



Experience Report:

Exploiting Advanced Database Optimization Features for Large-Scale SAP R/3 Installations*

Bernhard Zeller

Alfons Kemper

University of Passau
<last name>@db.fmi.uni-passau.de

* This work was supported by an SAP contract within
the so-called Terabyte-Project



Outline

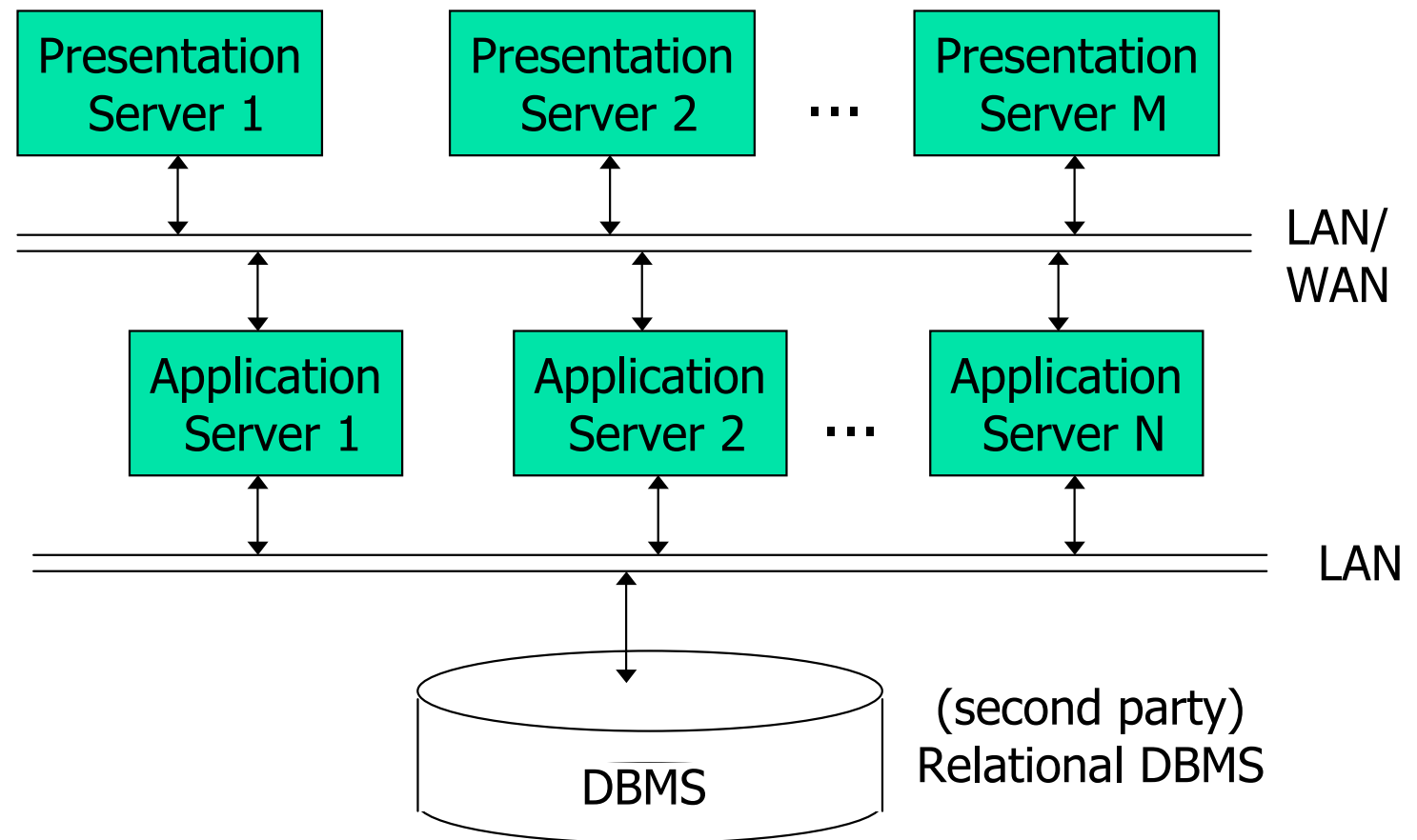
- Brief Overview of SAP R/3
- Motivation
- Related Work
- Traditional Performance Tuning Techniques
- Exploiting Horizontal Partitioning for Tuning Purposes
 - Partitioning Scenarios/Techniques and their Pros and Cons
 - Possible Benefits and Drawbacks of Partitioning
- Performance Analysis
- Conclusion



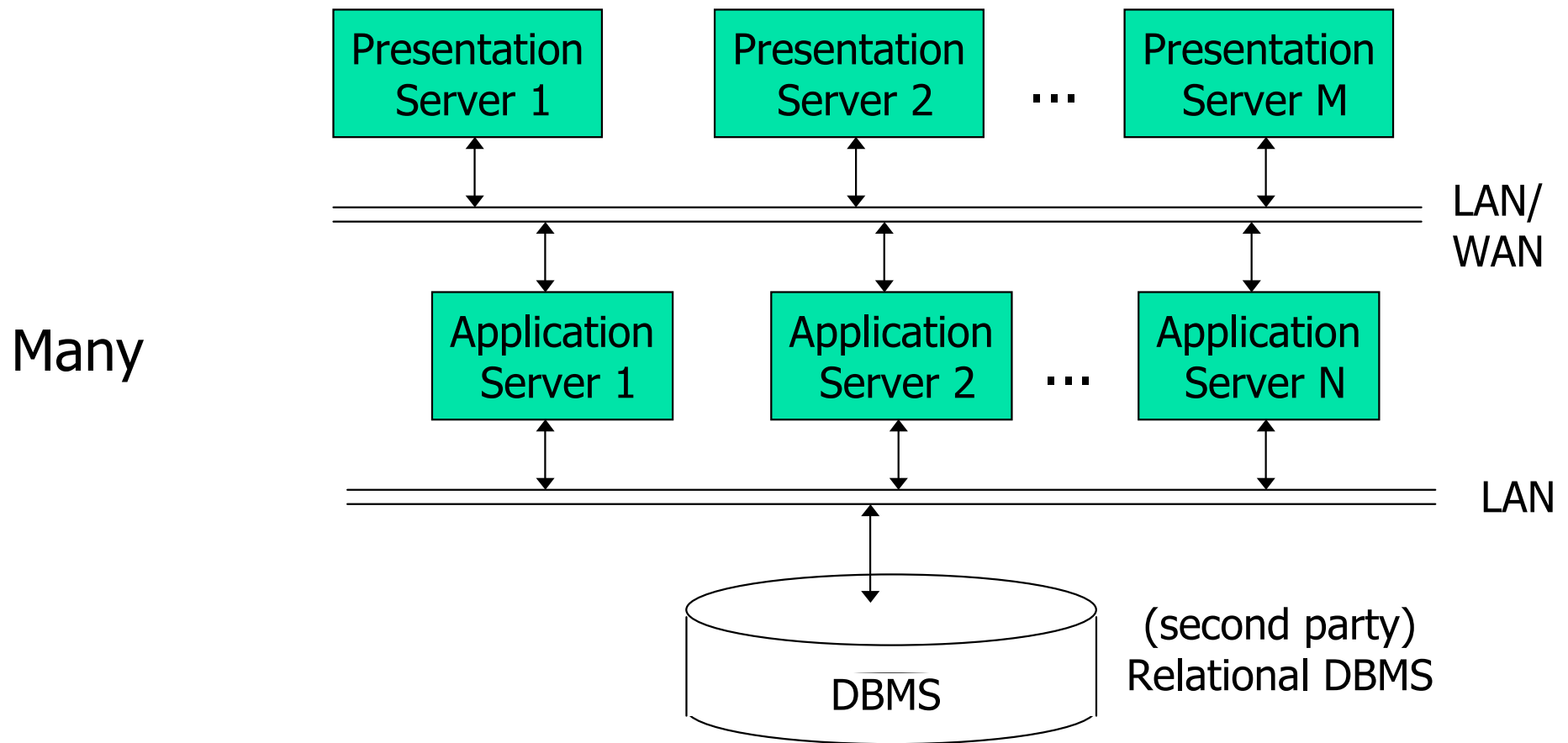
Overview of SAP R/3

- SAP is the market leader for integrated business solutions
- SAP R/3 is SAP's enterprise resource planning (ERP) product
- SAP R/3 provides modules for finance, human resources, material management, etc.
- today about 18.000 customers world wide use SAP R/3 (used by most Fortune 500 companies)
- more than 44.000 Installations world wide
- Three-Tier Client/Server-Architecture

