Indexing and Retrieving Structured Documents

Yves Chiaramella
IMAG
BP 43
38041 Grenoble Cedex
France
chiara@imag.fr - http://www-clips.imag.fr/

Structured Documents

- Introduction

  » **In standard IR, documents are considered as atomic information units whatever their type or size**

  - Indexed as a whole

  ➨ Indexes do not express the internal organisation of the discourse set by the author(s)

  - Retrieved as a whole

  ➨ Users cannot retrieve independant components of documents that might be more adapted (more focused) to their information needs

Structured Documents

- Introduction
  - **Fast development of tools about structured documents**
    - » **Database systems,**
    - » **New standards (SGML, XML, HTML, ODA…)**
    - » **But also MPEG for video data**

➨ **Questions :**

What is the impact of structure on information retrieval?

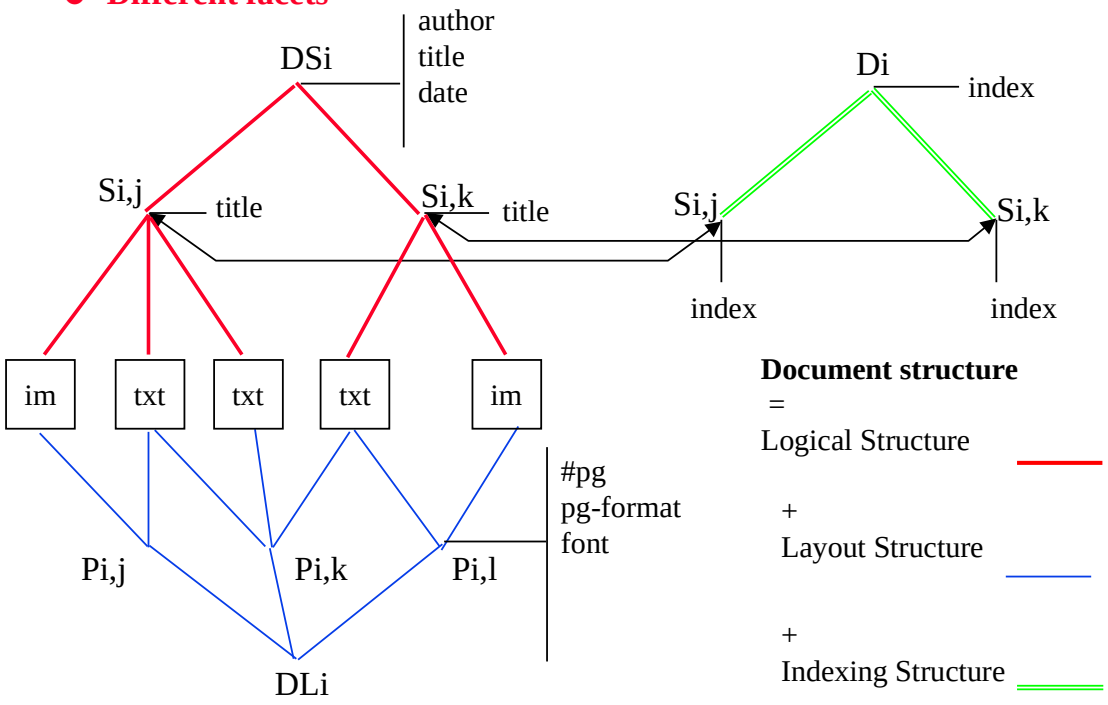
If any, then what kinds of approaches to improve retrieval performances?

Plan

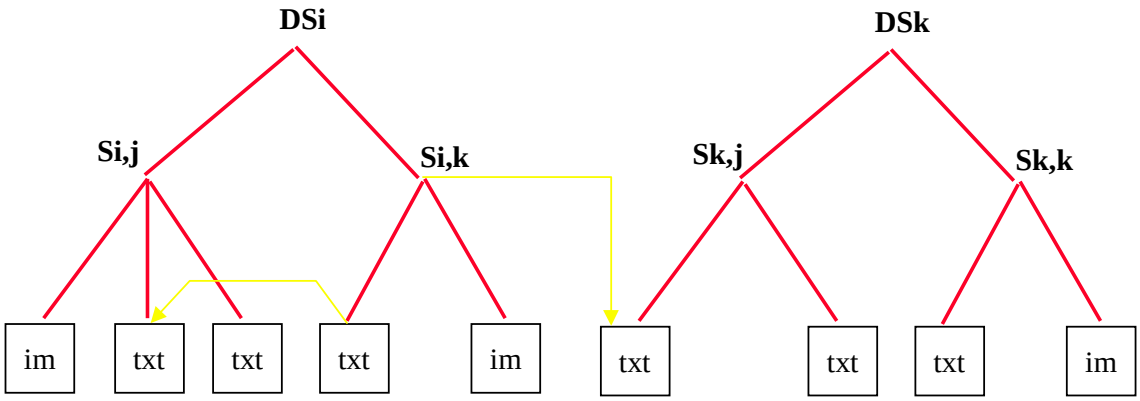
- **Notion of Structured Document**
- **The Impact of Structure**
- **Approaches**
  - Hypermedia
  - Passage Retrieval
  - Indexing / Retrieving Hierarchical Structures
- **Conclusions**

Structured Documents

- **Different facets**



Structured Documents



Document = logical structure + Layout structure + Indexing Structure

+

Browsing Links (internal, external) => HYPERMEDIA DOCUMENTS

The Impact of Structure

- **1. Information needs often involve structure**

On top of content requirements, information needs often include requirements about :

» **Attributes**

eg. Novels written by Hemingway

involved structural information : attributes type of doc, author

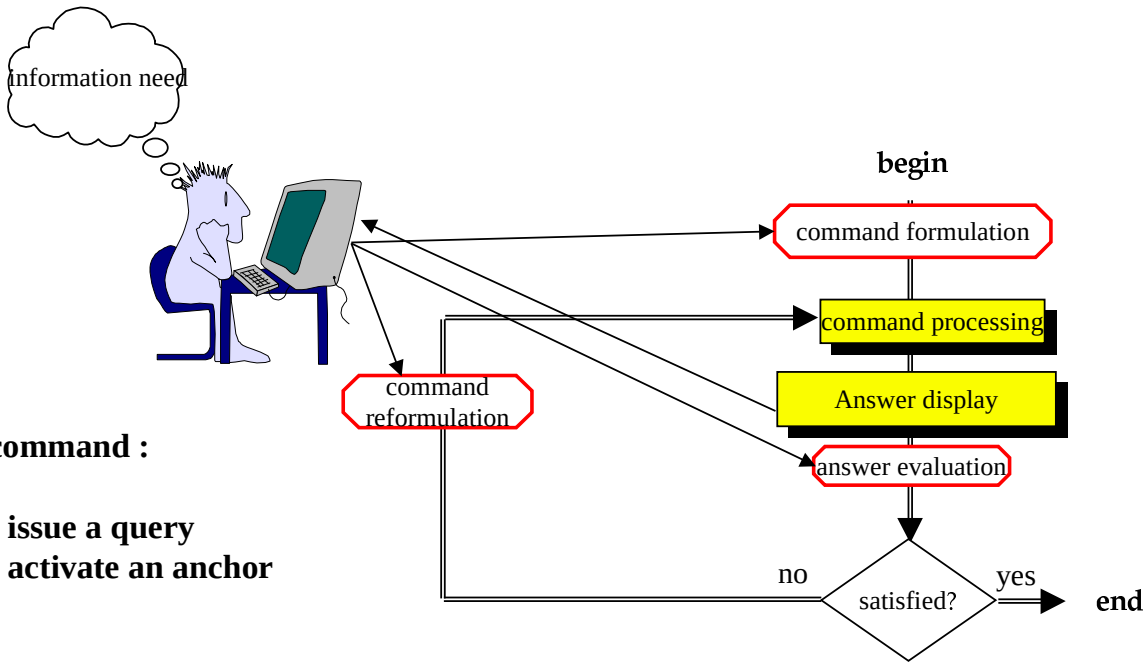
» **links**

eg. A biography of Hemingway illustrated by pictures of the novelist.

