University of Windsor

05 DDBMS Design

Kalyani Selvarajah
School of Computer Science
University of Windsor

Advanced Database Topics
COMP 8157 01
Fall 2023

Announcement

Assignment 1 Due: October 22/23/24 2023 as per the section

Today's Agenda

Distributed Database Design

Data Allocation

Fragmentation

https://domains.upperlink.ng/elementor-947/

Introductory Questions

What are the important things need to be considered in data allocation?

What is fragmentation?

Why is fragmentation important in DDBMS?

How can we do fragmentation?

Distributed Database Design

Fragmentation

A relation may be divided into a number of sub-relations.

Two main types:

- Horizontal fragments are subsets of tuples.
- Vertical fragments are subsets of attributes.

Uses qualitative information.

Allocation

Each fragment is stored at the site with “optimal” distribution.

uses quantitative information.

Replication

The DDBMS may maintain a copy of a fragment at several different sites.

Distributed Database Design

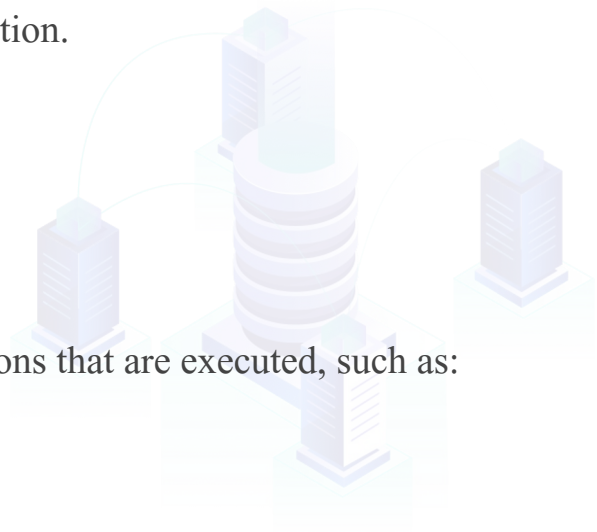
The design should be based on both quantitative and qualitative information.

The quantitative information may include:

- the frequency with which a transaction is run;
- the site from which a transaction is run;
- the performance criteria for transactions.

The qualitative information may include information about the transactions that are executed, such as:

- the relations, attributes, and tuples accessed;
- the type of access (read or write);
- the predicates of read operations.

