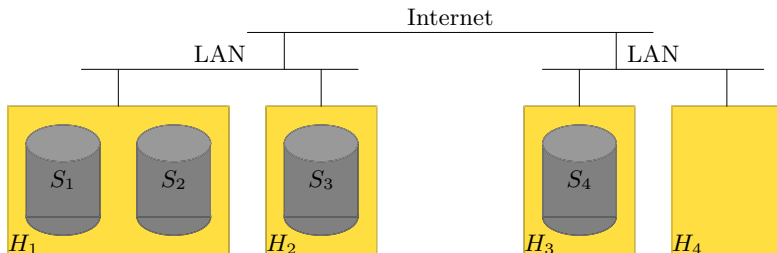


Distributed Databases

P.J. McBrien

Imperial College London

Distributed Databases

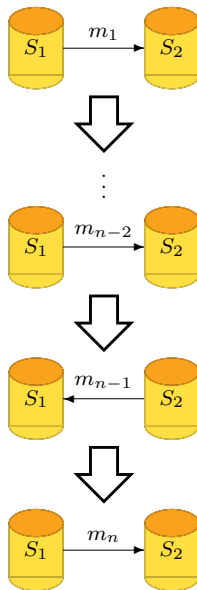


distributed database (DDB)

several databases at different **sites**, connected by

- Server racks
- LAN
- Internet

Coordination Failure with Communication Failure

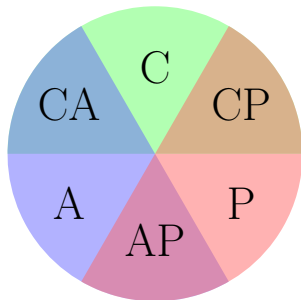


CAP Theorem

CAP Theorem

No distributed system may maintain all three of

- **Consistency:** all nodes see the same version of data
- **Availability:** the system always responds within fixed upper limits of time
- **Partition Tolerance:** the system always gives correct responses even when messages are lost or network failures occur



- CA
e.g. Centralised Database
- CP
e.g. Distributed RDBMS
- AP
e.g. DNS

Distributed Databases: Two Approaches

Heterogeneous DDB

Sites

- vary in database technology
- managed by different DBAs
- designed at different times

Example Architectures

- Data Warehouse
- Mediator

Example Technologies

- ODBC
- JDBC

Homogeneous DDB

Sites

- use same database technology
- managed by one DBA
- designed at same time

Example Architectures

- RDBMS Clusters
- Map Reduce

Example Technologies

- MySQL Cluster
- Hadoop

Distributed Heterogeneous Databases: Physical Differences

Database 1

branch		
sortcode	bname	cash
55-66-56	'Wimbledon'	94340.45
55-66-34	'Goodge St'	8900.67
55-66-67	'Strand'	34005.00

Database 2

```
sortcode,bname,cash
"55-66-56","Wimbledon",94340.45
"55-66-34","Goodge St",8900.67
"55-66-67","Strand",34005.00
```

RDBMS

Query using SQL

CSV file

Query by reading file

- Need **common data model (CDM)** *e.g.* Relational, ER

Distributed Heterogeneous Databases: Semantic Differences

Database 1

branch		
<u>sortcode</u>	bname	cash
56	'Wimbledon'	94340.45
34	'Goodge St'	8900.67
67	'Strand'	34005.00

Database 2

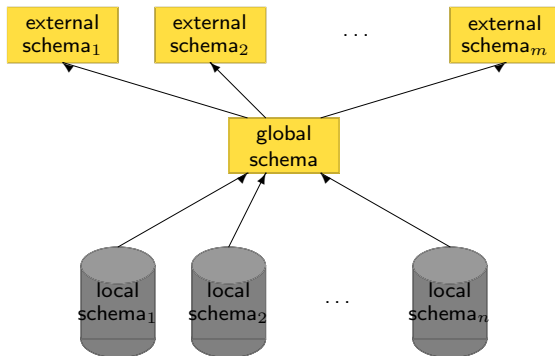
branch		
<u>sortcode</u>	branchname	cash
55-66-56	'Wimbledon'	94340.45
55-66-34	'Goodge St'	8900.67
55-66-67	'Strand'	34005.00