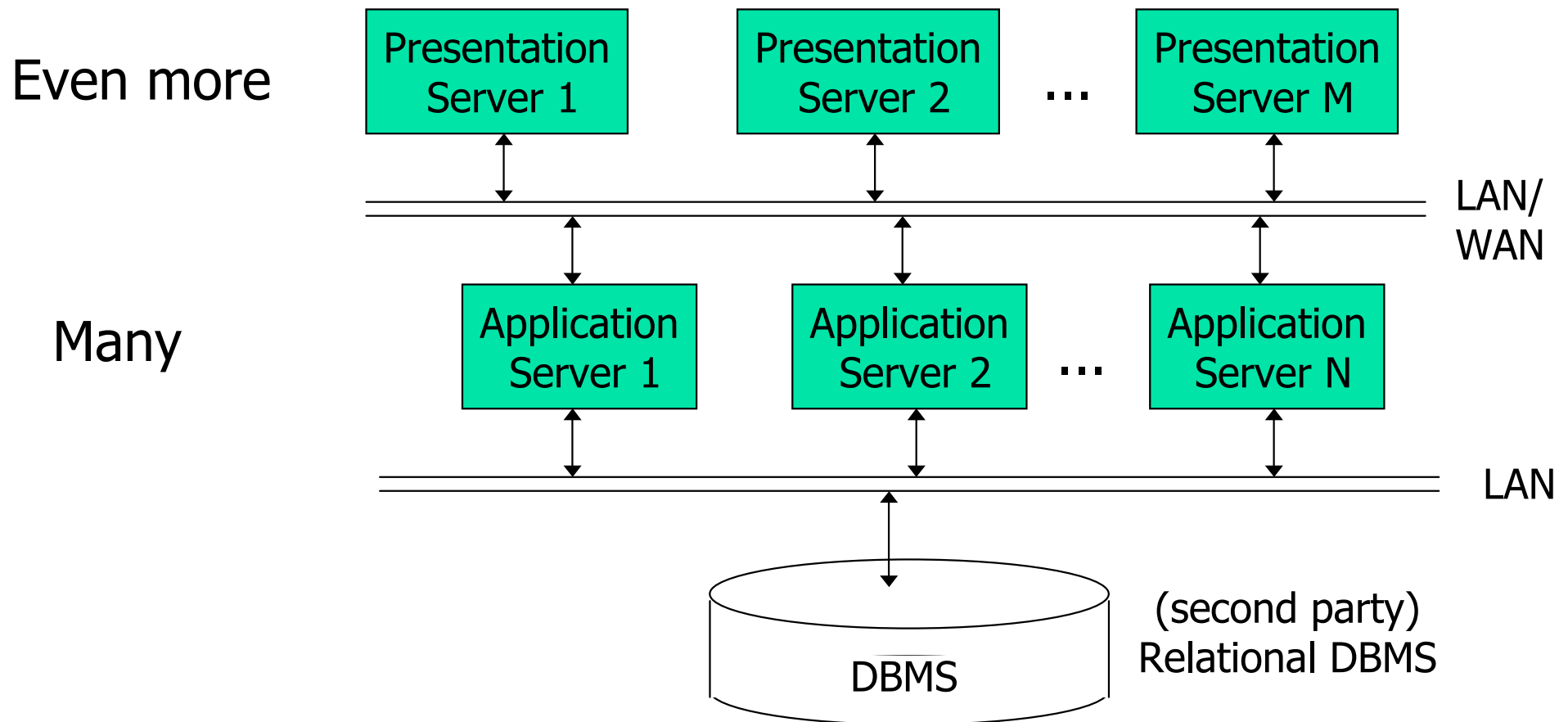
Three-Tier Client/Server- Architecture of SAP R/3



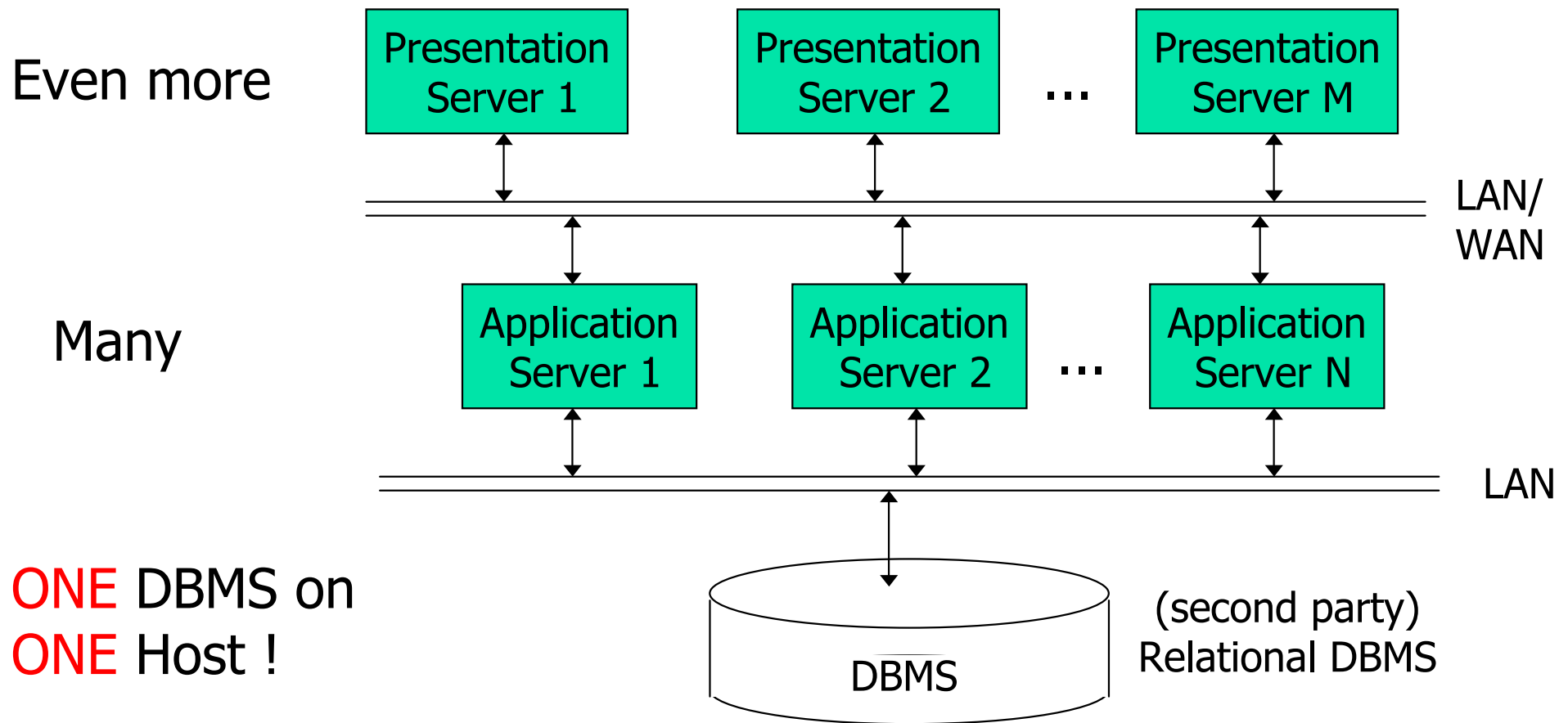
Three-Tier Client/Server- Architecture of SAP R/3



Three-Tier Client/Server- Architecture of SAP R/3



Three-Tier Client/Server- Architecture of SAP R/3





Motivation (1)

Today's **high end** SAP R/3 installations have reached their load capacity **limits**:

- data volumes of SAP R/3 System (i.e., the database volumes) are growing tremendously (several hundred Terabytes)
- hard to maintain (7 x 24)
- performance worsens



Motivation (1)

Today's **high end** SAP R/3 installations have reached their load capacity **limits**:

- data volumes of SAP R/3 System (i.e., the database volumes) are growing tremendously (several hundred Terabytes)
- hard to maintain (7 x 24)
- performance worsens

→ Exploiting advanced features like **horizontal partitioning** can widen these load capacity limits



Motivation (1)

Today's **high end** SAP R/3 installations have reached their load capacity **limits**:

- data volumes of SAP R/3 System (i.e., the database volumes) are growing tremendously (several hundred Terabytes)
 - hard to maintain (7 x 24)
 - performance worsens
- Exploiting advanced features like **horizontal partitioning** can widen these load capacity limits
- Already implemented by most database vendors:
No additional effort, just switch on.



