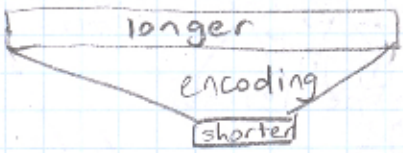
- Signature vs Inverted Files – Zobel, Moffat
- IEEE Trans. on Data and Knowledge Eng. 2008 - Signature Trees
- Lexicon Search
- Alan Tharp, File Organization and Processing – Chap. 5

$$D = \begin{array}{c} \begin{matrix} t_1 & t_2 & \cdots & t_n \end{matrix} \\ \begin{bmatrix} 0 & 1 & \cdots & 1 \\ & & & \\ & & & \\ & & & \end{bmatrix} \end{array}$$

INSPEC

$n = 16000$

doc array / vector $= \frac{16000}{8} = 2000$ bytes/doc

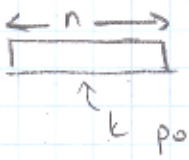


$t_1 \rightarrow 01000001$

$t_2 \rightarrow 10000010$

11000011 ← superimpose

↖ superimposed code



- For each term have a different term signature

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

$$x + y = n \Rightarrow \binom{n}{x} = \binom{n}{y}$$

How to generate term signatures

Set all term signature bit positions equal to 0

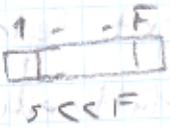
$r \leftarrow$ map term into a number using a hash function

$i \leftarrow 1$

while $i \leq s$ { (s = no. of bit position that we want to let)

bitPosition ← max(1, F.random(r))

// F : signature length

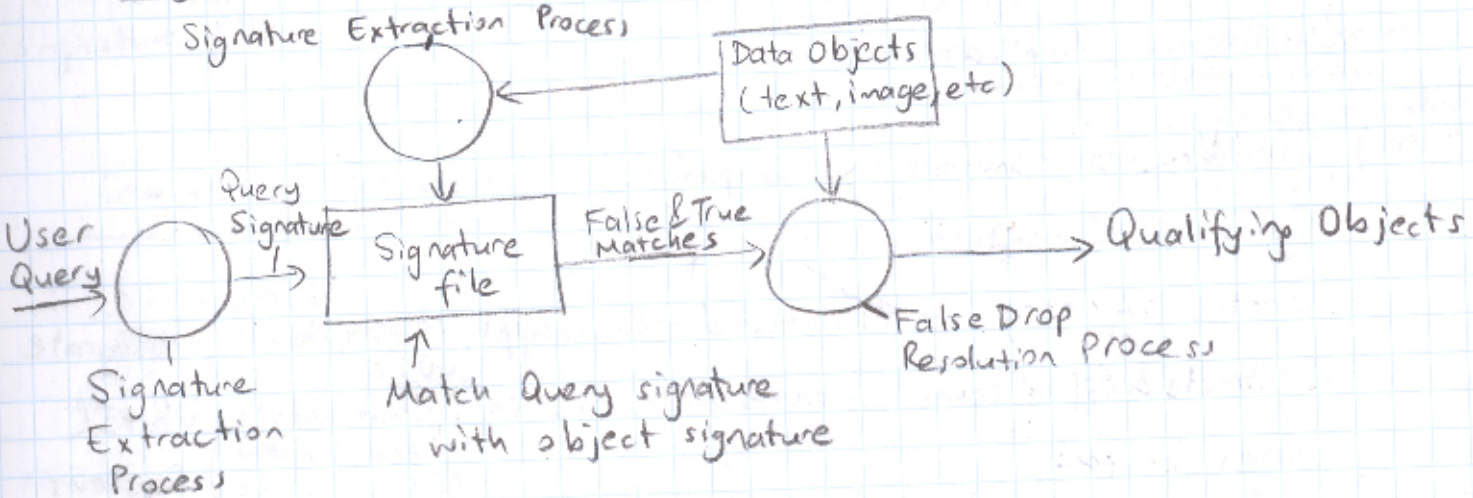


termSignature [bitPosition] ← 1

i ← i+1

}

Signature-based Retrieval Process



Example

Block

Object Signature Generation

term	term signature
object	1000 1000
signature	0010 0100
generation	1000 0100
	1010 1100 → Logical block signature

- If Q_s and $B_s = Q_s$
  then block may contain query terms

query	query signature	result
database	1100 0000	no match
generation	1000 0100	true match
information	1010 0000	false match

Use of Signatures in Various Applications

1- Text Retrieval

2 - Multimedia Applications

3- Database Systems

4- In Prolog databases to store and access knowledge

5. CAD - Computer Aided Design

6- OODB - Object Oriented Database Design

Multimedia Applications

a-) Consider a manufacturing company

Types of information

Parts: inventory → structured info → apply db methods } Integrate DB+IR Systems

Contracts & office communication → unstructured info } Multimedia component

Images for parts

Medical Application : MINUS, MULTOS

How to use signatures for Images

Image → [Image Analysis Process] → Description of Images in terms of objects involved

