
Underlying Model	Relational	Object Oriented

☐ Information System (1)

- Information Retrieval
- Question Answering
- Data Base Management
- Management Information

☐ Information System (2)



☐ Information System (3)

▸ IR, DBMS, AI Comparison

	Data Object	Primary Operation	Database Size
IR	document	retrieval (probabilistic)	small to very large
DBMS (relational)	table	retrieval (deterministic)	small to very large
AI	logical statement	inference	usually small

☐ Information System (4)

▸ Texcel / Information Server

http://www.texcel.no/imarch.htm

Site-Specific Customization	Services / Technology
Workflow	Work assignments / Workflow Management
Content Generation Tools	SGML Editing / Collaborative Authoring
Application Manager APIs	Electronic Review / Document Assembly
Repository Manager	Event Notification / SGML-based Schema Shared Content / Metadata / Version Control Query & Retrieval / Access Control / Links
Object-Relational Database Management	

☐ Information System (5)

▸ Document & EDMS Lifecycles

- Create

  Locate/Search → Transform/Conversion → Write/Authoring

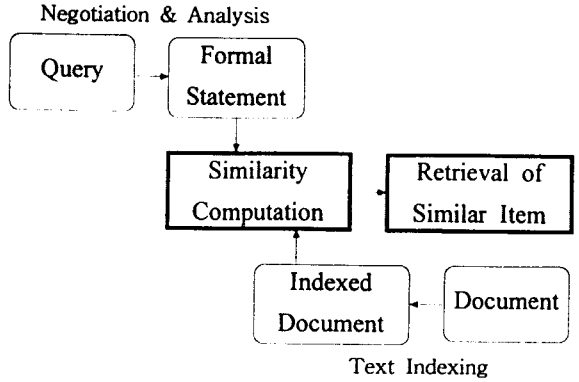
- Manage

  Save/Control → Review/Work Process → Integrate/Assemble

- Distribute & Revise

  Publish/Distribute → Use/Consume → Update/Revise

☐ Information Retrieval



☐ Information Retrieval System

- Document

  Text → Words → Non-Stoplist words

  → Stemmed words ⇒ Database

- Queries

  Query Terms → Stemmed words ⇒ Database

  ⇒ Relevant document set → Retrieved document set

  → Ranked document set

※ Term Weighting / User Interface / Query Operation

☐ Digital Index

- Keyword
- Identifier / Descriptor

- Inverted File
- Signature
- Hashing

※ Dictionary

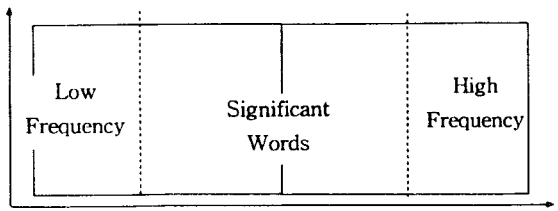
☐ Automatic Indexing (1)

▸ Term Extraction

- The constant rank-frequency law of Zipf

  Frequency · rank ≒ constant

- Resolving power



☐ Automatic Indexing (2)

▸ Term Weighting

- TF(Term Frequency)
- IDF(Inverse Document Frequency)

- Information $= \frac{1}{\log p}$

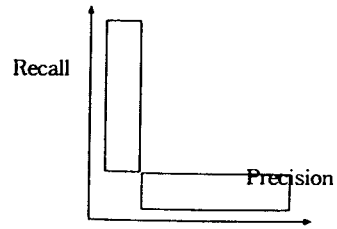
※ TotFreq / DocFreq / DocSize …

☐ Automatic Indexing (3)

- ▸ Term Analysis / Language Dependency
- · Lexical Analyzer
- · Stoplist / Negative Dictionary
- · Stemmer / Morphological Analyzer
- · Thesaurus

☐ Query Operation (1)

- · Recall = $\frac{\textit{number of items retrieved and relevant}}{\textit{total relevant in collection}}$
- · Precision = $\frac{\textit{number of items retrieved and relevant}}{\textit{total retrieved}}$
- · Recall-precision graph



☐ Query Operation (2)

- · Lexical Analyzer
- · Query Expansion
- · Term Weighting
- · Feedback

- · Boolean Operation
- · Natural Language Query

☐ Document Operation

- · Boolean Operation
- · Vector Space Model

- · Automatic Document Classification
- · Document Abstraction

☐ Information Gathering (1)

- ▸ Robot
  - ‣ Bot / Spider / WebBot / WebRobot
  - ‣ Ant / Worm
- ▸ Agent(Intelligent Agent)
  - ‣ Finding & Filtering
  - ‣ Customizing / Adaptive
  - ‣ Automating / Autonomous
  - ‣ Reactive / Continue to Run
  - ‣ Social / Believable

☐ Information Distribution

- ‣ Delivering
- ‣ Filtering
- ‣ Routing
- ‣ Personalizing
- ‣ Pushing
- ‣ Information Abstraction
- ‣ Information Extraction

☐ Information Gathering (2)

- ▸ Web Agent
  - ‣ Use less Resource
    - ⇒ Network / Server
  - ‣ Don't Run, Walk Slowly
  - ‣ Robot Exclusion
  - ‣ Name Resolving

※ Indexing Robot

☐ What Do People Want from IR ? (1)