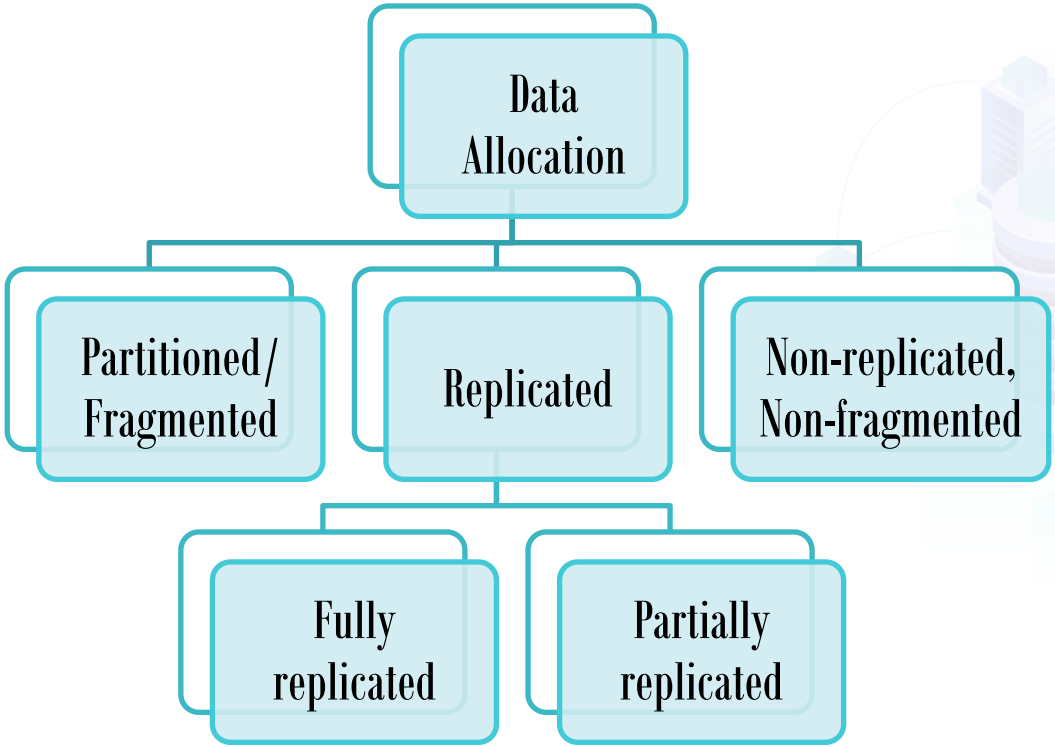


How the database is to be used?

Definition and allocation of fragments carried out strategically to achieve following objectives:

1. **Locality of reference**: Where possible, data should be stored close to where it is used. If a fragment is used at several sites, it may be advantageous to store copies of the fragment at these sites.
2. **Improved reliability and availability:** Reliability and availability are improved by replication.
3. **Acceptable performance**: Bad allocation may result in bottlenecks occurring.
4. **Balanced storage capacities and costs:** Consideration should be given to the availability and cost of storage at each site, so that cheap mass storage can be used where possible.
5. **Minimal communication costs:** Consideration should be given to the cost of remote requests.

Data Allocation



Comparison of Strategies for Data Allocation

	LOCALITY OF REFERENCE	RELIABILITY AND AVAILABILITY	PERFORMANCE	STORAGE COSTS	COMMUNICATION COSTS
Centralized					
Fragmented					
Complete replication					
Selective replication					

Comparison of Strategies for Data Allocation

	LOCALITY OF REFERENCE	RELIABILITY AND AVAILABILITY	PERFORMANCE	STORAGE COSTS	COMMUNICATION COSTS
Centralized	Lowest	Lowest	Unsatisfactory	Lowest	Highest
Fragmented	High	Low	Satisfactory	Lowest	Low
Complete replication	Highest	Highest	Best for read	Highest	High for update; low for read
Selective replication	High	Low	Satisfactory	Average	Low

Why Fragmentation?

Usage: Applications work with views rather than entire relations.

Parallelism: transaction can be divided into several subqueries that operate on fragments.

Efficiency: Data is stored close to where it is most frequently used. Data that is not needed by local applications is not stored.

Security: Data not required by local applications is not stored and so not available to unauthorized users.

Disadvantages with Fragmentation

Performance: The performance of global applications that require data from several fragments located at different sites may be slower.

Integrity: Integrity control may be more difficult if data and functional dependencies are fragmented and located at different sites.

Correctness of fragmentation

Fragmentation cannot be carried out randomly. There are **three rules** that must be followed during fragmentation:

Completeness: If relation R is decomposed into fragments R1, R2, ... Rn, each data item that can be found in R must appear in at least one fragment.

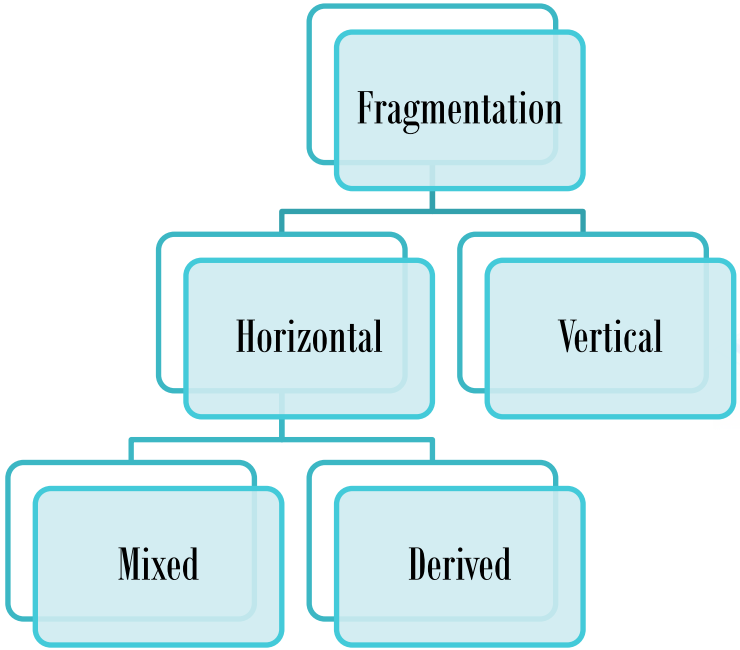
Reconstruction: Must be possible to define a relational operation that will reconstruct R from the fragments. Reconstruction for horizontal fragmentation is Union operation and Join for vertical .