involved structural information : component link between text and image

The Impact of Structure

- **2. Interaction & User tasks**

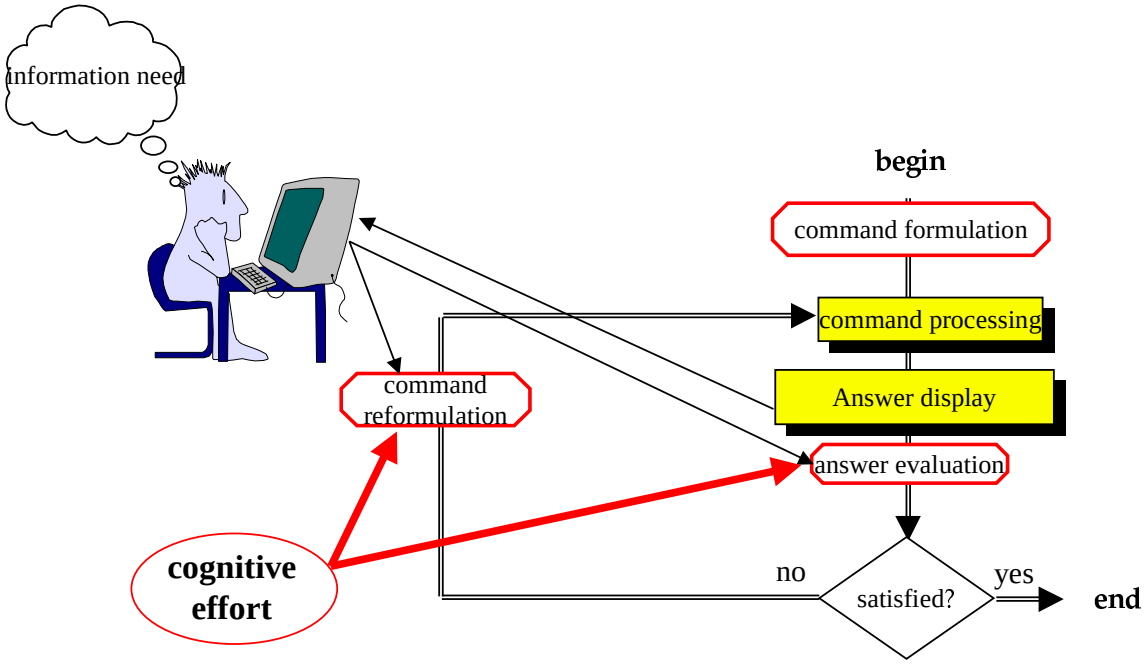


command :

- issue a query
- activate an anchor

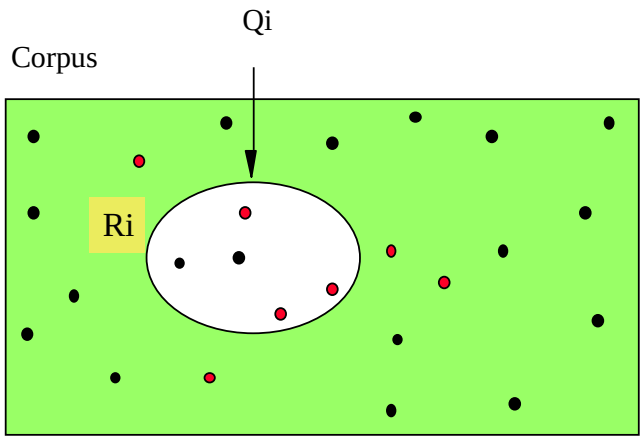
The Impact of Structure

- **Interaction & Cognitive effort**



The Impact of Structure

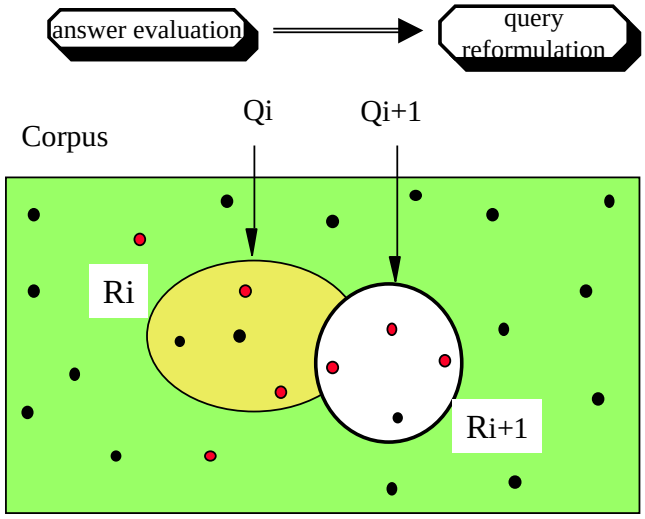
- **Querying (1)**



System's response is a closed window on the document space

The Impact of Structure

- **Querying (2) : moving around the window**



Cognitive effort :

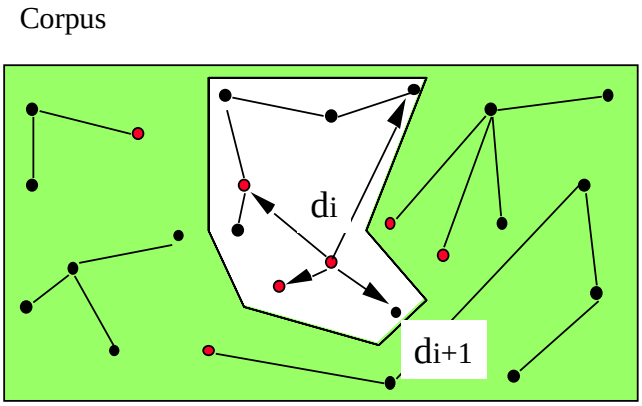
- evaluating responses (Ri)
- properly reformulating query (Q i+1) from Ri

Disorientation :

- length of responses
- bad ranking

The Impact of Structure

- **Browsing**



Cognitive effort:

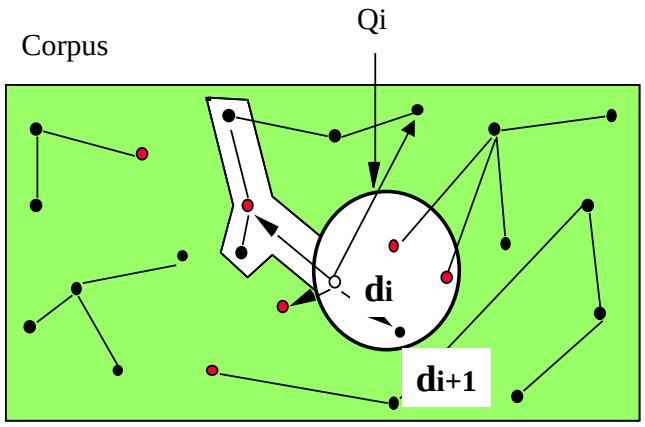
- stacking paths and relevance judgements from previous steps

Disorientation:

- length of paths
- loops
- redundancy
- misleading paths

The Impact of Structure

- **Combining Querying and Browsing**



Cognitive effort

- browsing = incremental , try and error, easier process
- may help query reformulation

Disorientation

-querying effective for topic relocation in the document space

The Impact of Structure

- **Conclusion :**

  – Querying and Browsing have complementary advantages & limitations

  – Both are based on explicit manipulation of structure (though at different levels and for different purposes) :

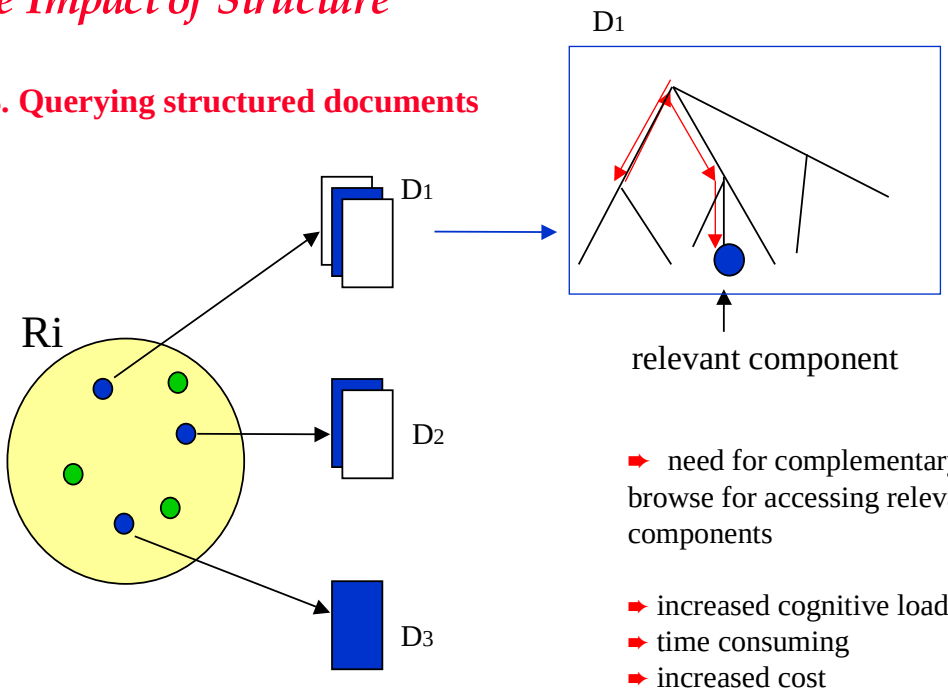
    – querying : attributes, logical structure