assume 55-66- sortcode
call branches bname

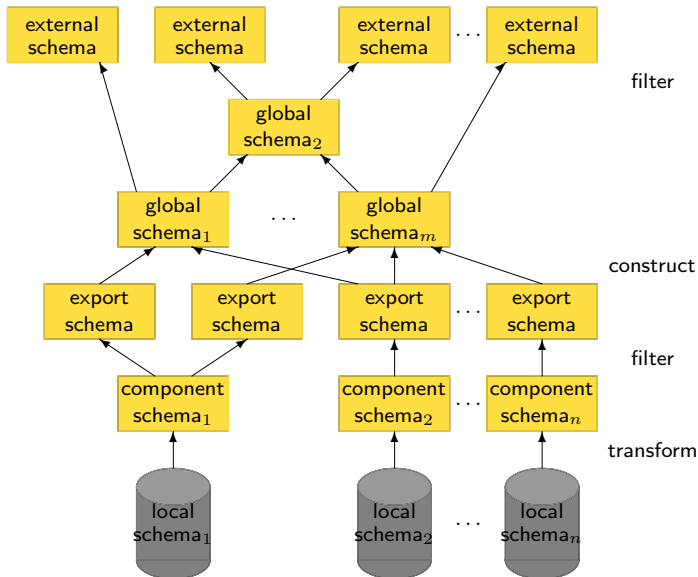
full sortcode
call branches branchname

- Need to perform **schema integration**

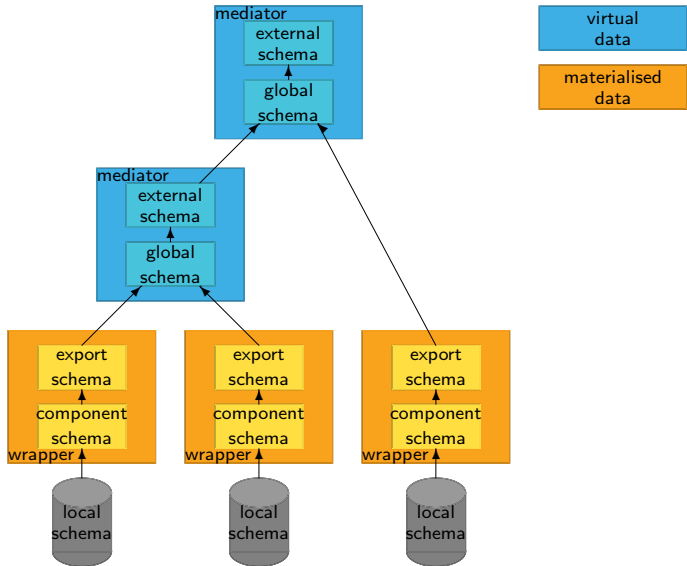
Logical Model for Homogeneous DDB



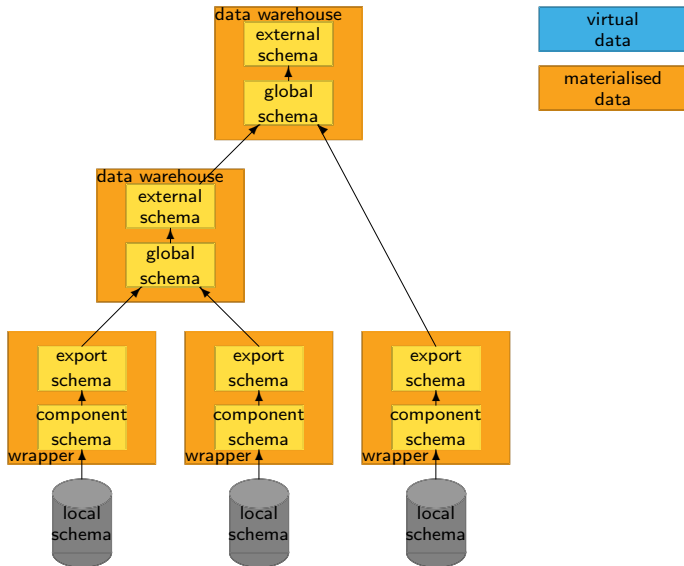
Logical Model for Heterogeneous Databases