Disjointness: If data item d_i appears in fragment Ri, then it should not appear in any other fragment. Exception: vertical fragmentation, where primary key attributes must be repeated to allow reconstruction.

For horizontal fragmentation, data item is a tuple.

For vertical fragmentation, data item is an attribute.

Types of fragmentation



Instance of the DreamHome rental database.

Branch

BranchNo	Street	City	Postcode
B005	22 Deer Rd	London	SW1 4EH
B007	16 Argyll St	Aberdeen	AB2 3SU
B003	163 Main St	Glasgow	G11 9QX
B004	32 Manse Rd	Bristol	BS99 1NZ
B002	56 Clover Dr	London	NW10 6EU

PropertyForRent

PropertyNo	Street	City	Postcode	Type	Rooms	Rent	OwnerNo	StaffNo	BranchNo
PA14	16 Holhead	Aberdeen	AB7 5SU	House	6	650	C046	SA9	B007
PL94	6 Argyll St	London	NW2	Flat	4	400	C087	SL41	B005
PG4	6 Lawrence St	Glasgow	G11 9QX	Flat	3	350	C040		B003
PG36	2 Manor Rd	Glasgow	G32 4QX	Flat	3	375	C093	SG37	B003
PG21	18 Dale Rd	Glasgow	G12	House	5	600	C087	SG37	B003
PG16	5 Novar Dr	Glasgow	G12 9AX	Flat	4	450	C093	SG14	B003

Staff

StaffNo	fName	lName	Position	Sex	DOB	Salary	BranchNo
SL21	John	White	Manager	M	1-Oct-45	30000	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24-Mar-58	18000	B003
SA9	Mary	Howe	Assistant	F	19-Feb-70	9000	B007
SG5	Susan	Brand	Manager	F	3-Jun-40	24000	B003
SL41	Julie	Lee	Assistant	F	13-Jun-65	9000	B005

Client

clientNo	fName	lName	telNo	prefType	maxRent	eMail
CR76	John	Kay	0207-774-5632	Flat	425	ohn.kay@gmail.com
CR56	Aline	Stewart	0141-848-1825	Flat	350	astewart@hotmail.com
CR74	Mike	Ritchie	01475-392178	House	750	mritchie01@yahoo.co.uk
CR62	Mary	Tregear	01224-196720	Flat	600	maryt@hotmail.co.uk

Instance of the DreamHome rental database.

PrivateOwner

ownerNo	fName	lName	address	telNo	eMail	password
CO46	Joe	Keogh	2 Fergus Dr, Aberdeen AB2 7SX	01224-861212	jkeogh@lhh.com	********
CO87	Carol	Farrel	6 Achray St, Glasgow G32 9DX	0141-357-7419	cfarrel@gmail.com	********
CO40	Tina	Murphy	63 Well St, Glasgow G42	0141-943-1728	tinam@hotmail.com	********
CO93	Tony	Shaw	12 Park Pl, Glasgow G4 0QR	0141-225-7025	tony.shaw@ark.com	********

Registration

clientNo	branchNo	staffNo	dateJoined
CR76	B00S	SL41	2-Jan- 13
CR56	8003	SG37	1 1-Apr-12
CR74	8003	SG37	16-Nov-ll
CR62	B007	SA9	7-03-12

Viewing

clientNo	propertyNo	ViewDate	Comment
CR56	PA14	24-May-13	too small
CR76	PG4	20-Apr-13	too remote
CR56	PG4	26-May-13	
CR62	PA14	14-May-13	no dining room
CR56	PG36	28-Apr-13	

Types of fragmentation- Horizontal

Consists of a subset of the tuples of a relation.