    – browsing : links

  – From the single point of view of interaction, there is a need to make a proper use of structure

The Impact of Structure

- **3. Querying structured documents**



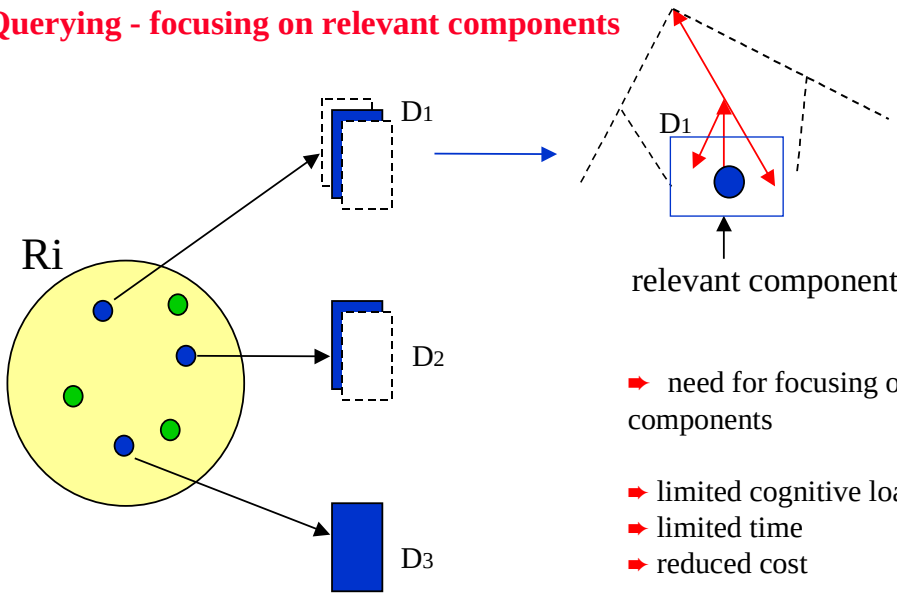
➨ need for complementary browse for accessing relevant components

➨ increased cognitive load,
➨ time consuming
➨ increased cost

➨ **Negative impact of ignoring the underlying structure**

The Impact of Structure

- **Querying - focusing on relevant components**



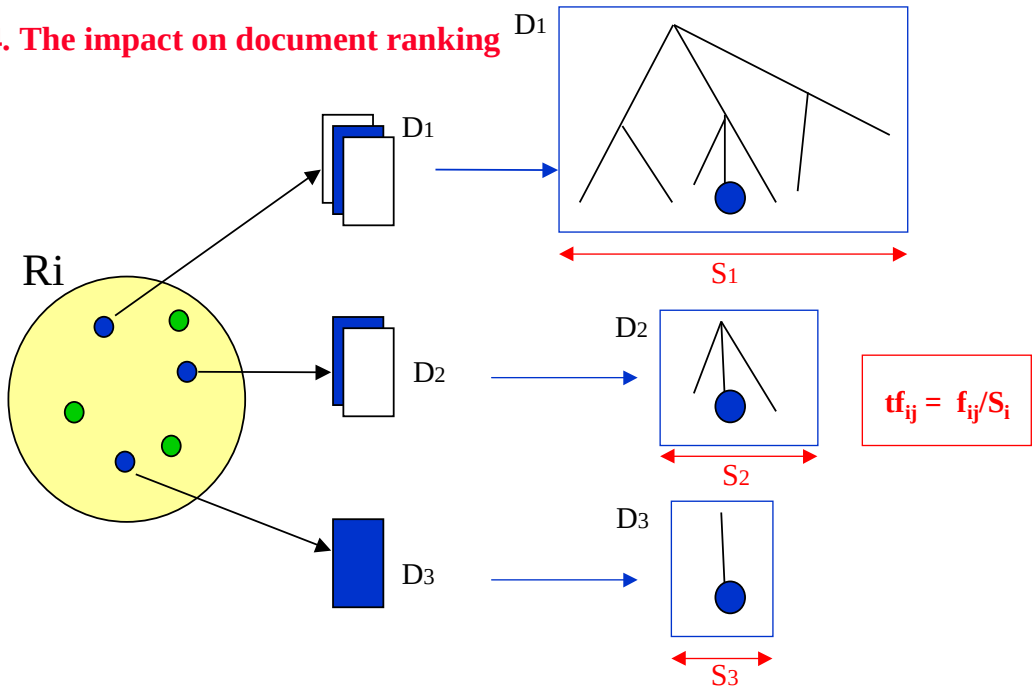
➨ need for focusing on relevant components

➨ limited cognitive load,
➨ limited time
➨ reduced cost

➨ **Notion of Document Specificity to the query**

The Impact of Structure

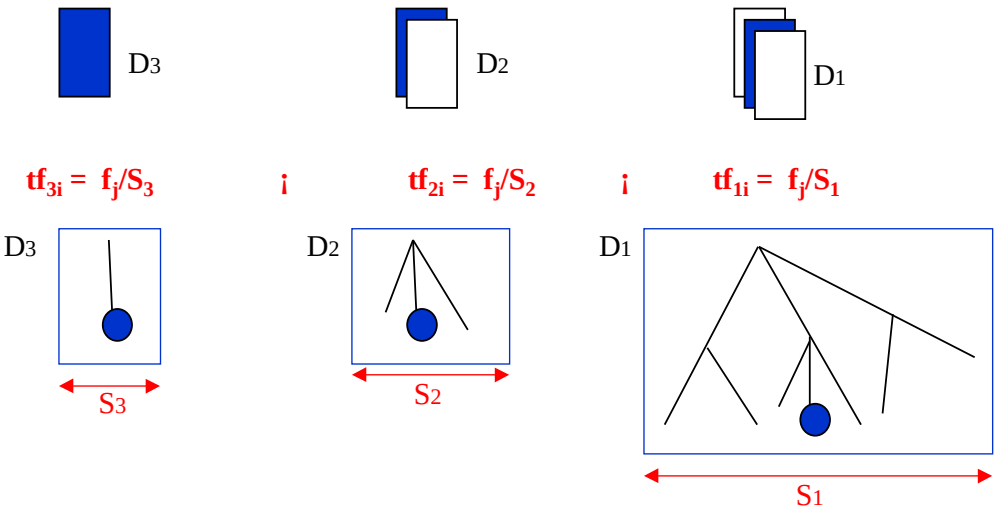
- **4. The impact on document ranking**



- S_i size of the whole (atomic) document
- term frequencies related to document size (concentration effect)

The Impact of Structure

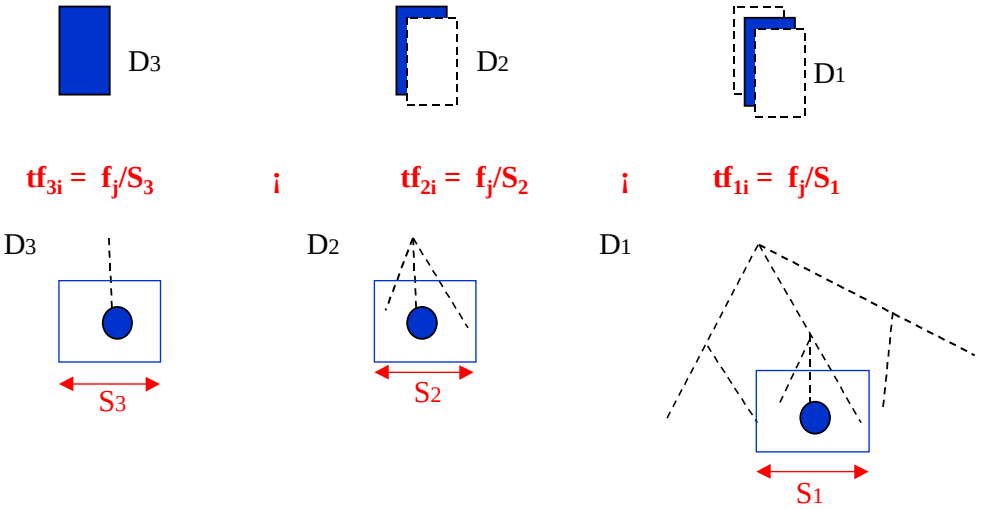
- **The impact on document ranking**



- larger documents tend to be low-ranked
- **Bad Incidence on recall / precision**