Motivation (2)

High end systems are the most important systems:

- revenue
- **prestige** → new contracts, business competition

Therefore: every day business is the top priority, i.e.

- only tolerable slow down of important (=OLTP) transactions due to the use of new techniques
- if this can't be guaranteed: *don't do it !*



Motivation (2)

High end systems are the most important systems:

- revenue
- **prestige** → new contracts, business competition

Therefore: every day business is the top priority, i.e.

- only tolerable slow down of important (=OLTP) transactions due to the use of new techniques
- if this can't be guaranteed: *don't do it !*

In this case: The benefits are obvious. Prove that horizontal partitioning doesn't conflict with daily business !



Related Work

- G. Copeland, W. Alexander, E. Boughter, and T. Keller
Data placement in bubba.
In *Proc. of the ACM SIGMOD Conf. on Management of Data*, Chicago, IL, USA, 1988.
- D. J. DeWitt, R. H. Gerber, G. Graefe, M. L. Heytens,
K. B. Kumar, and M. Muralikrishna
Gamma - a high performance dataflow database machine.
In *Proc. Of the Conf. on Very Large Data Bases (VLDB)*, Kyoto, Japan, 1986.
- M. Mehta and D. J. DeWitt
Data placement in shared-nothing parallel database systems.
VLDB Journal, 6(1):53-72, 1997.
- S. Ceri, M. Negri, and G. Pelagatti
Horizontal data partitioning in database design.
In *Proc. of the ACM SIGMOD Conf. on Management of Data*, Orlando, USA, 1982.



What's Different

- shared nothing distributed databases:
 - use of local computing power,
i.e. #partitions \approx #CPUs , #disks
 - network between the nodes is the bottleneck
 - "centralized" systems like SAP R/3:
 - limited number of CPUs, main memory, disks,
i.e., #partitions \gg #CPUs, #disks
 - shared memory/disk access
- hazards on disk page / CPU level likely (thrashing)



Outline

- Brief Overview of SAP R/3
- Motivation
- Related Work
- Traditional Performance Tuning Techniques
- Exploiting Horizontal Partitioning for Tuning Purposes
 - Partitioning Scenarios/Techniques and their Pros and Cons
 - Possible Benefits and Drawbacks of Partitioning
- Performance Analysis
- Conclusion



Traditional Tuning Techniques

Reduce data (Archiving)

- sometimes not possible
-

Additional Indices

- even more data
 - huge → hard to maintain
 - slow down updates/inserts
-

Additional job instances

- limited number of CPU's
 - problems due to data skews
-

Additional/better hardware

- too expensive
 - already the newest HW installed
-

Special designed software

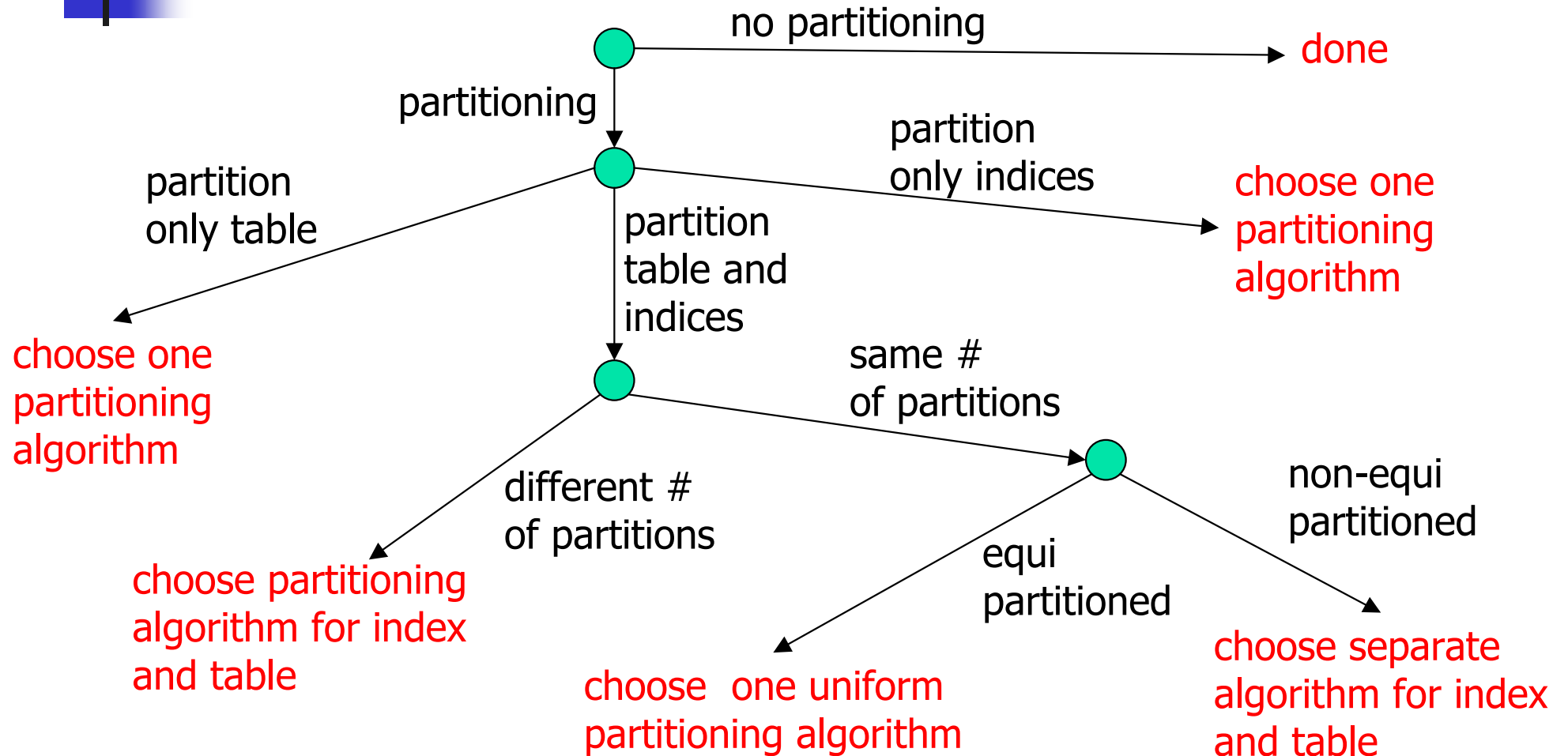
- really expensive
- long deployment times



