Computing Appropriate Representations for Multidimensional Data

Choong, Laurent & Marcel

Université François Rabelais de Tours

FRANCE

Before Restructuring

World Consumption (US$bil)

Sales: 2000	Beer	Water	Soda	Wine	Milk
Europe	4	4	7	6	5
America	4	5	7	7	6
Asia	3	3	6	5	5
Africa	2	2	6	5	4

After Restructuring

World Consumption (US$bil)

Sales: 2000	Beer	Water	Milk	Wine	Soda
America	4	5	6	7	7
Europe	4	4	5	6	7
Asia	3	3	5	5	6
Africa	2	2	4	5	6

“Switch”

	Beer	Water	Soda	Wine	Milk
Europe	4	4	7	6	5
America	4	5	7	7	6
Asia	3	3	6	5	5
Africa	2	2	6	5	4

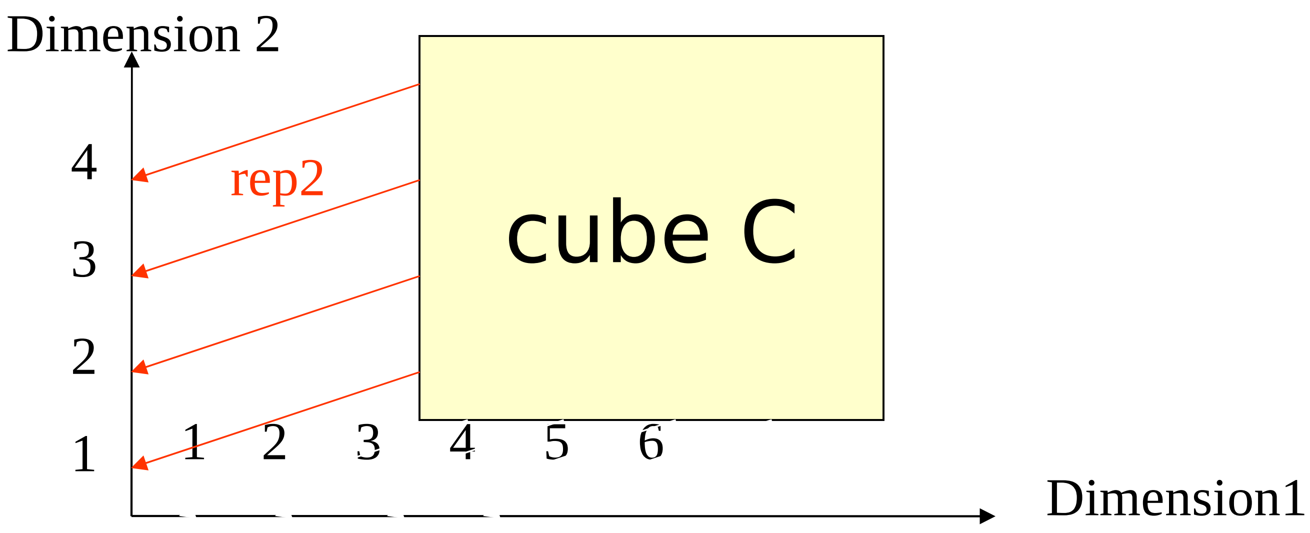
	Beer	Water	Milk	Wine	Soda
America	4	5	6	7	7
Europe	4	4	5	6	7
Asia	3	3	5	5	6
Africa	2	2	4	5	6

Motivation

- Many representations of a given cube
  - Constructed by the User
- What are the most appropriate representations
  - Quality of a Representation
- How to compute these Representations
  - "switch" operator

Representation

- Given a n-dimensional cube C, a representation of C is a set of n mappings:
  - one mapping per dimension
  - associates each member to an integer



Example: 2-dimensional cube

< C, {x, y}, {a, b}, {1, 2, 3, 4}, m_c > where

m_c **(x, a) = 1,** m_c **(y, a) = 2,** m_c **(x, b) = 3,** m_c **(y, b) = 4**

Four possible representations:

R_1

	x	y
2 a	**1**	**2**
1 b	**3**	**4**

R_2

	x	y
2 b	**3**	**4**
1 a	**1**	**2**

R_3

	y	x
2 b	**4**	**3**
1 a	**2**	**1**

R_4

	y	x
2 a	**2**	**1**
1 b	**4**	**3**

Note:

	y	x
a	1	3
b	2	4

Not a representation of C

Position of a Cell

R_1

		x	y
2	a	**1**	**2**
1	b	**3**	**4**

The position of cell $c_1 = < x, a, 1 >$ **is** $< 1, 2 >$

The position of cell $c_2 = < y, b, 4 >$ **is** $< 2, 1 >$

Cell Ordering

- A cube $\boldsymbol{C} = < C, dom_1, \ldots, dom_n, dom_m, m_c >$
- A representation $\boldsymbol{R_c} = \{rep_1, \ldots, rep_n\}$
- Cells $\boldsymbol{c} = < m_1, \ldots, m_n, m >$