The Impact of Structure

- **The impact on document ranking**



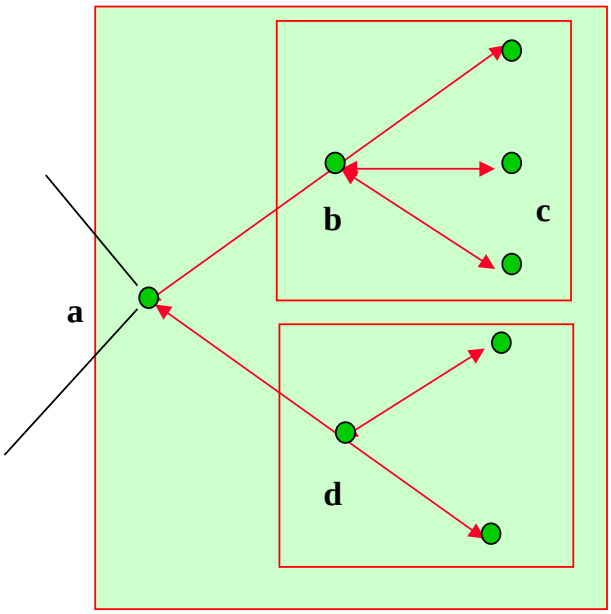
➨ similar components tend to be closely ranked (whatever the embedding document)

➨ **Good Incidence on recall / precision**

The Impact of Structure

- **5. Disorientation Problems : the example of Web pages**

Logical structure (abstract)



Ex: **a** is the homepage of a Web site

Links implement **navigation** structure, not the **logical** structure

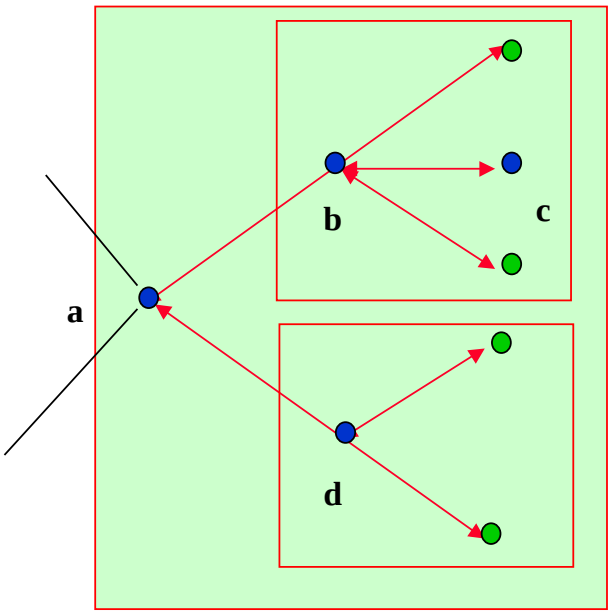
➨ page nodes are indexed separately

➨ index of page **a** does not represent the content of subtree **a** (only of page a)

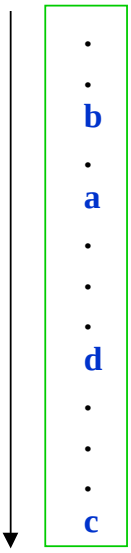
The Impact of Structure

- **Disorientation Problems : the example of Web pages**

Logical structure (abstract)



System response (linear list)



System ranking

Logical structure not visible

➨ redundant accesses (cognitive load, disorientation)

➨ the user has to re-engineer The structure to some extent

Approaches - An Integrated Model

- **Browsing/Querying : integration**

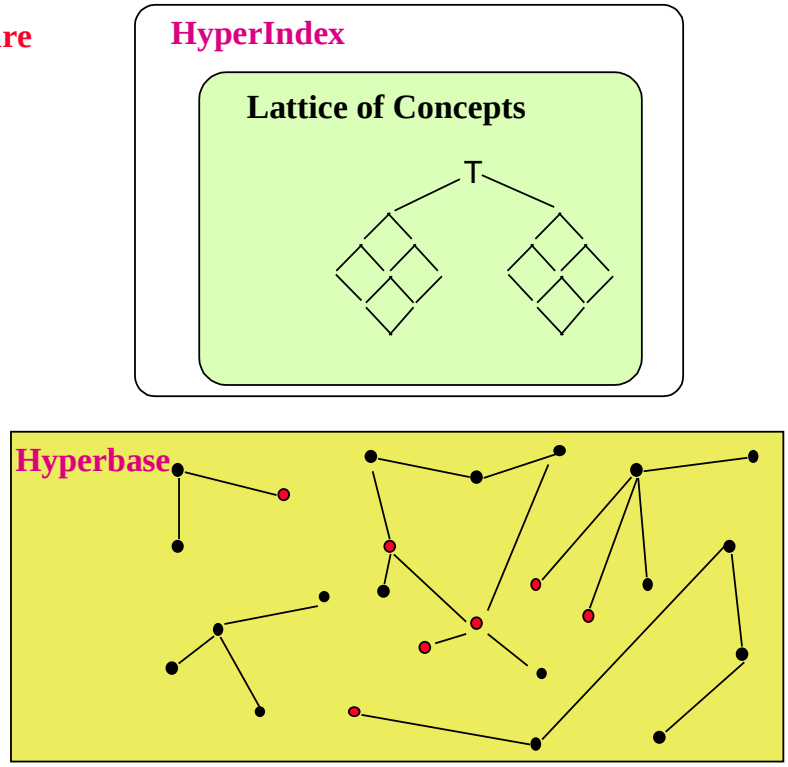
  – Extended IR model (hypermedia features):
    » considering the document structure

  – Extended Hypermedia Model :
    » content management
    » typed links
    » weighted links
    » link construction

content
and
structural knowledge
(conceptual graphs)

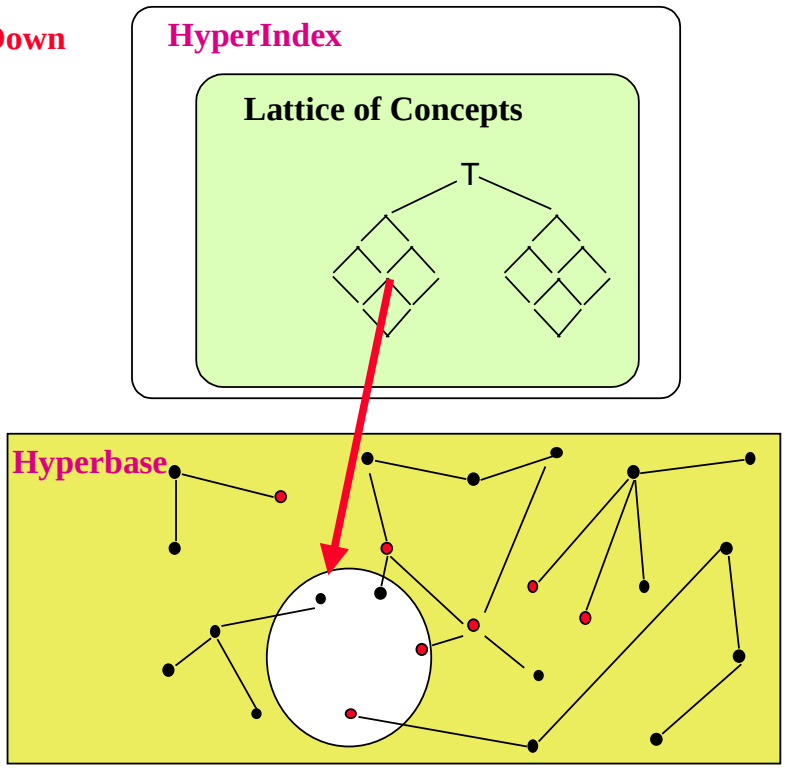
Approaches - An Integrated Model (hypermedia)

- **Structure**



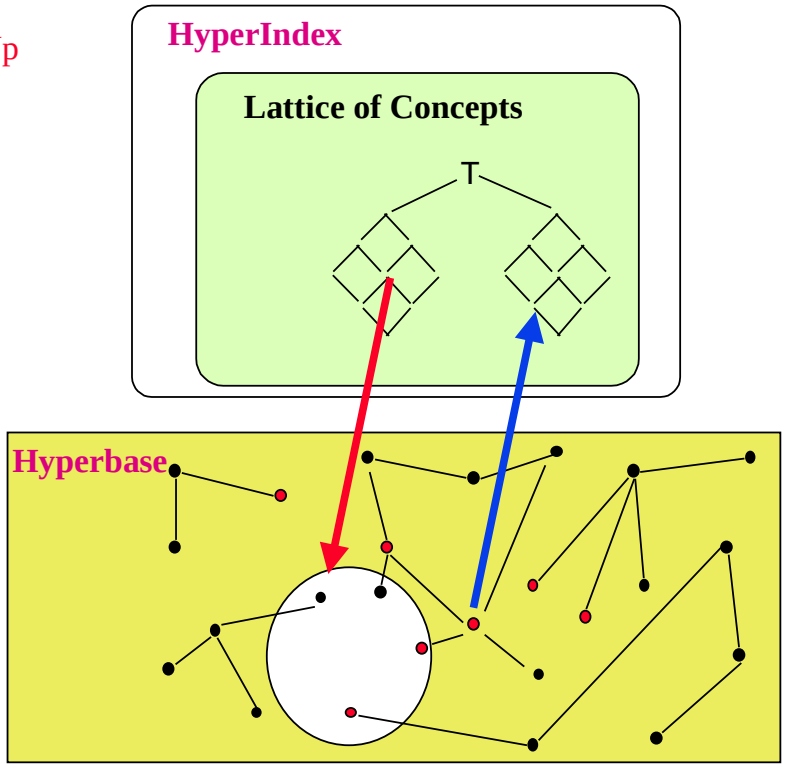
Approaches - An Integrated Model (hypermedia)