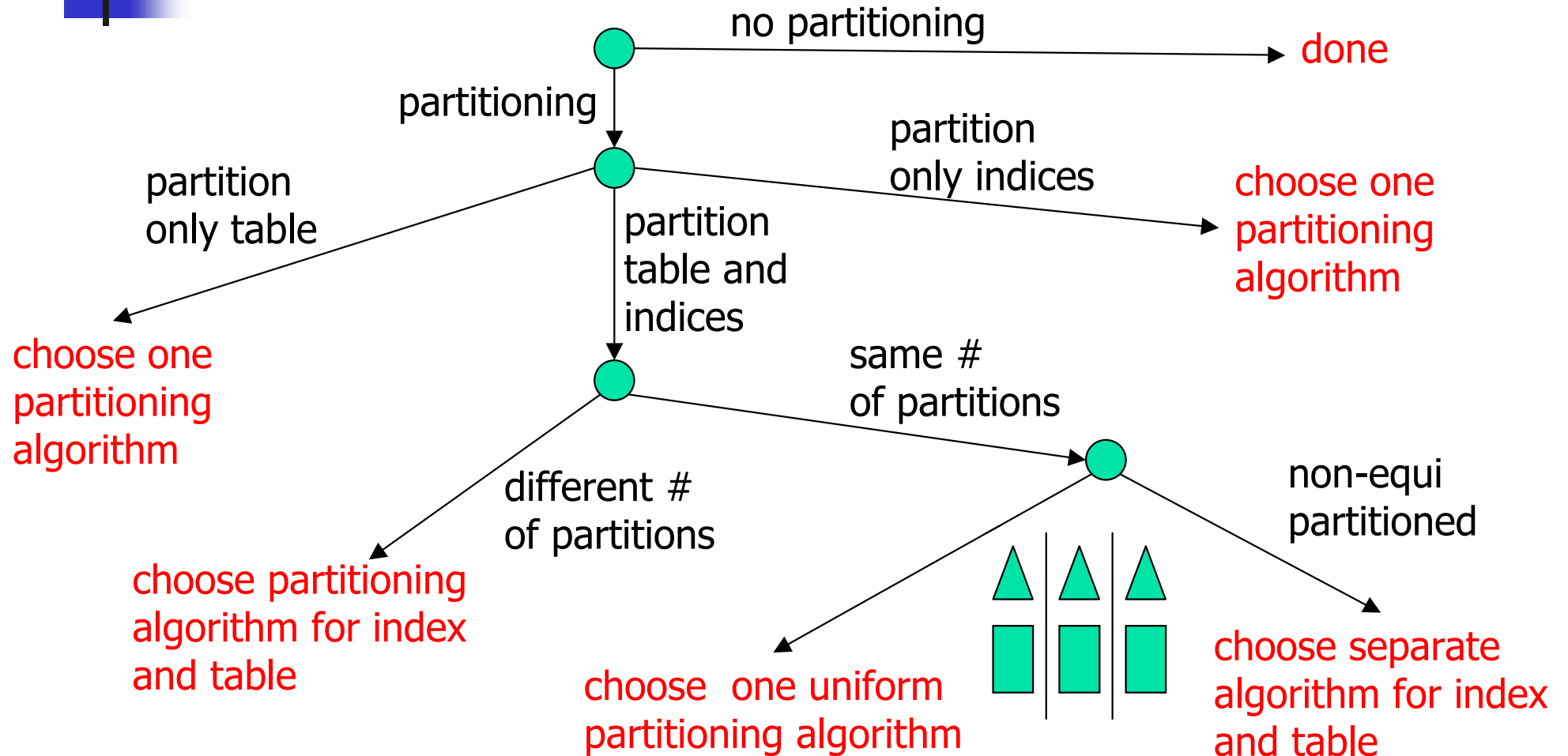
Partitioning Techniques and their Pros and Cons

- round robin, hash partitioning
 - + balanced partition sizes
 - in which partition will record R be stored ?
- range partitioning
 - + users have knowledge about the data distribution → can use this knowledge at application level (work load balancing, definition of working sets, ...)
 - unbalanced partition sizes due to data skews

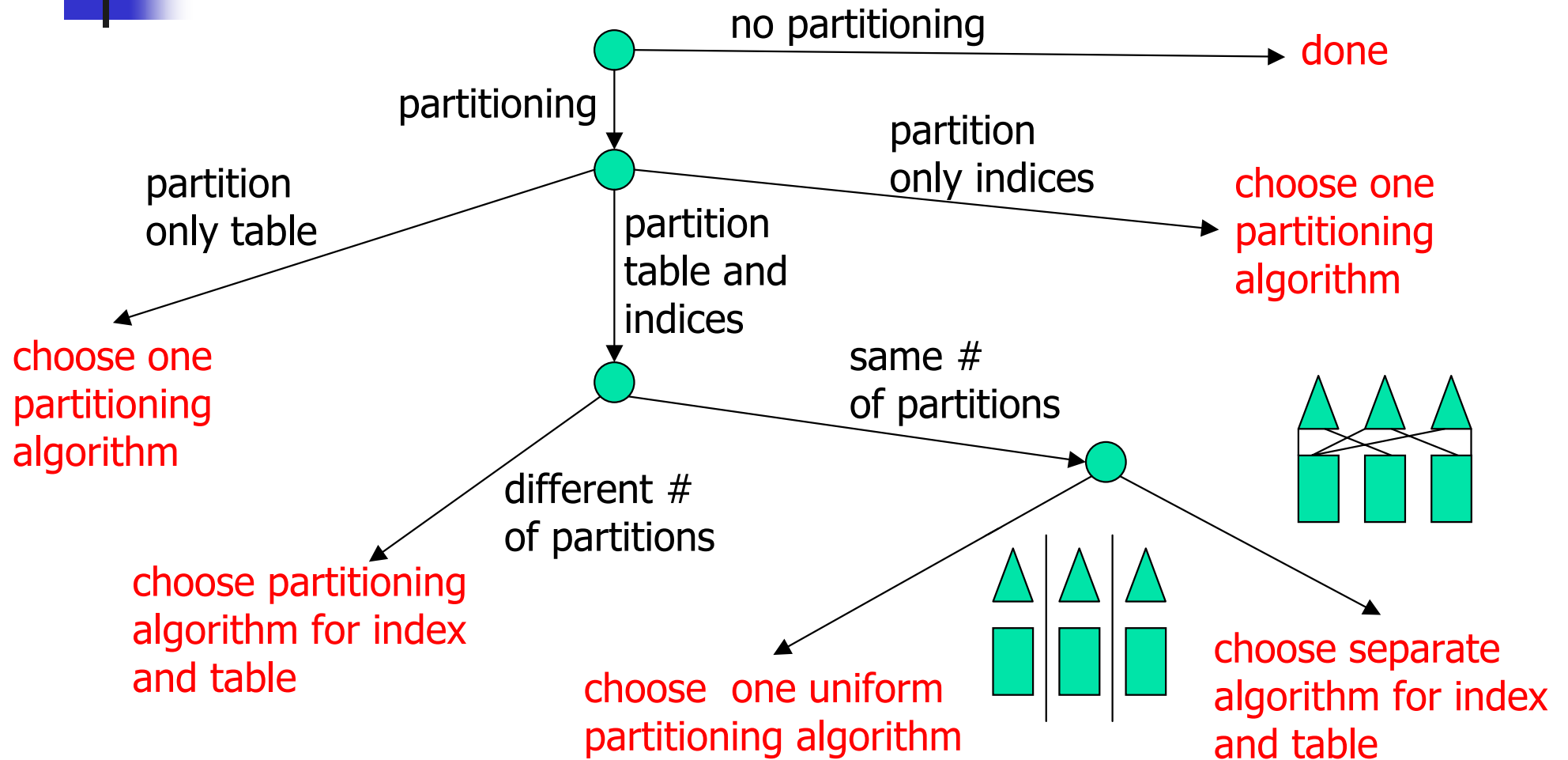
Partitioning Scenarios



Partitioning Scenarios



Partitioning Scenarios





Benefits of Partitioning

Keep data manageable by processing data partition wise:

- administrative tasks like index re-creation, gathering statistics, table re-organization, ...
- partition-wise, parallel processing at application level, e.g., partition by plant number and start an inventory job for each plant
- (equi) join processing (when partitioning fields are subset of join attributes)
- bulk deletes → drop whole partitions



Possible Drawbacks: Row Movement (RM)

- RM = movement of data from one partition to another because of an update
- doubles the cost of an update transaction
- produces additional logging and locking overhead



Possible Drawbacks: Row Movement (Example)

Table Equipment: Equipment of company COMP Inc.

Partition Plant LA

PlantID	EquID	Description
LA	006	Desk
LA	007	Chair de luxe
LA	008	Office copier

Partition Plant NY

PlantID	EquID	Description
NY	015	LCD Display
NY	017	Chair simple
NY	018	Mainframe

Task: New office copier for LA, old one to NY



Possible Drawbacks: Row Movement (Example)

Table Equipment: Equipment of company COMP Inc.

Partition Plant LA

PlantID	EquID	Description
LA	006	Desk
LA	007	Chair de luxe
LA	008	Office copier

Partition Plant NY

PlantID	EquID	Description
NY	015	LCD Display
NY	017	Chair simple
NY	018	Mainframe

DB: update equipment set PlantID = NY
where PlantID = LA and EquID = 008



Possible Drawbacks: Row Movement (Example)

Table Equipment: Equipment of company COMP Inc.