  $\boldsymbol{c'} = < m_1', \ldots, m_n', m' >$

$c <_{Rc} c'$ **iff** $\forall i \in [1, \ldots, n], rep_i(m_i) \leq rep_i(m_i')$

- $<_{Rc}$ **is a partial ordering**

Cell Ordering – an example

$c_1 = < x, a, 1 >$ $c_3 = < x, b, 3 >$

$c_2 = < y, a, 2 >$ $c_4 = < y, b, 4 >$

R

	x	y
a	**1**	**2**
b	**3**	**4**

We have:

$c_3 <_R c_1$ $c_3 <_R c_2$ $c_3 <_R c_4$

$c_1 <_R c_2$ $c_4 <_R c_2$

Note that c_1 cannot be compared to c_4

Misplaced Cell

- Given a representation $\boldsymbol{R_c}$ of a cube $\boldsymbol{C}$

- $\boldsymbol{c} = < m_1, \ldots, m_n, m >$ is **misplaced w.r.t.** $\boldsymbol{R_c}$ if
  1. $m \neq \perp$

  and

  2. $\exists\ c_1 = < m_1', \ldots, m_n', m' > \in \boldsymbol{C}$ such that

     $c <_{Rc} c_1$ and $m > m'$

     or

     $\exists\ c_2 = < m_1'', \ldots, m_n'', m'' > \in \boldsymbol{C}$ such that

     $c_2 <_{Rc} c$ and $m'' > m$

Characterizing the Representation

Given a representation R_C of a cube C

$M_{Rc}(C)$: number of misplaced cells in C w.r.t. R_c

- R_c is a ***Perfect Representation (PR)*** if

  $M_{Rc}(C) = 0$

  (i.e. there are no misplaced cells w.r.t. R_c)

- R_c is an ***Optimal Representation (OR)*** if

  $\forall R'_c, \quad M_{R'c}(C) \geq M_{Rc}(C)$

  (i.e. there is no other 'better' representation)

Switching

$$switch(j, p, q)(R_c) = R'_c$$

R'_c is obtained from R_c by permutation of rows $\boldsymbol{p}$ and $\boldsymbol{q}$ of dimension $\boldsymbol{j}$

Example

R_1

	x	y
a	**1**	**2**
b	**3**	**4**

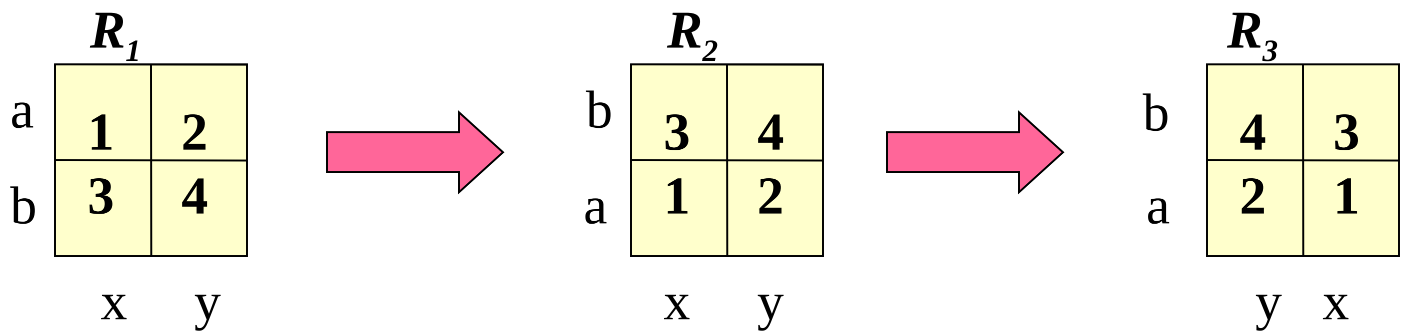
$switch(1, a, b)(R_1)$

R_2

	x	y
b	**3**	**4**
a	**1**	**2**

Arrangement

An **arrangement** is a finite composition of switches



$$\mathit{switch}(1, a, b)(R_1) = R_2 \qquad \mathit{switch}(2, x, y)(R_2) = R_3$$

$$\mathit{switch}(2, x, y)(\mathit{switch}(1, a, b)(R_1)) = R_3$$

Notation: $R_3 = \mathit{arr}(R_1)$

PR Problem

For a given cube and a given representation of this cube,

- **Test whether there exists at least one PR**
  - **If so, compute one PR**

- **If there are more than one PR**
  - **Compute the number of PRs**
  - **List all the arrangements leading to these PRs**

Basic Theorem

A representation of a cube is a PR
if and only if
every row in every dimension is sorted

Sketch of the proof:

- *If:* Trivial since if a representation is a PR then every row in every dimension must be sorted
- *Only if:* Consider the following example which can *not* be a PR. If every row is sorted then we must have: $X \geq 1, Y \geq 1, X \leq 0, Y \leq 0$

impossible

R

	x	y
a	**X**	**0**
b	**1**	**Y**

Case 1
No duplicates and no null values in each row

- There exists *at most* one *PR* of a given cube C

- If there exists a representation such that for one dimension, a row *r* is sorted and another row *r'* is *not* sorted, then there exists no *PR*