Defined using ***Selection*** operation of relational algebra:

$$\sigma_p(R)$$

For example:

PropertyForRent

PropertyNo	Street	City	Postcode	Type	Rooms	Rent	OwnerNo	StaffNo	BranchNo
PA14	16 Holhead	Aberdeen	AB7 5SU	House	6	650	C046	SA9	B007
PL94	6 Argyll St	London	NW2	Flat	4	400	C087	SL41	B005
PG4	6 Lawrence St	Glasgow	G11 9QX	Flat	3	350	C040		B003
PG36	2 Manor Rd	Glasgow	G32 4QX	Flat	3	375	C093	SG37	B003
PG21	18 Dale Rd	Glasgow	G12	House	5	600	C087	SG37	B003
PG16	5 Novar Dr	Glasgow	G12 9AX	Flat	4	450	C093	SG14	B003

Types of fragmentation- Horizontal

Consists of a subset of the tuples of a relation.

Defined using ***Selection*** operation of relational algebra:

$$\sigma_p(R)$$

For example:

$P_1 = \sigma_{type='House'}(PropertyForRent)$

$P_2 = \sigma_{type='Flat'}(PropertyForRent)$

PropertyForRent

PropertyNo	Street	City	Postcode	Type	Rooms	Rent	OwnerNo	StaffNo	BranchNo
PA14	16 Holhead	Aberdeen	AB7 5SU	House	6	650	C046	SA9	B007
PL94	6 Argyll St	London	NW2	Flat	4	400	C087	SL41	B005
PG4	6 Lawrence St	Glasgow	G11 9QX	Flat	3	350	C040		B003
PG36	2 Manor Rd	Glasgow	G32 4QX	Flat	3	375	C093	SG37	B003
PG21	18 Dale Rd	Glasgow	G12	House	5	600	C087	SG37	B003
PG16	5 Novar Dr	Glasgow	G12 9AX	Flat	4	450	C093	SG14	B003

PropertyNo	Street	City	Postcode	Type	Rooms	Rent	OwnerNo	StaffNo	BranchNo
PA14	16 Holhead	Aberdeen	AB7 5SU	House	6	650	C046	SA9	B007
PG21	18 Dale Rd	Glasgow	G12	House	5	600	C087	SG37	B003

Fragment P_1

PropertyNo	Street	City	Postcode	Type	Rooms	Rent	OwnerNo	StaffNo	BranchNo
PL94	6 Argyll St	London	NW2	Flat	4	400	C087	SL41	B005
PG4	6 Lawrence St	Glasgow	G11 9QX	Flat	3	350	C040		B003
PG36	2 Manor Rd	Glasgow	G32 4QX	Flat	3	375	C093	SG37	B003
PG16	5 Novar Dr	Glasgow	G12 9AX	Flat	4	450	C093	SG14	B003

Fragment P_2

The choice of horizontal fragmentation strategy

1. Access frequency: An examination of the predicates (or search conditions) used by transactions or queries in the applications.
2. Minterm selectivity: The strategy involves finding a **set of minimal** (that is, complete and relevant) predicates that can be used as the basis for the fragmentation schema.
    i. A set of predicates is **complete** if and only if any two tuples in the same fragment are referenced with the same probability by any transaction.
    ii. A predicate is **relevant** if there is at least one transaction that accesses the resulting fragments differently.

Example

EMP

ENO	ENAME	TITLE
E1	J. Doe	Elect. Eng.
E2	M. Smith	Syst. Anal.
E3	A. Lee	Mech. Eng.
E4	J. Miller	Programmer
E5	B. Casey	Syst. Anal.
E6	L. Chu	Elect. Eng.
E7	R. Davis	Mech. Eng.
E8	J. Jones	Syst. Anal.

ASG

ENO	PNO	RESP	DUR
E1	P1	Manager	12
E2	P1	Analyst	24
E2	P2	Analyst	6
E3	P3	Consultant	10
E3	P4	Engineer	48
E4	P2	Programmer	18
E5	P2	Manager	24
E6	P4	Manager	48
E7	P3	Engineer	36
E8	P3	Manager	40

PROJ

PNO	PNAME	BUDGET	LOC
P1	Instrumentation	150000	Montreal
P2	Database Develop.	135000	New York
P3	CAD/CAM	250000	New York
P4	Maintenance	310000	Paris

PAY

TITLE	SAL
Elect. Eng.	40000
Syst. Anal.	34000
Mech. Eng.	27000
Programmer	24000

The following are some of the possible simple predicates that can be defined on PAY.