Operational Model: Mediator Architecture



Operational Model: Data Warehouses



Topics to Cover in DDB

Operational Aspects

- How might data and hence queries be distributed between local schemas?
- How might a query on the global schema be optimised to run over the local schemas?
- How are ACID properties of transactions maintained in DDB?

Logical Aspects

- How are export schemas integrated to form a global schema?

Data Distribution

branch		
<u>sortcode</u>	bname	cash
56	'Wimbledon'	94340.45
34	'Goodge St'	8900.67
67	'Strand'	34005.00

movement			
<u>mid</u>	no	amount	tdate
1000	100	2300.00	5/1/1999
1001	101	4000.00	5/1/1999
1002	100	-223.45	8/1/1999
1004	107	-100.00	11/1/1999
1005	103	145.50	12/1/1999
1006	100	10.23	15/1/1999
1007	107	345.56	15/1/1999
1008	101	1230.00	15/1/1999
1009	119	5600.00	18/1/1999

account				
<u>no</u>	type	cname	rate?	sortcode
100	'current'	'McBrien, P.'	NULL	67
101	'deposit'	'McBrien, P.'	5.25	67
103	'current'	'Boyd, M.'	NULL	34
107	'current'	'Poulovassilis, A.'	NULL	56
119	'deposit'	'Poulovassilis, A.'	5.50	56
125	'current'	'Bailey, J.'	NULL	56

key branch(sortcode)

key branch(bname)

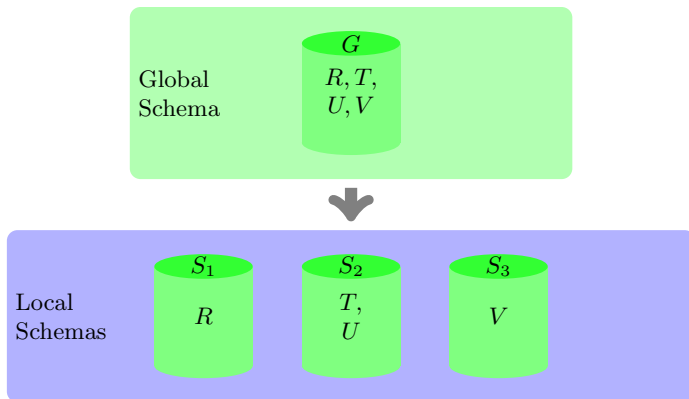
key movement(mid)

key account(no)

movement(no) $\xRightarrow{fk}$ account(no)

account(sortcode) $\xRightarrow{fk}$ branch(sortcode)

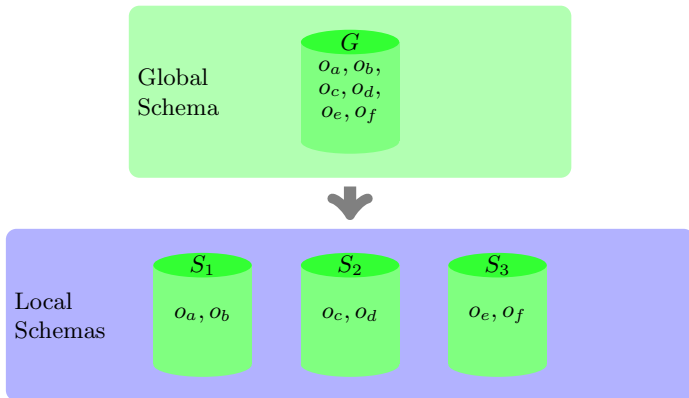
Data Distribution: Remote Tables



remote tables

- split tables $R, T, \dots$ between sites $S_1, S_2, \dots$
- reads r and writes w must correspondingly distributed
e.g. $r[R], r[T], r[U] \rightarrow r[R]$ on S_1 and $r[T], r[U]$ on S_2