- **Beam Down**



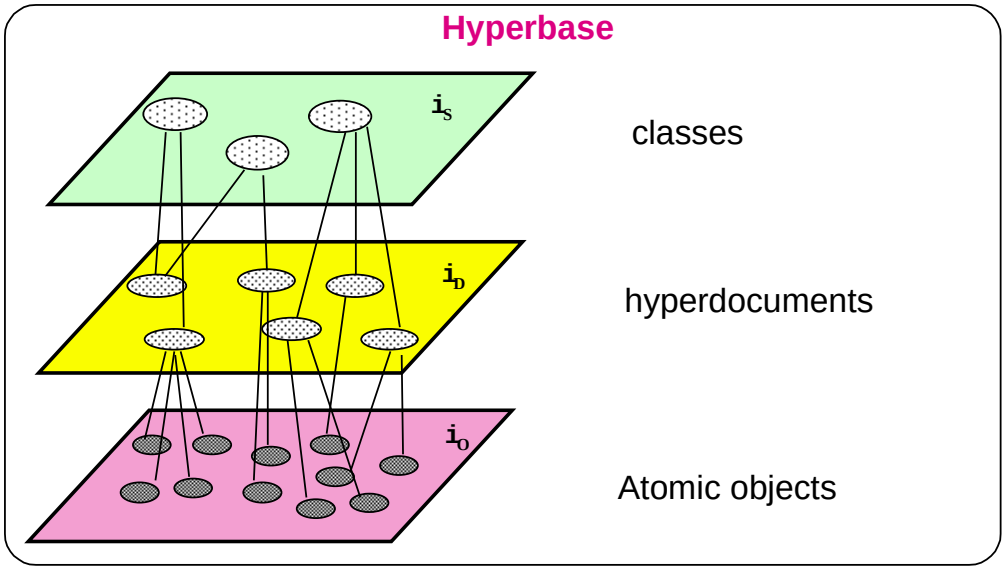
Approaches - An Integrated Model (hypermedia)

- Beam Up



Approaches - An Integrated Model (hypermedia)

- **Abstraction levels**

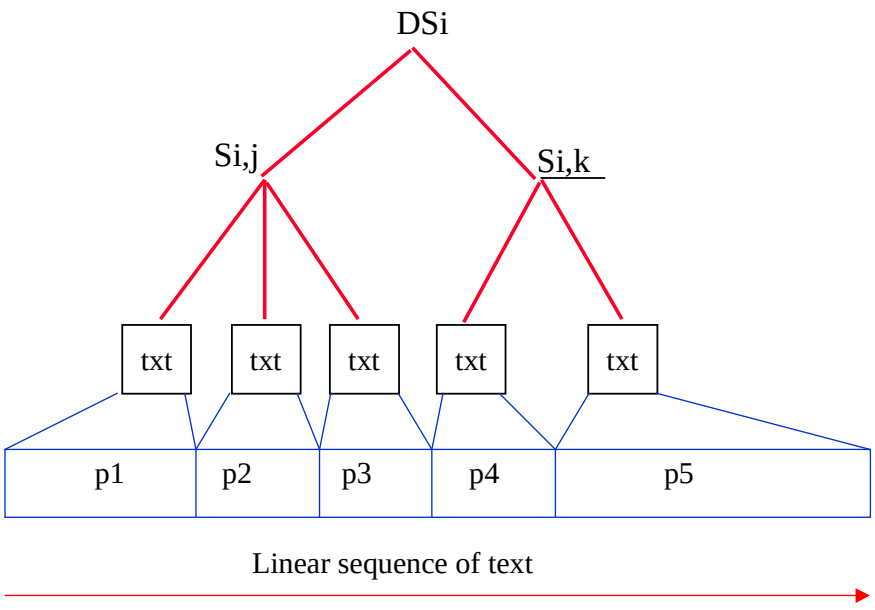


Approaches - An Integrated Model (hypermedia)

- **Conclusion**
  - Strong impact of Structure on Interaction Performances (cognitive load, disorientation, efficiency)
  - Strong impact of Structure on Retrieval Performances (effectiveness)
  - Integration (more than combination) of Browsing and Querying capabilities needed

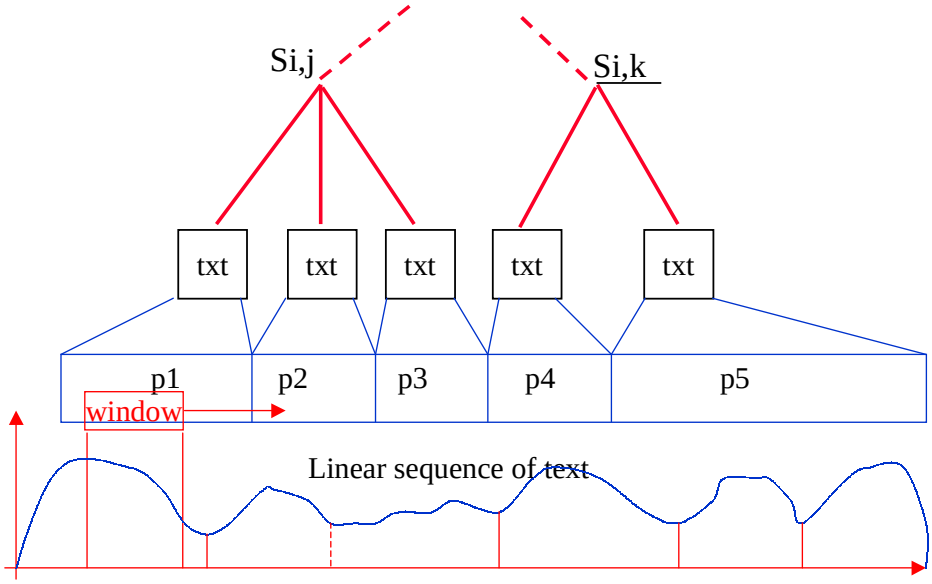
Approaches - Indexing Structured Documents

- **2. Passage Retrieval**



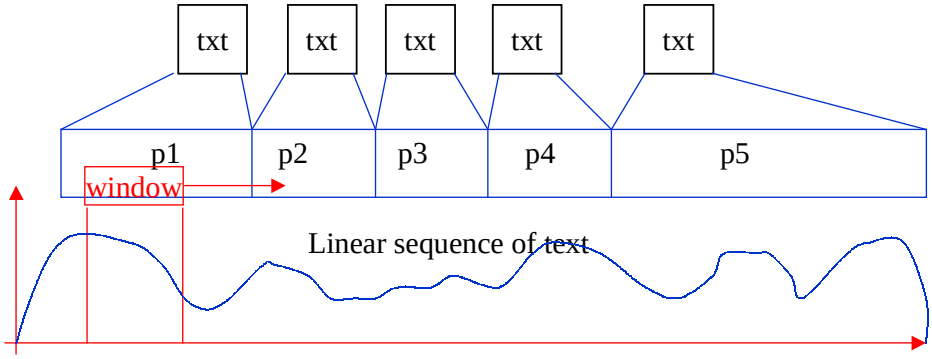
Approaches - Indexing Structured Documents

- **Passage Retrieval - segmenting passages**
  - **Computing local term density within a sliding window**
  - **Discontinuities denote topicality changes and passage boundaries**



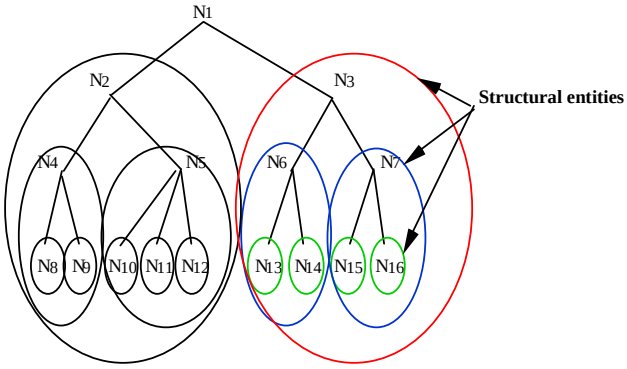
Approaches - Indexing Structured Documents