Partition Plant LA

PlantID	EquID	Description
LA	006	Desk
LA	007	Chair de luxe
LA	008	Office copier

Partition Plant NY

PlantID	EquID	Description
NY	015	LCD Display
NY	017	Chair simple
NY	018	Mainframe

DB Step 1: Delete in Partition "Plant LA"



Possible Drawbacks: Row Movement (Example)

Table Equipment: Equipment of company COMP Inc.

Partition Plant LA

PlantID	EquID	Description
LA	006	Desk
LA	007	Chair de luxe

Partition Plant NY

PlantID	EquID	Description
NY	015	LCD Display
NY	017	Chair simple
NY	018	Mainframe

DB Step 1: Delete in Partition "Plant LA"



Possible Drawbacks: Row Movement (Example)

Table Equipment: Equipment of company COMP Inc.

Partition Plant LA

PlantID	EquID	Description
LA	006	Desk
LA	007	Chair de luxe

Partition Plant NY

PlantID	EquID	Description
NY	015	LCD Display
NY	017	Chair simple
NY	018	Mainframe

DB Step 2: Insert into Partition "Plant NY"



Possible Drawbacks: Row Movement (Example)

Table Equipment: Equipment of company COMP Inc.

Partition Plant LA

PlantID	EquID	Description
LA	006	Desk
LA	007	Chair de luxe

Partition Plant NY

PlantID	EquID	Description
NY	015	LCD Display
NY	017	Chair simple
NY	018	Mainframe
NY	008	Office copier

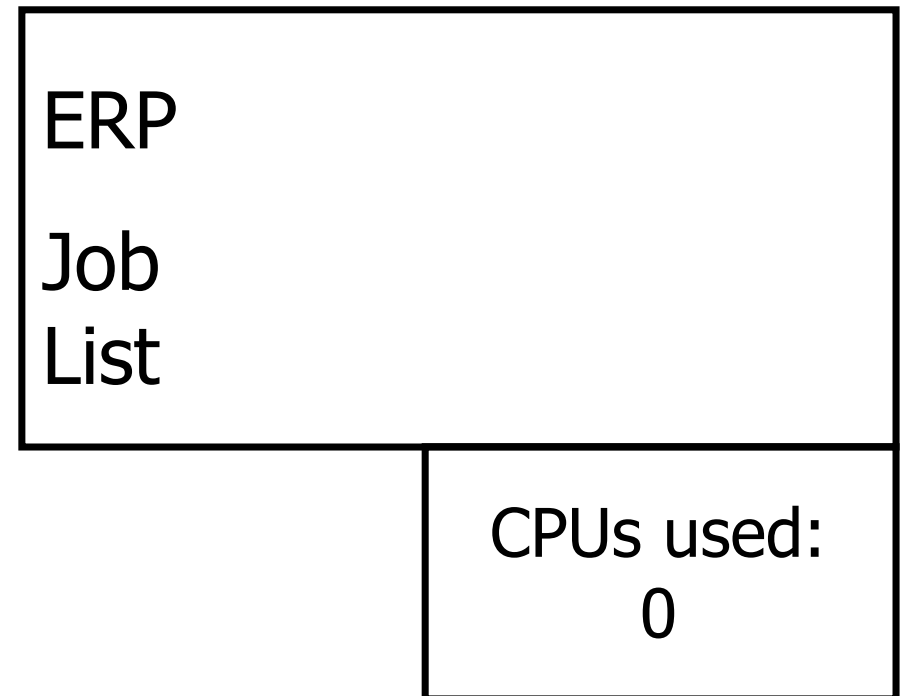
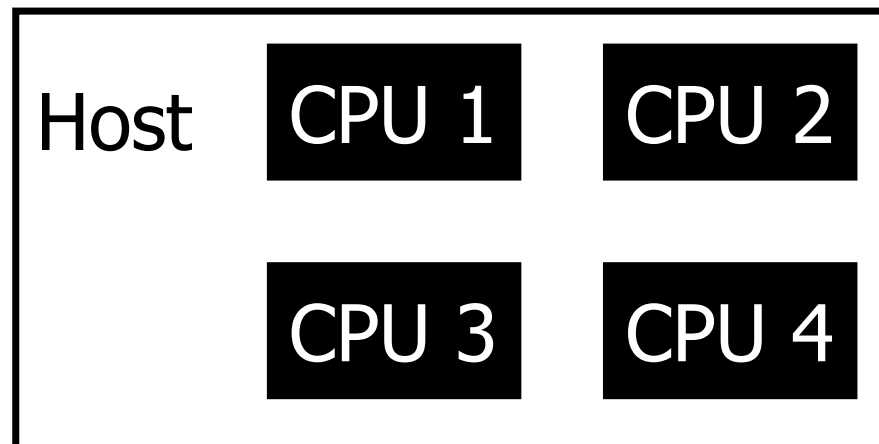
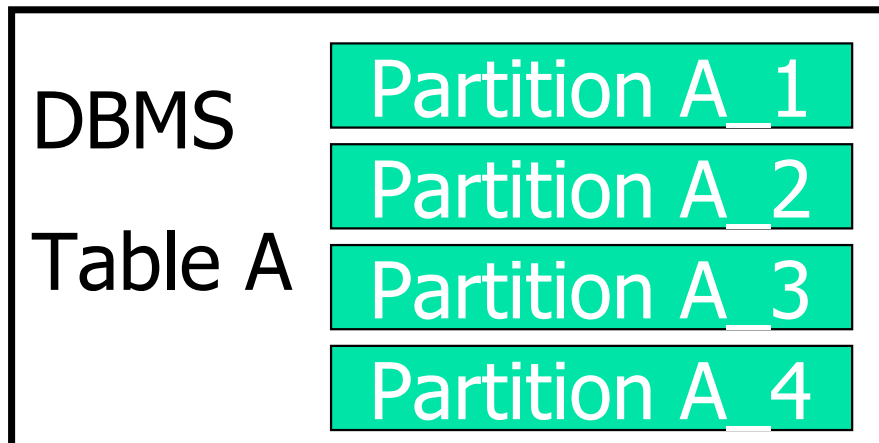
DB Step 2: Insert into Partition "Plant NY"