- ▸ D-Lib Magazine, CIIR, W. Bruce Croft
  http://www.dlib.org/dlib/november95/11croft.html

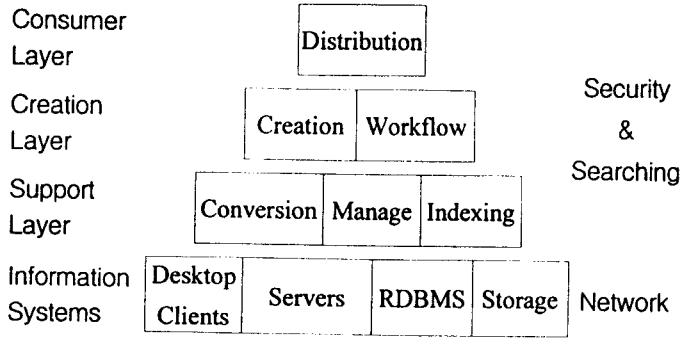
① Integrated Solutions
② Distributed IR
③ Efficient, Flexible Indexing & Retrieval
④ Magic
⑤ Interfaces & Browsing

☐ What Do People Want from IR ? (2)

⑥ Routing & Filtering
⑦ Effective Retrieval
⑧ Multimedia Retrieval
⑨ Information Extraction
⑩ Relevance Feedback

☐ Electronic Document Management System

▸ EDMS Technology Framework



☐ Advanced Study (1)

▸ Client/Server Paradigm

- Information-Centric Organization
- Information-Driven Productivity
- Information-Driven Economy
- Economy of Scope
- Team-Oriented Workflow
- Virtual Enterprise

※ Information Revolution will create a gap between the information illiterate and the information enabled.

☐ Advanced Study (2)

▸ Client/Server Attributes

Attribute	Client	Server
Execution	Fixed Start & End	Runs Forever
Primary Purpose	Maintain User Dialog • Screen/Window Handling • Menu/Command Interpretation • Mouse/Keyboard Entry • Data Entry & Validation • Help Processing • Error Recovery	Provide Functional Service • Application Data Sharing • Communication Sharing • File Sharing • Printer Sharing • CPU Sharing • Display Sharing
Transparency	Hides Network & Servers	Hides Service Implementation Details
Includes	Comm. with Different Servers	Comm. with Different Clients
Excludes	No Client-Client Comm.	No Server-Server Comm.

☐ Advanced Study (3)

▸ MetaData

- XML(eXtensible Markup Language)
  ⇒ MCF(Meta Content Framework)
- GILS(Government Information Locator Service)
  ⇒ GILS core
- SeriCore

※ SMDK Catalog Server

☐ Advanced Study (4)

▸ GILS(Global Information Locator Service)

- Def : A decentralized collection of locators and associated information services used by the public either directly or through intermediaries to find information

- Making it Easier to Search for Information
- Using Well-Known Technology
- Labels for Information Containers
- Locate Information of All Kinds

☐ Case Study (1)

▸ Simmany
http://simmany.chollian.net

- SimBot / Simmany / Simmini(Simmany lite)

- Client/Server Application
- Multi-Threaded Program
- File System Based
- H/W : Pentium-Pro 200MHz, 128M(RAM), 4G(HDD) * 1
  Pentium-Pro 200MHz, 512M(RAM), 31G(HDD) * 2

☐ Case Study (2)

▸ SMDK(SimMany Development Kit)

- Full-Text Indexing
- Support NLP Query
- Support Boolean Expression
- Support Query Expansion
- Document Ranking

- Indexing / Retrieval Library
- Support Multi-Threading
- Multi-Platform(UNIX, Windows-95, Windows-NT)

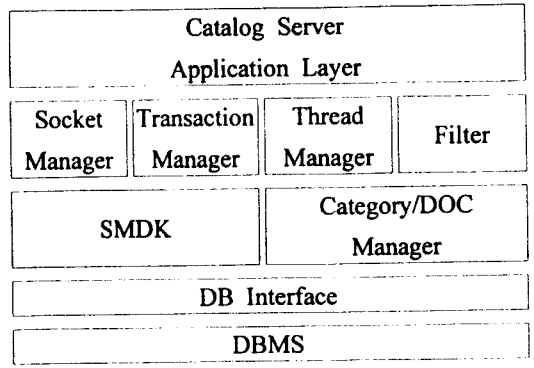
☐ Case Study (3)

▸ SMDK Catalog Server

- User : Easy to Use
- Developer : Easy to Develop
- Incremental Indexing
- Managing Full-Text & Meta Data
- Managing Category
- Support Word-Processor Document(HWP, WORD, …)
- 3-Tier system : Application Server
- Using Communication Protocol
- Using R-DBMS

☐ Case Study (4)

▸ SMDK Catalog Server Architecture



☐ Case Study (5)

▸ Communication Protocol

- Support Transparent Development
- Support Expansive Development
- Support Easy Development

※ Example

SearchMetaData QueryString="컴퓨터와 인터넷";
SearchDomainName="/산업,경제/컴퓨터/정보통신";
SectionId=0; NumberOfStart=0; NumberOfRequest=10;

☐ Conclusion

- Data vs. Information vs. Document vs. Contents
- IR : Creation → Manage → Distribute
  Gathering → Processing → Distributing
- MetaData
- Simmany
- Client/Server Paradigm