- **Passage Retrieval**
  - **advantages : simple, efficient, can be effective**
  - **limitations :**
    - **text only**
    - **incidence of document types**
    - ➡ needs to be tuned to document types



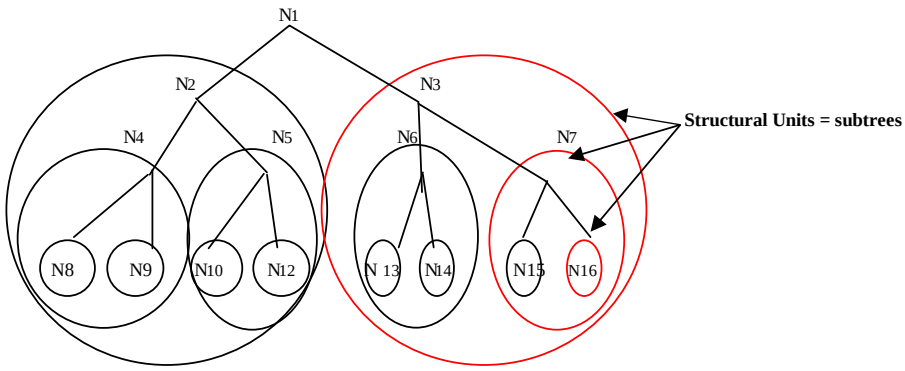
Approaches - Indexing Hierarchical Structures

- **3. Retrieving Structured Information**
  - » Focus on logical structure and links
  - » Corpus = { structural entities }
  - ➨ **Each unit indexed / retrieved independantly**



Approaches - Indexing Hierarchical Structures

- An important case : hierarchical structure (textual documents, Web sites presentations, video..)
- Non atomic view of Document = { Structural Units }

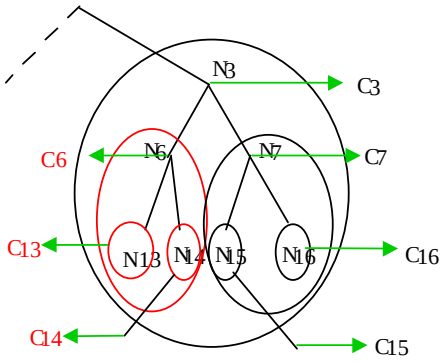


➨ Relationship between structure - semantic content : individual indexing of structural units

Approaches - Indexing Hierarchical Structures

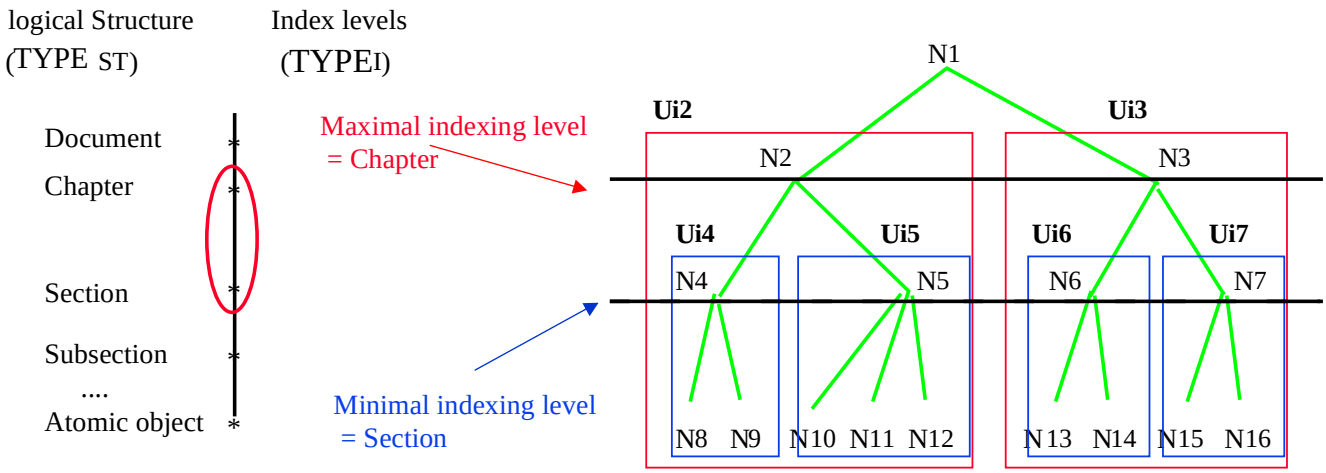
- **Relationship between Structure and Content (indexing)**
  - Aggregation (propagation of content)
  - Integration of various content models related to specific media : (text, image, speech..)

C6=f(C13, C14)
C7=f(C15, C16)
C3=f(C6, C7)



Approaches - Indexing Hierarchical Structures

- **Indexing Structured documents (Eg. IOTA -RIME)**
  - Structural Units vs Indexing Units



Approaches - Indexing Hierarchical Structures

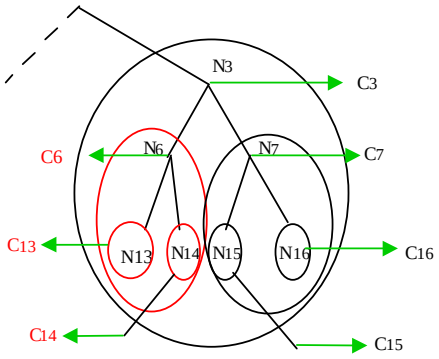
- **Propagation of values :Attribute Classes**

  - **Goal** : Infering information related to document components (content, attributes)
  - **Approach** : propagation of attribute values within the logical structure

C6=f(C13, C14)

C7=f(C15, C16)

C3=f(C6, C7)



Three attribute classes :

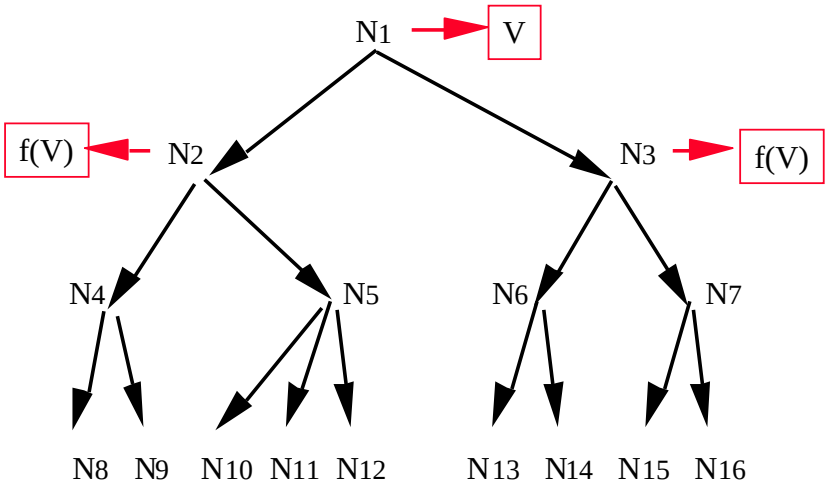
- Dynamic Descending Attributes
- Dynamic Ascending Attributes
- Static Attributes

Approaches - Indexing Hierarchical Structures

» **Descending Dynamic Attributes**

– $V_{a,i} = V \Rightarrow V_{a,ij} = f(V)$

– Ex: Publication date (f is identity)