P. Zezula et al

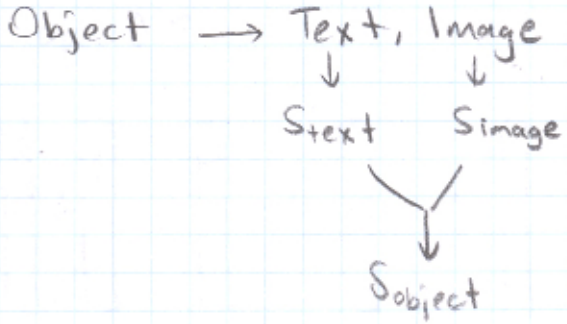
"Dynamic Partitioning of Signature Files"

ACM TOIS, Oct 1991, Vol. 9, No 4, pp 336-367

Image 

$I = O_1, O_2(O_4, O_5(O_6)), O_3(O_7, O_{10})\ldots$

↑ simple object



Storage Structure for Signature Files

Single Level
- Sequential signature files

Multi level
- Signature tree
- Two level

Vertical Partitioning
- Bit sliced
- Frame sliced

Horizontal Partitioning
- Quick Filter
- LHSS: Linear Hashing with Superimposed Coding
- Key based
- Gustafsen

Sequential Signatures (SS)

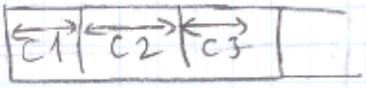
S1	0001	1110
S2	1101	0001
S3	0011	1100
S4	1100	0011
S5	0011	0110
S6	1100	1001

- No overhead in terms of storage
- Easy insertion
- In case deletions needs compaction
- Query processing time: Proportional to file size (disadv.)
  Independent of query weight* (adv.)
  * weight: # of 1's in a signature

Bit-Sliced Signatures (BS)

column no →

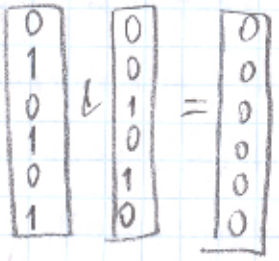
C1:	010	101
C2	010	101
C3	001	010
C4	111	010
C5	101	001
C6	101	010
C7	100	110
C8	010	101



- Insertions can be difficult.
- Cost of query processing is proportional to query weight.

Q_s : 1011 0000

c1 c3 c4

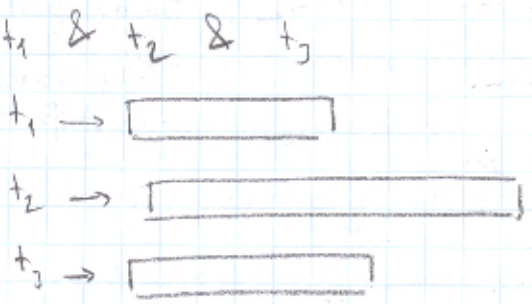


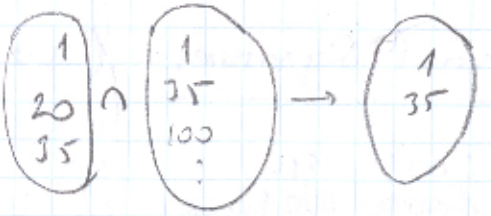
c1 c3 intermediate result

- Partial Evaluation in Bit Slice Processing
- Partial Evaluation in Inverted Files

Conjunctive Queries

t_1 & t_2 & t_3



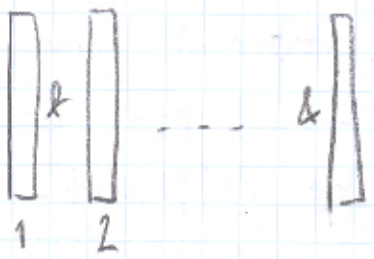


Partial Evaluation of Bit-Sliced Signature Files

$op = \frac{\text{no. of 1's in a signature}}{\text{signature length}}$: on bit density

fd: false drop probability $= \frac{\text{no. of matches}}{\text{no. of signatures } (N)}$

- Note that no. of true matches is very small $\Rightarrow$ we assume that all matches are false drops.



After processing i no. of bits

$fd_i = op^i$

$op = 0.5$

Response Time After Processing i no of bits = $RT(i)$

$$RT(i) = i \cdot T_{slice} + N \cdot op^{i} \cdot T_{resolve}$$

($N \cdot op^{i}$: no. of matching signatures)

To find i value for the minimum response time, we take the derivative of $RT(i)$ with respect to i.

$$\frac{dRT(i)}{di} = T_{slice} + N \cdot T_{resolve} \cdot op^{i} \cdot \ln op$$

To find the optimum value of i let the above equation equal to 0 and solve it

$$op^{i} = \frac{T_{slice}}{N \cdot T_{resolve} \cdot (-\ln op)}$$

$$\ln op^{i} = i \ln op = \ln \frac{T_{slice}}{N \cdot T_{resolve} (-\ln op)}$$

$$i = \ln \left(\frac{T_{slice}}{N \cdot T_{resolve} (-\ln op)} \right) / \ln op$$

if $i > w(Q)_t$ = (no. of bits in a t term query)

then $i = w(Q)_t$

$i = [1, \underbrace{w(Q)_t}_{\text{query weight}}]$