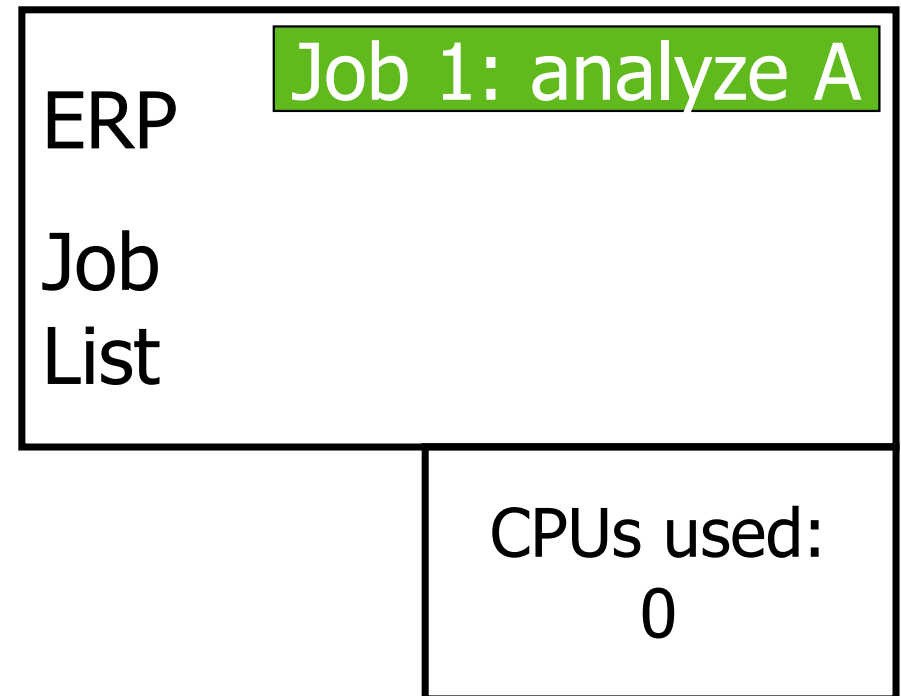
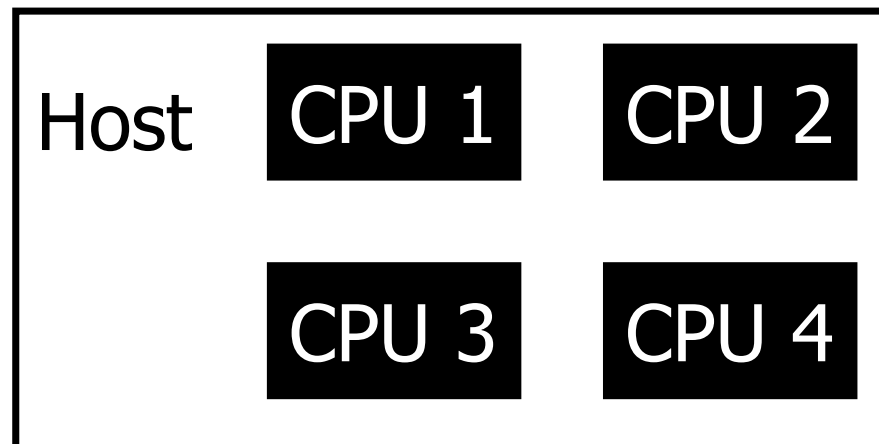
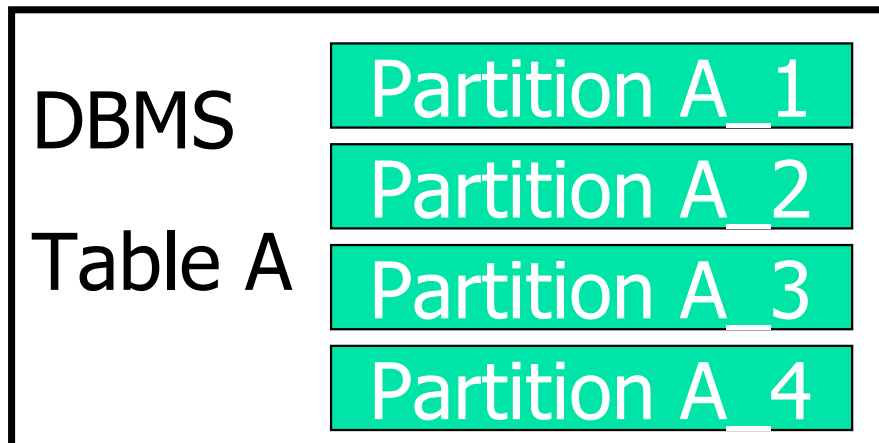
Possible Drawbacks: Conflicts with Parallel Jobs

- resources of ERP systems (CPU, memory, storage) managed at application level
 - ERP System has no knowledge of partitioning (i.e., parallelization) at database level
- Conflicts at CPU and disk page level are likely

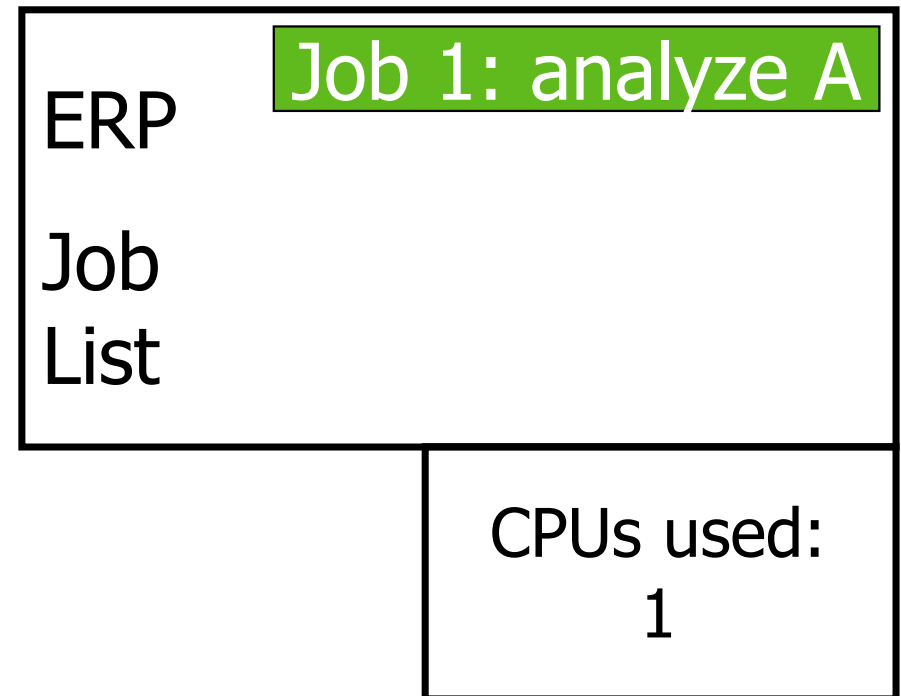
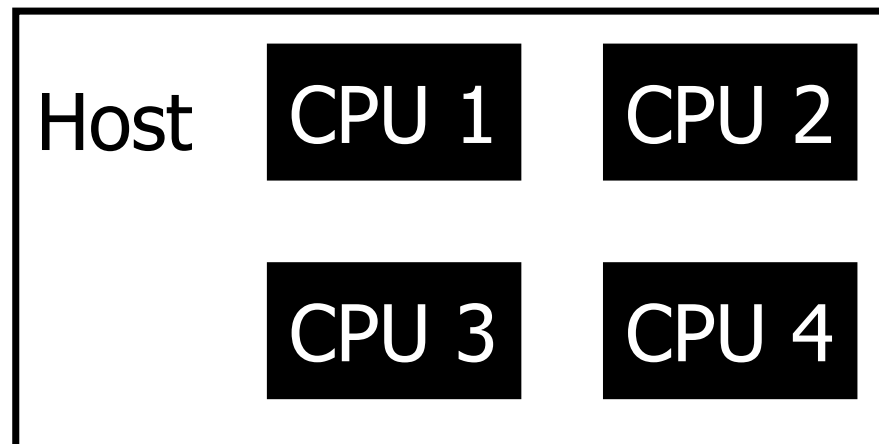
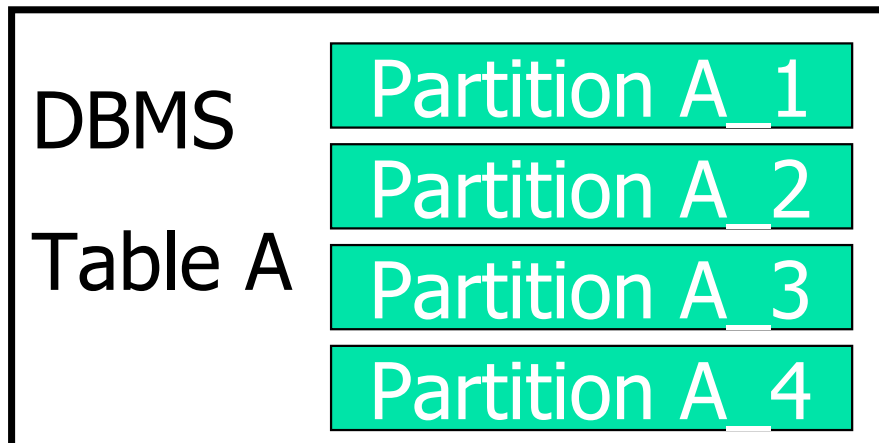
Possible Drawbacks: Conflicts with Parallel Jobs