- If a *PR* exists, then it can be obtained by sorting *only* one row in each dimension

Case 1

No duplicates and no null values in each row

input: The representation of a cube *C*
output: The PR of *C* or the indication “no PR”

For each dimension *k* of *C* do:
    choose a row *r* in *k*
    sort *r*
    for every row *r’* in k do
        check if *r’* is sorted
        if *r’* is unsorted then
            exit with output “non PR”

Case 2

Dealing with duplicates & no null values

R_1

	x	y
b	**4**	**3**
a	**1**	**1**

R_2

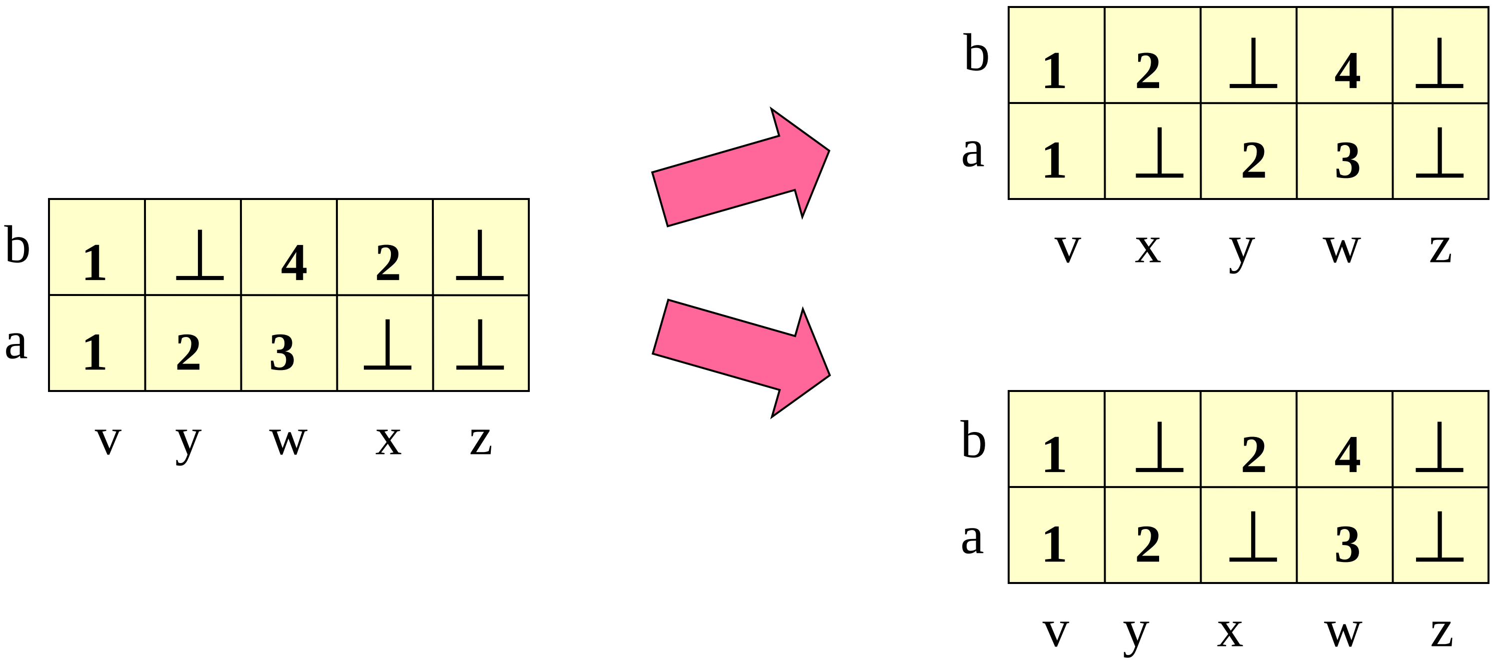
	y	x
b	**3**	**4**
a	**1**	**1**

Sorting row < **a** > may lead to representation $\boldsymbol{R_1}$ which is **not** perfect since row < **b** > is not sorted

Sorting row < **b** > leaves row < **a** > **unchanged** and gives a PR

Case 3

Dealing with null values



Null values

R_1

b	$\perp$	**0**	**sorted**
a	**1**	$\perp$	**sorted**
	x	y	
	sorted	**sorted**	

R_2

b	**0**	$\perp$
a	$\perp$	**1**
	y	x

But R_1 is not a PR !

R_2 is a PR

R_1 is a Weak PR (WPR)

Open issues

- PR Problem with Null Values
- Introducing an Efficient Implementation
- Identifying all ORs and their Arrangements
- Use other OLAP operations (e.g. roll-up)
- *t*-OR Problem: given a theshold $\boldsymbol{t}$, find $\boldsymbol{R_c}$ of C^c such that

$$M_R\ (\mathbf{C})\ \leq \boldsymbol{t}$$

What we did next

- Investigate AI search techniques to compute representations of good quality
  - Genetic algorithm
  - Hill climbing