$p1$: TITLE = “Elect. Eng.”

$p2$: TITLE = “Syst. Anal.”

$p3$: TITLE = “Mech. Eng.”

$p4$: TITLE = “Programmer”

$p5$: SAL $\leq$ 30000

The following are some of the minterm predicates that can be defined based on these simple predicates.

m1 : TITLE = “Elect. Eng.” $\wedge$ SAL $\leq$ 30000

m2 : TITLE = “Elect. Eng.” $\wedge$ SAL > 30000

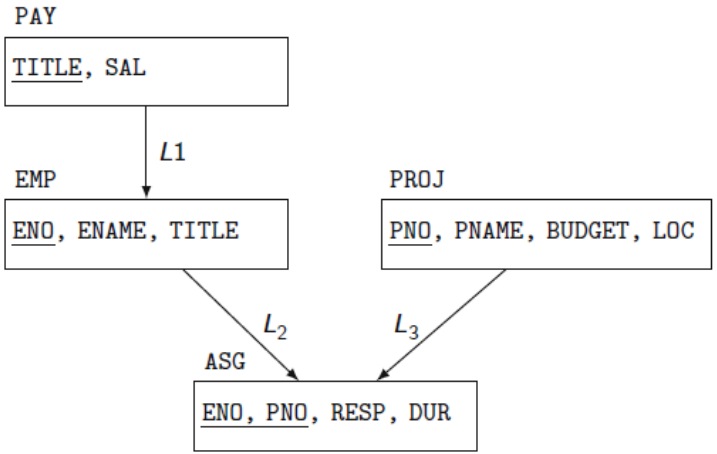
m3 : $\neg$(TITLE = “Elect. Eng.”) $\wedge$ SAL $\leq$ 30000

m4 : $\neg$(TITLE = “Elect. Eng.”) $\wedge$ SAL > 30000

m5 : TITLE = “Programmer” $\wedge$ SAL $\leq$ 30000

m6 : TITLE = “Programmer” $\wedge$ SAL > 30000

Example



Join graph representing relationships among relations

Horizontal fragmentation applies to the relations that have no incoming edges in the join graph and performed using the predicates that are defined on that relation.

- ✓ relations PAY and PROJ are subject to primary horizontal fragmentation, and
- ✓ EMP and ASG are subject to derived horizontal fragmentation.

Horizontal fragments based on the project location.

PROJ1 = $\sigma_{\text{LOC="Montreal"}}$ (PROJ)

PROJ2 = $\sigma_{\text{LOC="New York"}}$ (PROJ)

PROJ3 = $\sigma_{\text{LOC="Paris"}}$ (PROJ)

Predicates: {LOC = "Montreal", LOC = "New York", LOC = "Paris"}

Q1: Find the names and budgets of projects given their location? (issued at three sites)

Q2: Find the projects where the budget is less than or equal to $200,000? issued at two sites

To make the set of predicates complete,

Predicates: {LOC = "Montreal", LOC = "New York", LOC = "Paris", BUDGET $\leq$ 200000, BUDGET > 200000}

Types of fragmentation- Derived

Instead of using primary horizontal fragmentation , a designer may decide to fragment a table according to the way that another table is fragmented.

Applies to the target relations in the join graph and is performed based on predicates defined over the source relation of the join graph edge.

Given a child relation R and parent S, the derived fragmentation of R is defined as:

$$R_i = R \lhd_F S_i,\ 1 \leq i \leq w$$

where w is the number of horizontal fragments defined on S and f is the join attribute.

For example:

$S_3 = \sigma_{branchNo='B003'}(Staff)$

$S_4 = \sigma_{branchNo='B005'}(Staff)$

$S_5 = \sigma_{branchNo='B007'}(Staff)$

Could use derived fragmentation for Property:

$P_i =$ PropertyForRent $\lhd_{branchNo} S_i$,

$3 \leq i \leq 5$

Fragment P_3

PropertyNo	Street	City	Postcode	Type	Rooms	Rent	OwnerNo	StaffNo	BranchNo
PG4	6 Lawrence St	Glasgow	G11 9QX	Flat	3	350	CO40	SG14	B003
PG36	2 Manor Rd	Glasgow	G32 4QX	Flat	3	375	CO93	SG37	B003
PG21	18 Dale Rd	Glasgow	G12	House	5	600	CO87	SG37	B003
PG16	5 Novar Dr	Glasgow	G12 9AX	Flat	4	450	CO93	SG14	B003