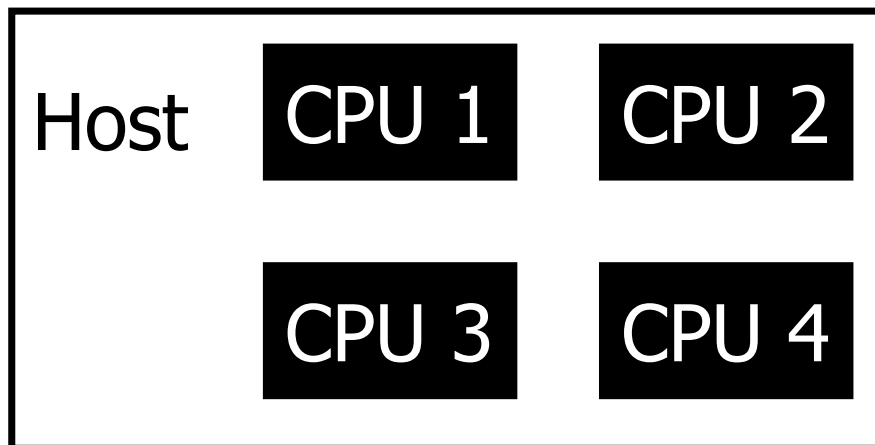
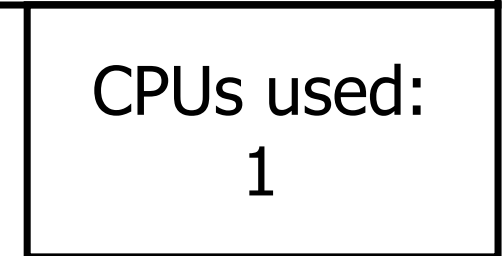
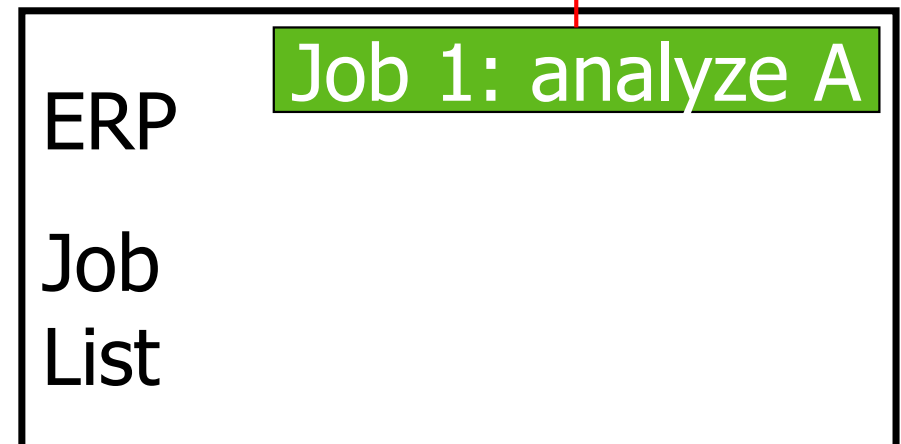
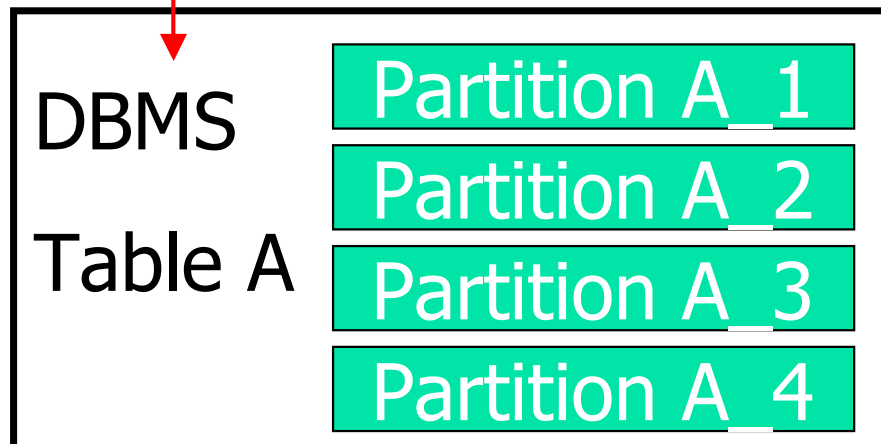
Possible Drawbacks: Conflicts with Parallel Jobs



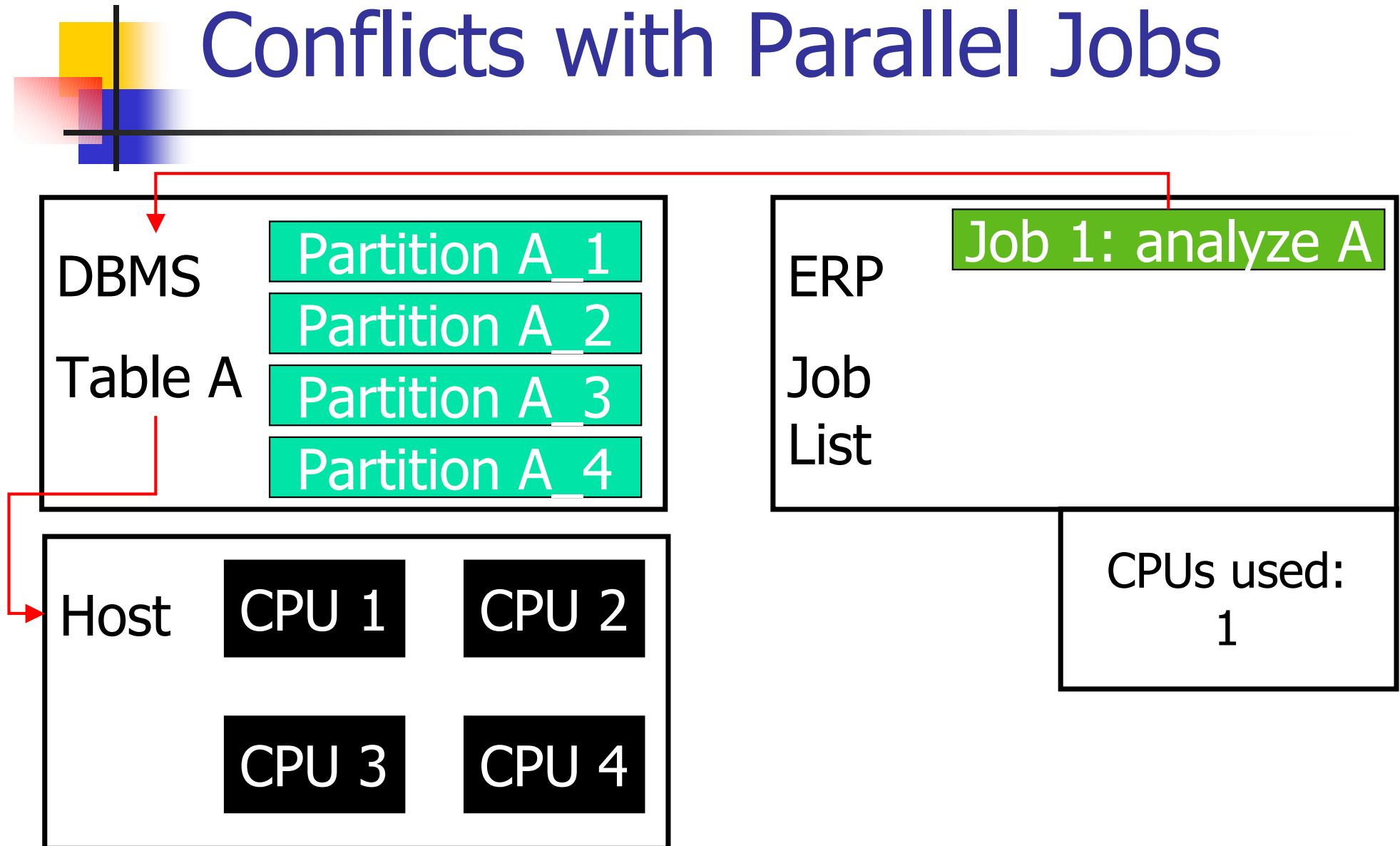
Possible Drawbacks: Conflicts with Parallel Jobs