Data Distribution: Fragmentation



fragmentation

- split data objects $o_a, o_b, \dots$ of relation R between sites
- queries and updates must be correspondingly distributed
 $r[o_a], r[o_b], r[o_c] \rightarrow r[o_a], r[o_b]$ on S_1 and $r[o_c]$ on S_2

RDB: Horizontal Fragmentation

Global
Schema

account				
no	type	cname	rate?	sortcode
100	'current'	'McBrien, P.'	NULL	67
101	'deposit'	'McBrien, P.'	5.25	67
103	'current'	'Boyd, M.'	NULL	34
107	'current'	'Poulovassilis, A.'	NULL	56
119	'deposit'	'Poulovassilis, A.'	5.50	56
125	'current'	'Bailey, J.'	NULL	56



Local
Schemas

account ₁				
no	type	cname	rate?	sortcode
100	'current'	'McBrien, P.'	NULL	67
103	'current'	'Boyd, M.'	NULL	34
107	'current'	'Poulovassilis, A.'	NULL	56
125	'current'	'Bailey, J.'	NULL	56

account ₂				
no	type	cname	rate?	sortcode
101	'deposit'	'McBrien, P.'	5.25	67
119	'deposit'	'Poulovassilis, A.'	5.50	56

■ $\text{account}_1 = \sigma_{\text{type}=\text{'current'}} \text{account}$

■ $\text{account}_2 = \sigma_{\text{type} \neq \text{'current'}} \text{account}$

Quiz 1: Speed and Reliability of Horizontal Fragmentation

account				
no	type	cname	rate?	sortcode
100	'current'	'McBrien, P.'	NULL	67
101	'deposit'	'McBrien, P.'	5.25	67
103	'current'	'Boyd, M.'	NULL	34
107	'current'	'Poulovassilis, A.'	NULL	56
119	'deposit'	'Poulovassilis, A.'	5.50	56
125	'current'	'Bailey, J.'	NULL	56

```
SELECT no
FROM account
WHERE cname LIKE 'McBrien%'
```

If account is horizontally fragmented on no, compared with a single site version, is the DDB

A

Faster
Less Reliable

B

Faster
More Reliable

C

Slower
Less Reliable

D

Slower
More Reliable

Quiz 2: Horizontal fragmentation attributes

account				
<u>no</u>	type	cname	rate?	sortcode
100	'current'	'McBrien, P.'	NULL	67
101	'deposit'	'McBrien, P.'	5.25	67
103	'current'	'Boyd, M.'	NULL	34
107	'current'	'Poulovassilis, A.'	NULL	56
119	'deposit'	'Poulovassilis, A.'	5.50	56
125	'current'	'Bailey, J.'	NULL	56

Which is the worst attribute to fragment on?

A

no

B

type

C

cname

D

rate

RDB: Derived Horizontal Fragmentation

movement			
<u>mid</u>	no	amount	tdate
1000	100	2300.00	5/1/1999
1002	100	-223.45	8/1/1999
1005	103	145.50	12/1/1999
1006	100	10.23	15/1/1999
1004	107	-100.00	11/1/1999
1007	107	345.56	15/1/1999

 S_1

movement			
<u>mid</u>	no	amount	tdate
1001	101	4000.00	5/1/1999
1008	101	1230.00	15/1/1999
1009	119	5600.00	18/1/1999

 S_2

$$\text{movement}_i = \text{movement} \bowtie \text{account}_i$$

Derived Relational Algebra: Semi Join $\ltimes$

Semi Join

$$R \ltimes T = R \bowtie \pi_{Attr(R) \cap Attr(T)} T$$

Semi Join

account $\ltimes$ movement = account $\bowtie$ π_{no} movement

account $\ltimes$ movement				
<u>no</u>	type	cname	rate?	sortcode
100	'current'	'McBrien, P.'	NULL	67
101	'deposit'	'McBrien, P.'	5.25	67
103	'current'	'Boyd, M.'	NULL	34
107	'current'	'Poulovassilis, A.'	NULL	56
119	'deposit'	'Poulovassilis, A.'	5.50	56