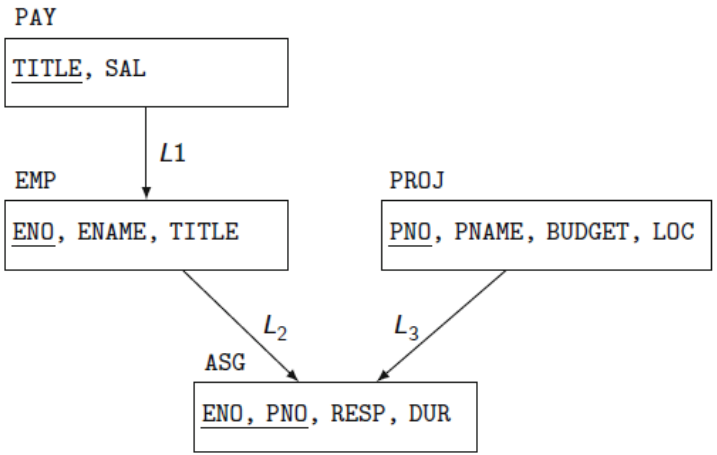
Fragment P_4

PropertyNo	Street	City	Postcode	Type	Rooms	Rent	OwnerNo	StaffNo	BranchNo
PL94	6 Argyll St	London	NW2	Flat	4	400	CO87	SL41	B005

Fragment P_5

PropertyNo	Street	City	Postcode	Type	Rooms	Rent	OwnerNo	StaffNo	BranchNo
PA14	16 Holhead	Aberdeen	AB7 5SU	House	6	650	CO46	SA9	B007

Example



Join graph representing relationships among relations

Consider edge L1, where source(L1) = PAY and target (L1) = EMP.

Then, we can group engineers into two groups according to their salary: those making less than or equal to $30,000, and those making more than $30,000.

EMP1 = EMP ◁ PAY1

EMP2 = EMP ◁ PAY2

where

$$PAY1 = \sigma_{SAL \leq 30000}(PAY)$$

$$PAY2 = \sigma_{SAL > 30000}(PAY)$$

EMP_1

ENO	ENAME	TITLE
E3	A. Lee	Mech. Eng.
E4	J. Miller	Programmer
E7	R. Davis	Mech. Eng.

EMP_2

ENO	ENAME	TITLE
E1	J. Doe	Elect. Eng.
E2	M. Smith	Syst. Anal.
E5	B. Casey	Syst. Anal.
E6	L. Chu	Elect. Eng.
E8	J. Jones	Syst. Anal.

Types of fragmentation- Vertical

Consists of a subset of attributes of a relation.

Defined using ***Projection*** operation of relational algebra:

$$\prod_{a1, \ldots, an}(R)$$

For example:

Staff

StaffNo	fName	lName	Position	Sex	DOB	Salary	BranchNo
SL21	John	White	Manager	M	1-Oct-45	30000	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24-Mar-58	18000	B003
SA9	Mary	Howe	Assistant	F	19-Feb-70	9000	B007
SG5	Susan	Brand	Manager	F	3-Jun-40	24000	B003
SL41	Julie	Lee	Assistant	F	13-Jun-65	9000	B005

Types of fragmentation- Vertical

Consists of a subset of attributes of a relation.

Defined using ***Projection*** operation of relational algebra:

$$\prod_{a1, \ldots, an}(R)$$

For example:

$$S_1 = \prod_{staffNo,\ position,\ sex,\ DOB,\ salary}(Staff)$$

$$S_2 = \prod_{staffNo,\ fName,\ lName,\ branchNo}(Staff)$$

Staff

StaffNo	fName	lName	Position	Sex	DOB	Salary	BranchNo
SL21	John	White	Manager	M	1-Oct-45	30000	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24-Mar-58	18000	B003
SA9	Mary	Howe	Assistant	F	19-Feb-70	9000	B007
SG5	Susan	Brand	Manager	F	3-Jun-40	24000	B003
SL41	Julie	Lee	Assistant	F	13-Jun-65	9000	B005