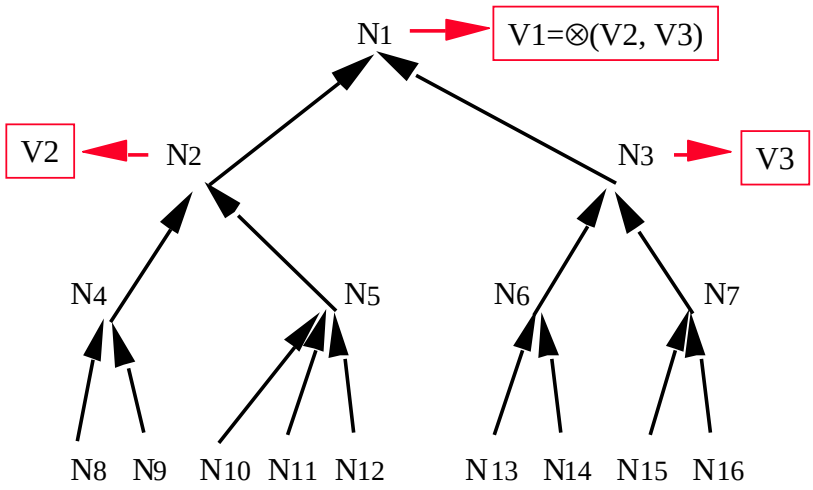


Approaches - Indexing Hierarchical Structures

» **Ascending Dynamic Attributes**

– $V_{a,i}=V_1,\ldots,V_{a,n}=V_n \Rightarrow V_{a,i}=\otimes(V_1,\ldots,V_n)$

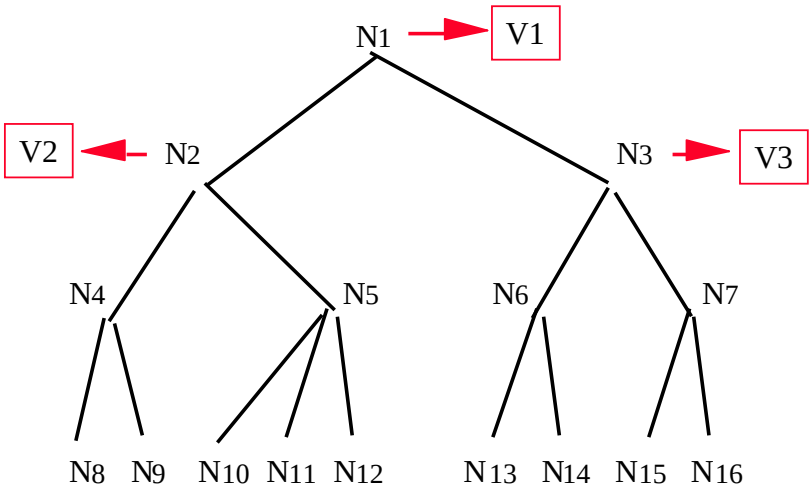
– Ex: Author, Content attributes



Approaches - Indexing Hierarchical Structures

» **Static Attributs**

– Ex: Titles



Approaches - Indexing Hierarchical Structures

- **Content Attribute**

  – Computation and assignation of values to the symbolic attribute (expressions of language $\mathcal{L}^{symbolic}$)

    » Indexing Units : structural components of type :

    $TYPE_I \subseteq TYPE_{ST}$

    » Ascending Propagation and Aggregation of index expressions :

    aggregation operator $\oplus_{symbolic}$.

Approaches - Indexing Hierarchical Structures

- **Indexing Multimedia Data based on attribute values :**
  – multivalued
  – domain values are expressions of a given language $\mathcal{L}$ <facet> <media>
  – Used for integrating single-media models in the framework of structured, multimedia documents:

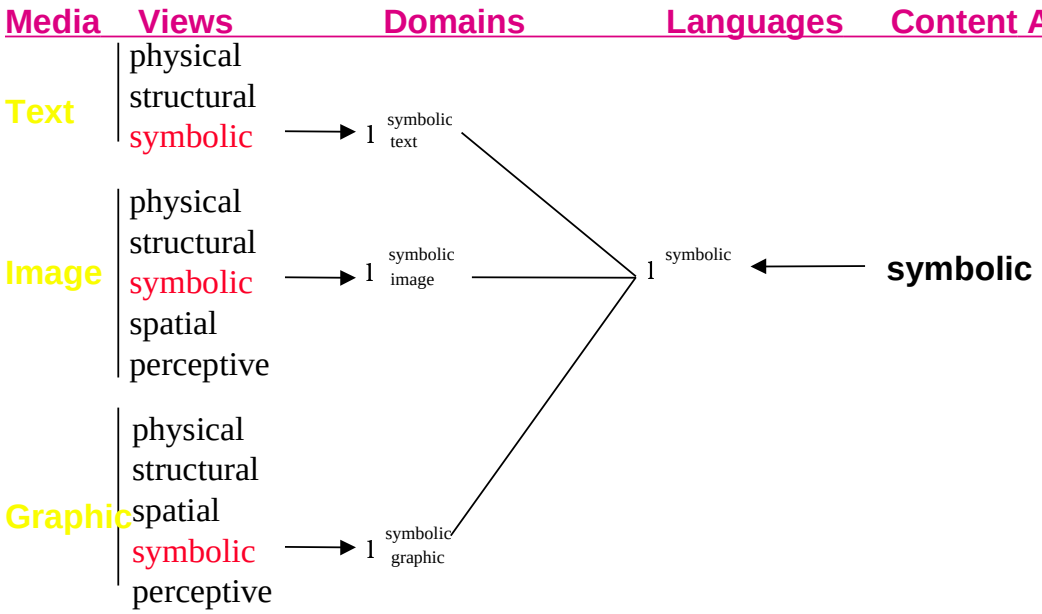
    » Types of Content Attributes (facets) :

    - Physical
    - Symbolic
    - Structural
    - Spatial
    - Perceptive

    correspond to views of single-media data

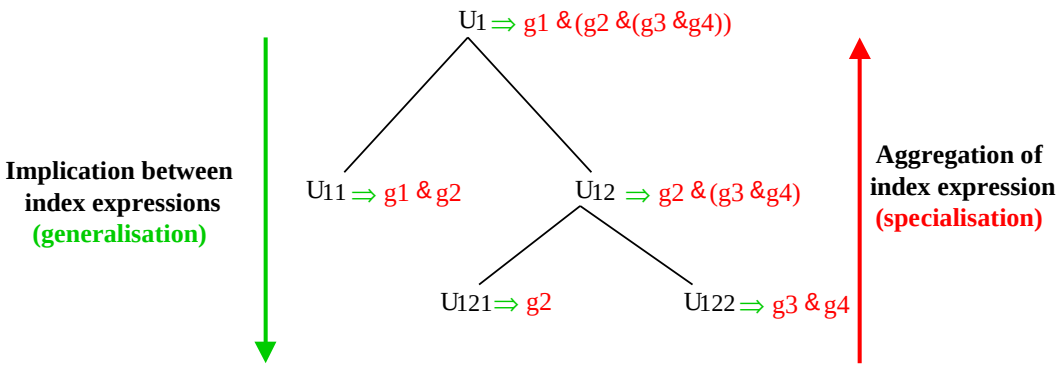
Approaches - Indexing Hierarchical Structures

- **Combining Content Attributes for different media**



Approaches - Indexing Hierarchical Structures

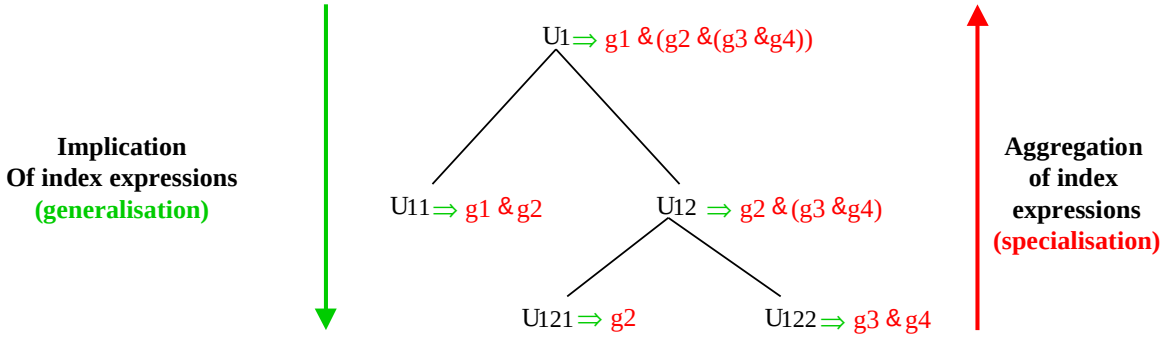
- **Indexing Structured Documents**
  - Content Attribute : Dynamic, Ascending
  - operator $\oplus_{symbolic}$ ascending aggregation of values of content attribute for each indexing unit :



Approaches - Indexing Hierarchical Structures

- **Properties of operator $\oplus_{symbolic}$**
  - Neutral element ε : $g \oplus \varepsilon = g$
  - Reflexivity :$g \oplus g = g$
  - Symmetry : $g \oplus f = f \oplus g$
  - Associativity : $(g \oplus f) \oplus h = g \oplus (f \oplus h)$

➨ **Determine a strategy for aggregating values**
➨ **Determine a strategy for dynamic indexing**