RDB: Vertical Fragmentation

account			
<u>no</u>	type	rate?	sortcode
100	'current'	NULL	67
101	'deposit'	5.25	67
103	'current'	NULL	34
107	'current'	NULL	56
119	'deposit'	5.50	56
125	'current'	NULL	56

S_1

account	
<u>no</u>	cname
100	'McBrien, P.'
101	'McBrien, P.'
103	'Boyd, M.'
107	'Poulovassilis, A.'
119	'Poulovassilis, A.'
125	'Bailey, J.'

S_2

■ $\text{account}_1 = \pi_{\text{no}, \text{type}, \text{rate}, \text{sortcode}} \text{account}$

■ $\text{account}_2 = \pi_{\text{no}, \text{cname}} \text{account}$

Quiz 3: Vertical Fragmentation

branch		
<u>sortcode</u>	bname	cash
56	'Wimbledon'	94340.45
34	'Goodge St'	8900.67
67	'Strand'	34005.00

key(bname)
key(sortcode)

Which of the following is a correct vertical fragmentation of branch?

A

$\pi_{\text{bname, cash}}$ branch
 $\pi_{\text{sortcode, cash}}$ branch

B

$\pi_{\text{bname, sortcode}}$ branch
 π_{cash} branch

C

$\pi_{\text{bname, cash}}$ branch
 π_{sortcode} branch

D

$\pi_{\text{bname, cash}}$ branch
 $\pi_{\text{sortcode, bname}}$ branch

RA Definitions

Horizontal Fragmentation

$$R = R_1 \cup \dots \cup R_n$$

- ‘plain’ horizontal fragmentation splits rows using σ

$$R_1 = \sigma_{P_1} R, \quad \dots, \quad R_n = \sigma_{P_n} R$$

- derived horizontal fragmentation splits rows using $\bowtie$

$$R_1 = R \bowtie S_1, \quad \dots, \quad R_n = R \bowtie S_n$$

$$R \bowtie S = R \bowtie \pi_{R \cap S}(S)$$

Vertical Fragmentation

$$R = R_1 \bowtie \dots \bowtie R_n$$

- vertical fragmentation splits rows using π

$$R_1 = \pi_{attrs_1} R, \quad \dots, \quad R_n = \pi_{attrs_n} R$$

- A **loss-less join** decomposition

Worksheet: Fragmentation

account				
<u>no</u>	type	cname	rate?	sortcode
100	'current'	'McBrien, P.'	NULL	67
101	'deposit'	'McBrien, P.'	5.25	67
103	'current'	'Boyd, M.'	NULL	34

 S_1

account				
<u>no</u>	type	cname	rate?	sortcode
107	'current'	'Poulovassilis, A.'	NULL	56
119	'deposit'	'Poulovassilis, A.'	5.50	56
125	'current'	'Bailey, J.'	NULL	56

 S_2

Worksheet: Fragmentation

account		
<u>no</u>	type	sortcode
100	'current'	67
101	'deposit'	67
103	'current'	34
107	'current'	56
119	'deposit'	56
125	'current'	56

 S_1

account	
sortcode	cname
67	'McBrien, P.'
67	'McBrien, P.'
34	'Boyd, M.'
56	'Poulovassilis, A.'
56	'Poulovassilis, A.'
56	'Bailey, J.'