Fragment S_1

StaffNo	Position	Sex	DOB	Salary
SL21	Manager	M	1-Oct-45	30000
SG37	Assistant	F	10-Nov-60	12000
SG14	Supervisor	M	24-Mar-58	18000
SA9	Assistant	F	19-Feb-70	9000
SG5	Manager	F	3-Jun-40	24000
SL41	Assistant	F	13-Jun-65	9000

Fragment S_2

StaffNo	fName	lName	BranchNo
SL21	John	White	B005
SG37	Ann	Beech	B003
SG14	David	Ford	B003
SA9	Mary	Howe	B007
SG5	Susan	Brand	B003
SL41	Julie	Lee	B005

The choice of Vertical fragmentation strategy

1. **Grouping**:

   Starts by creating as many vertical fragments as possible and then incrementally reducing the number of fragments by merging the fragments together. See [Hammer79] and [Sacca85].

2. **Splitting**:

   Starts with a relation and decides on beneficial partitioning based on the access behavior of applications to the attributes. See [Hammer79] and [Navathe84]

   Vertical fragments can be determined by establishing the **affinity** of one attribute to another.

	a_1	a_2	a_3	a_4
a_1		1	0	1
a_2	1		0	1
a_3	0	0		0
a_4	1	1	0	

The affinity of columns expresses the extent to which they are used together in processing.

- ✓ 1s represent an access involving the corresponding attribute pair and are eventually replaced by numbers representing the transaction frequency.
- ✓ Pairs with high affinity should appear in the same vertical fragment; pairs with low affinity may be separated.

Example

EMP

ENO	ENAME	TITLE
E1	J. Doe	Elect. Eng.
E2	M. Smith	Syst. Anal.
E3	A. Lee	Mech. Eng.
E4	J. Miller	Programmer
E5	B. Casey	Syst. Anal.
E6	L. Chu	Elect. Eng.
E7	R. Davis	Mech. Eng.
E8	J. Jones	Syst. Anal.

ASG

ENO	PNO	RESP	DUR
E1	P1	Manager	12
E2	P1	Analyst	24
E2	P2	Analyst	6
E3	P3	Consultant	10
E3	P4	Engineer	48
E4	P2	Programmer	18
E5	P2	Manager	24
E6	P4	Manager	48
E7	P3	Engineer	36
E8	P3	Manager	40

PROJ

PNO	PNAME	BUDGET	LOC
P1	Instrumentation	150000	Montreal
P2	Database Develop.	135000	New York
P3	CAD/CAM	250000	New York
P4	Maintenance	310000	Paris

PAY

TITLE	SAL
Elect. Eng.	40000
Syst. Anal.	34000
Mech. Eng.	27000
Programmer	24000

Assume that the following queries are defined to run on PROJ relation.

AP1 q1: SELECT BUDGET FROM PROJ WHERE PNO=Value;

AP2 q2: SELECT PNAME, BUDGET FROM PROJ;

AP3 q3: SELECT PNAME FROM PROJ WHERE LOC=Value;

AP4 q4: SELECT SUM(BUDGET) FROM PROJ WHERE LOC=Value;

	PNO	PNAME	BUDGET	LOC
q1	1	0	1	0
q2	0	1	1	0
q3	0	1	0	1
q4	0	0	1	1

Example attribute usage matrix

Example

The attribute affinity measure between two attributes Ai and Aj of a relation R(A1, A2, . . . , An) with respect to the set of queries Q={q1, q2,..qm},

$$aff(A_i, A_j) = \sum_{k|use(qk,Ai)=1 \wedge use(qk,Aj)=1|} \sum_{\forall Sl} refl(qk)accl(qk)$$

Where refl(qk)- the # of accesses to attributes (Ai , Aj) for each execution of application qk at site Sl

accl(qk) - the application access frequency measure previously defined and modified to include frequencies at different sites.

For simplicity, let us assume that refl(qk) = 1 for all qk and Sl .