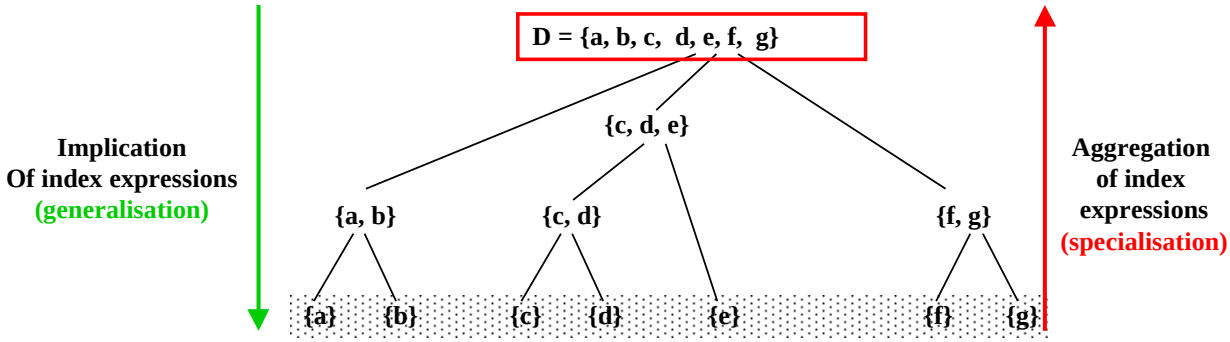


Approaches - Indexing Hierarchical Structures

- **Example :** set indexing - $t_{symbolic}$ = { index terms}

ascending strategy (specialisation of index expressions)

$\oplus_{symbolic}$ is the union operator



➨ **Could be extended to the aggregation of weighted terms**

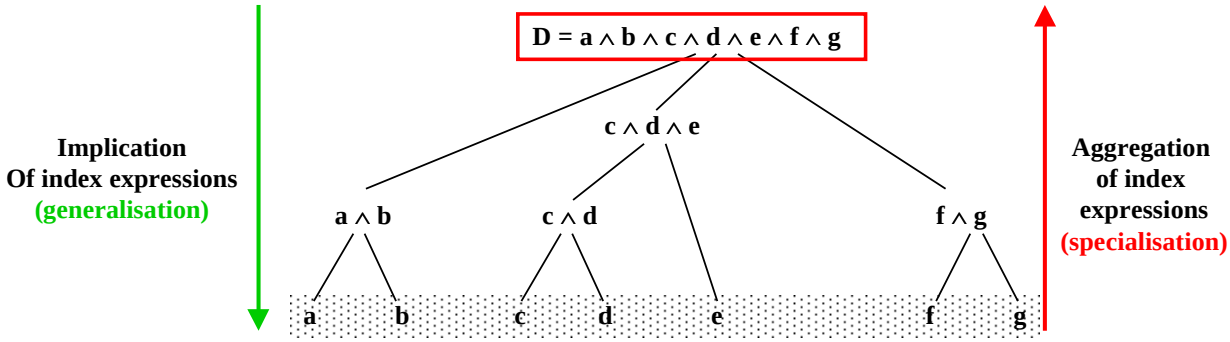
$(a, w_a) \oplus (b, w_b) = \{(a, f(w_a)), (b, f(w_b))\}$

Approaches - Indexing Hierarchical Structures

- **Example :** boolean indexing

  ascending strategy (specialisation of index expressions)

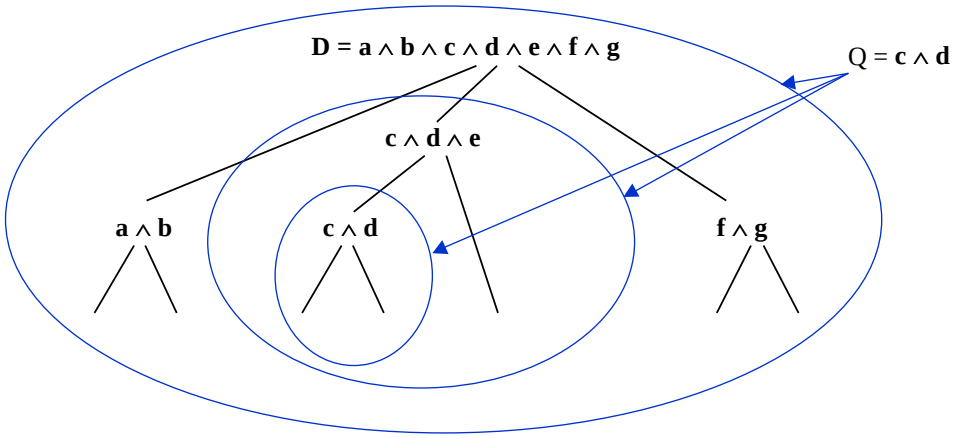
  $\oplus_{\text{symbolic}}$ is the **and** operator



Approaches - Retrieving Hierarchical Structures

- What strategy for retrieving indexed units ??
  - **A) Brute approach :**
    - » Example : boolean indexing

      direct application of $Di \supset Q$, where Di are indexing units



➨ **Non optimal, redundant answer**

Approaches - Retrieving Hierarchical Structures

- What strategy for retrieving indexed units ??
  - **B) Soft approach :**
    - » Example : boolean indexing

  - Selection of indexing units based on evaluation of

    $Di \supset Q$ (exhaustivity)

    and on

    $Q \supset Di$ (specificity)

  - ➡ If both conditions are matched, D is an exact match for Q
  - ➡ Else : the process looks for the smallest units (ie. deepest in the hierarchy) such that:

    $Di \supset Q$ (exhaustivity requirement)

    and

    $\neg(Q \supset Di)$ (specificity limitation)

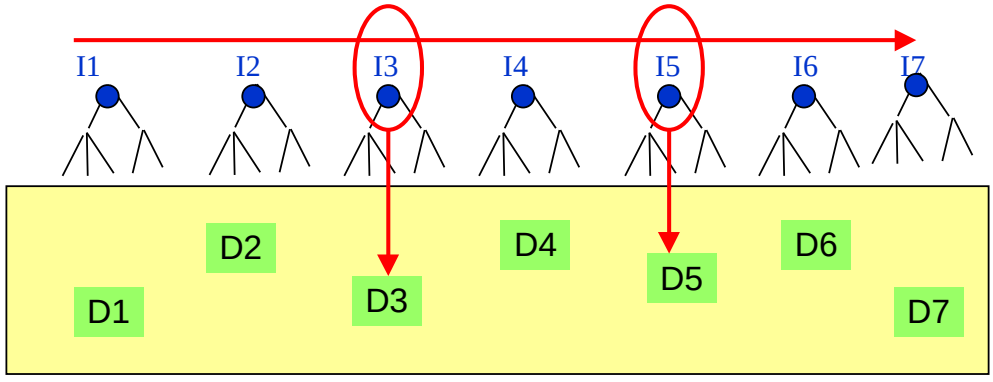
Approaches - Rerieving Hierarchical Structures

- **Example** : Fetch &Browse algorithm

  - **Fetch**

    "horizontal" **preselection** of **documents** D satisfying $D \supset Q$:

Q



CORPUS

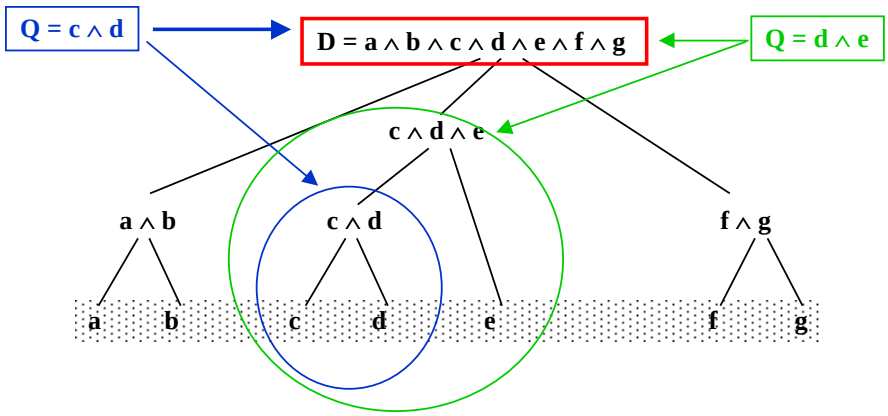
Approaches - Retrieving Hierarchical Structures

– **Browse**:

» "vertical" selection of most specific units within preselected documents (fetch) :

- Recursive Case : $D_i \supset Q$ and $\neg(Q \supset D_i)$
- Stop Case : $(D_i \supset Q$ and $Q \supset D_i) \Rightarrow$ result = D_i

or $\neg(D_i \supset Q) \Rightarrow$ result = Father(D_i)



Conclusions & Perspectives

– Considering (ie. making explicit) structure leads to better retrieval performances in terms of interactive characteristics and classical retrieval performances (precision, recall)

– Considering structure may allow for the integration of browsing vs querying as complementary ways for retrieving information

– Considering structure may allow for the integration of various media in a unified indexing/retrieval strategy

– Considering structure may help in improving focus / precision, a much needed improvement for searching the Web

– Considering structure implies better understanding of the core notions of document, user needs, document relevance