 S_2

RDB: Hybrid Fragmentation

The diagram illustrates four tables, S_1 , S_2 , S_3 , and S_4 , each containing account data. S_1 and S_2 are structured as 'account' tables with columns: no, type, rate?, and sortcode. S_3 and S_4 are also 'account' tables but with columns: no and cname. The data rows for each table are as follows:

Table	no	type	rate?	sortcode	cname
S_1	100	'current'	NULL	67	
	103	'current'	NULL	34	
	107	'current'	NULL	56	
	125	'current'	NULL	56	
S_2	101	'deposit'	5.25	67	
	119	'deposit'	5.50	56	
S_3	100				'McBrien, P.'
	103				'Boyd, M.'
	107				'Poulovassilis, A.'
	125				'Bailey, J.'
S_4	101				'McBrien, P.'
	119				'Poulovassilis, A.'

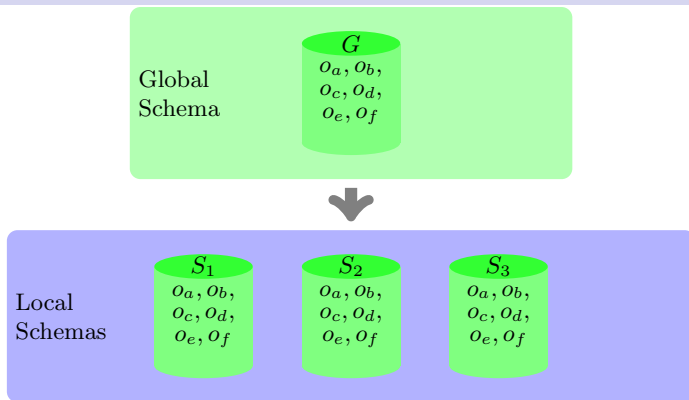
$$\text{account}_1 = \pi_{\text{no,type,rate,sortcode}}(\sigma_{\text{type}=\text{current}}\text{account})$$

$$\text{account}_2 = \pi_{\text{no,type,rate,sortcode}}(\sigma_{\text{type}\neq\text{current}}\text{account})$$

$$\text{account}_3 = \pi_{\text{no,cname}}(\sigma_{\text{type}=\text{current}}\text{account})$$

$$\text{account}_4 = \pi_{\text{no,cname}}(\sigma_{\text{type}\neq\text{current}}\text{account})$$

Data Distribution: Replication



replication

- copy data objects $o_a, o_b, \dots$ of R between sites
- queries may run on any site
 $r[o_a], r[o_b], r[o_c] \rightarrow r[o_a], r[o_b], r[o_c]$ on S_1 (or on S_2 or on S_3)
- updates write on all sites
 $r[o_a], w[o_a] \rightarrow r[o_a], w[o_a]$ on S_1 , and $w[o_a]$ on S_2 and S_3

Quiz 4: Speed and Reliability of Replication: Reads

account				
no	type	cname	rate?	sortcode
100	'current'	'McBrien, P.'	NULL	67
101	'deposit'	'McBrien, P.'	5.25	67
103	'current'	'Boyd, M.'	NULL	34
107	'current'	'Poulovassilis, A.'	NULL	56
119	'deposit'	'Poulovassilis, A.'	5.50	56
125	'current'	'Bailey, J.'	NULL	56

```
SELECT no
FROM account
WHERE cname LIKE 'McBrien%'
```

Considering queries like those above, if account is replicated on several sites, compared with a single site version, is the DDB

A

Faster
Less Reliable

B

Faster
More Reliable

C

Slower
Less Reliable

D

Slower
More Reliable

Quiz 5: Speed and Reliability of Replication: Writes

account				
no	type	cname	rate?	sortcode
100	'current'	'McBrien, P.'	NULL	67
101	'deposit'	'McBrien, P.'	5.25	67
103	'current'	'Boyd, M.'	NULL	34
107	'current'	'Poulovassilis, A.'	NULL	56
119	'deposit'	'Poulovassilis, A.'	5.50	56
125	'current'	'Bailey, J.'	NULL	56

```
UPDATE account
SET rate=2.0
WHERE type='deposit'
```

Considering queries like those above, if account is replicated on several sites, compared with a single site version, is the DDB

A

Faster
Less Reliable

B

Faster
More Reliable

C

Slower
Less Reliable

D

Slower
More Reliable

Data Distribution: Migration