If the application frequencies are

acc1(q1) = 15 acc1(q2) = 5
acc1(q3) = 25 acc1(q4) = 3
acc2(q1) = 20 acc2(q2) = 0
acc2(q3) = 25 acc3(q4) = 0
acc3(q1) = 10 acc3(q2) = 0
acc3(q3) = 25 acc2(q4) = 0

	S1	S2	S3
q1	15	20	10
q2	5	0	0
q3	25	25	25
q4	3	0	0

Then the **affinity measure** between attributes PNO and BUDGET can be measured as

$aff(PNO, BUDGET) = \sum_{k=1 \text{ to } 1} \sum_{S=1 \text{ to } 3} accl(qk) = acc1(q1) + acc2(q1) + acc3(q1) = 45$

	PNO	PNAME	BUDGET	LOC
PNO		0	45	0
PNAME	0		5	75
BUDGET	45	5		3
LOC	0	75	3	

Types of fragmentation- Mixed/Hybrid

Consists of a horizontal fragment that is vertically fragmented, or a vertical fragment that is horizontally fragmented.

Defined using *Selection* and *Projection* operations of relational algebra:

$$\sigma_{p}(\prod_{a1, \ldots, an}(R)) \quad \text{or} \quad \prod_{a1, \ldots, an}(\sigma_{p}(R))$$

For example:

$S_1 = \prod_{staffNo,\ position,\ sex,\ DOB,\ salary}(Staff)$

$S_2 = \prod_{staffNo,\ fName,\ lName,\ branchNo}(Staff)$

$S_{21} = \sigma_{branchNo='B003'}(S_2)$

$S_{22} = \sigma_{branchNo='B005'}(S_2)$

$S_{23} = \sigma_{branchNo='B007'}(S_2)$

Fragment S_2

StaffNo	fName	lName	BranchNo
SL21	John	White	B005
SG37	Ann	Beech	B003
SG14	David	Ford	B003
SA9	Mary	Howe	B007
SG5	Susan	Brand	B003
SL41	Julie	Lee	B005

Fragment S_{21}

StaffNo	fName	lName	BranchNo
SG37	Ann	Beech	B003
SG14	David	Ford	B003
SG5	Susan	Brand	B003

Fragment S_{22}

StaffNo	fName	lName	BranchNo
SL21	John	White	B005
SL41	Julie	Lee	B005

Fragment S_{23}

StaffNo	fName	lName	BranchNo
SA9	Mary	Howe	B007

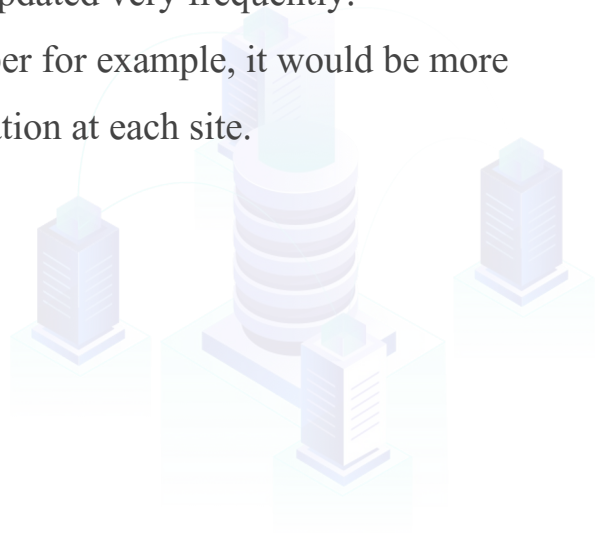
No fragmentation

The Branch relation contains only a small number of tuples and is not updated very frequently.

Rather than trying to horizontally fragment the relation on branch number for example, it would be more sensible to leave the relation whole and simply replicate the Branch relation at each site.

Branch

BranchNo	Street	City	Postcode
B005	22 Deer Rd	London	SW1 4EH
B007	16 Argyll St	Aberdeen	AB2 3SU
B003	163 Main St	Glasgow	G11 9QX
B004	32 Manse Rd	Bristol	BS99 1NZ
B002	56 Clover Dr	London	NW10 6EU



Summary

Distributed Database Design: Fragmentation, Allocation and Replication

Various type of data allocation: Partitioned/ Fragmented, Fully replicated, Partially replicated, Nonreplicated/Non-fragmented.

Comparison of Strategies for Data Allocation based on five important factors.

The important of fragmentation and the way of doing fragmentation.