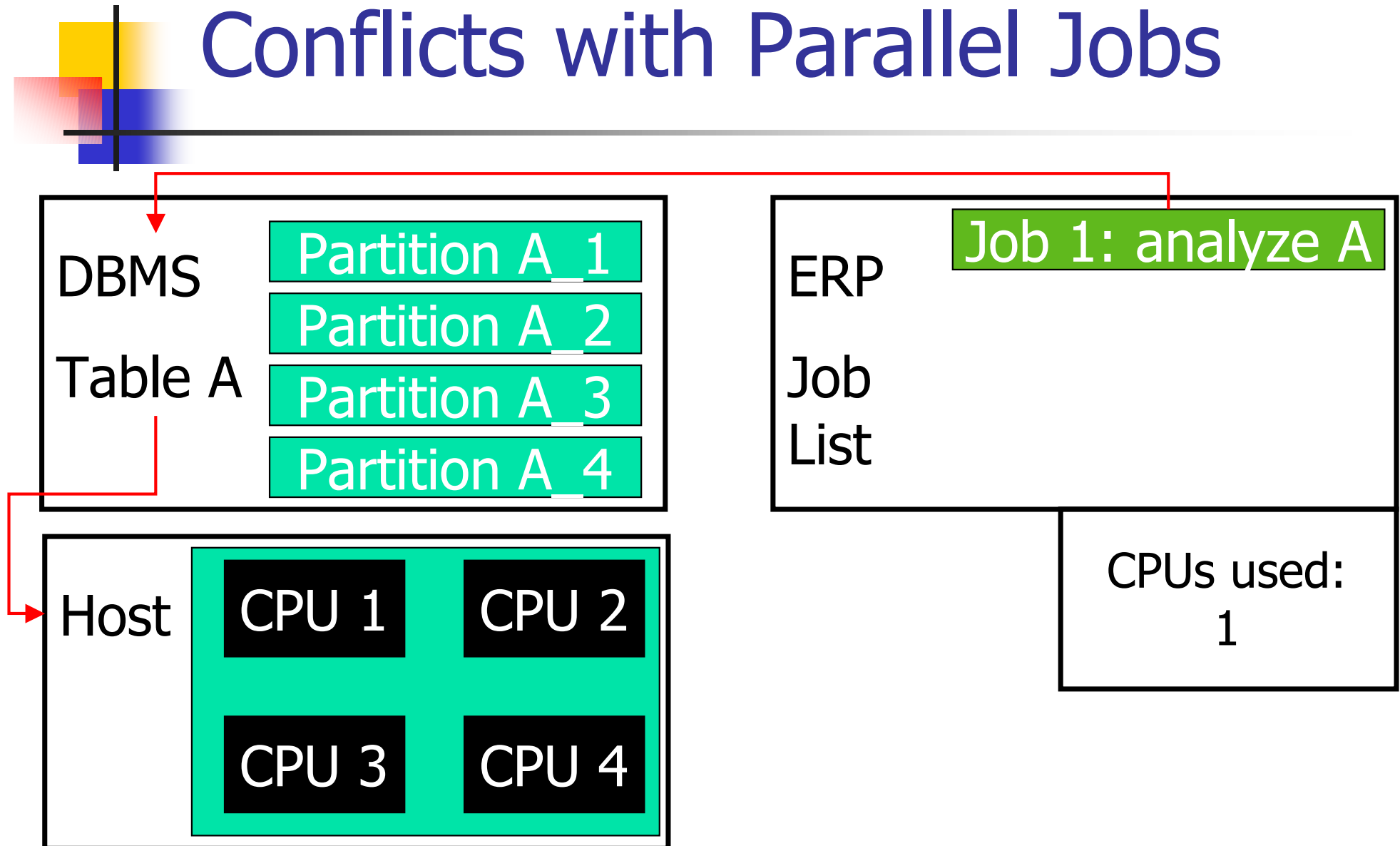
Possible Drawbacks: Conflicts with Parallel Jobs



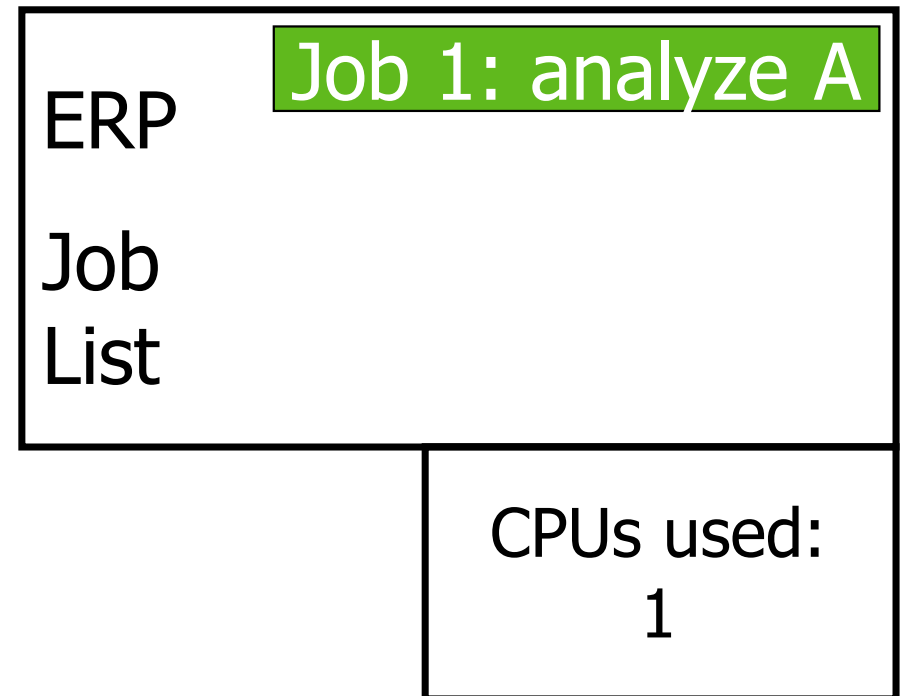
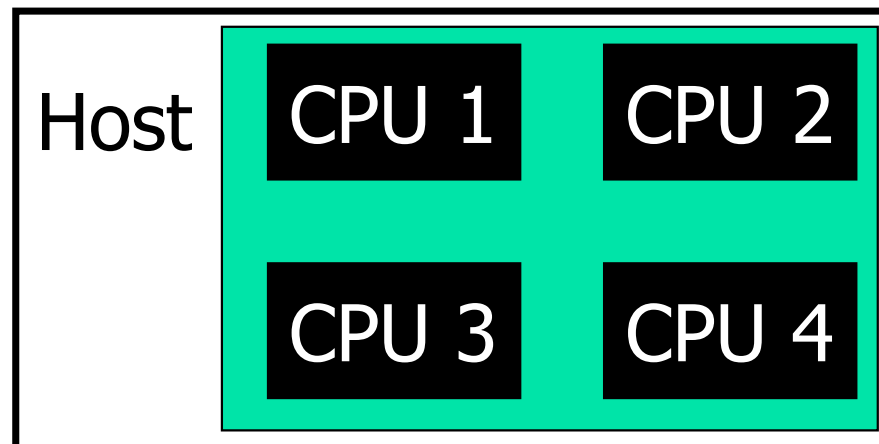
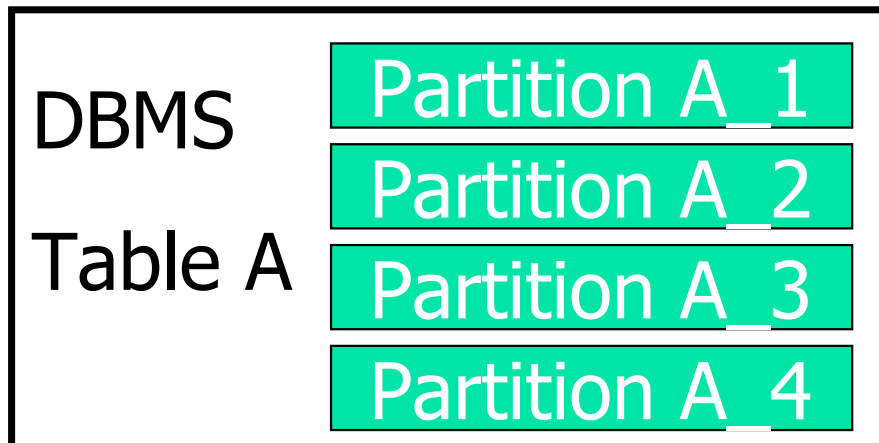
Possible Drawbacks: Conflicts with Parallel Jobs



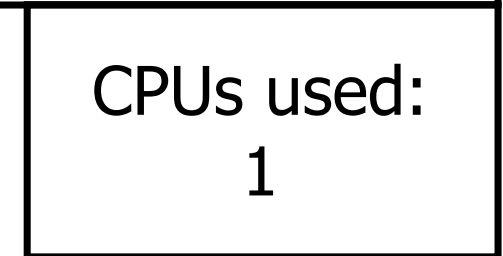
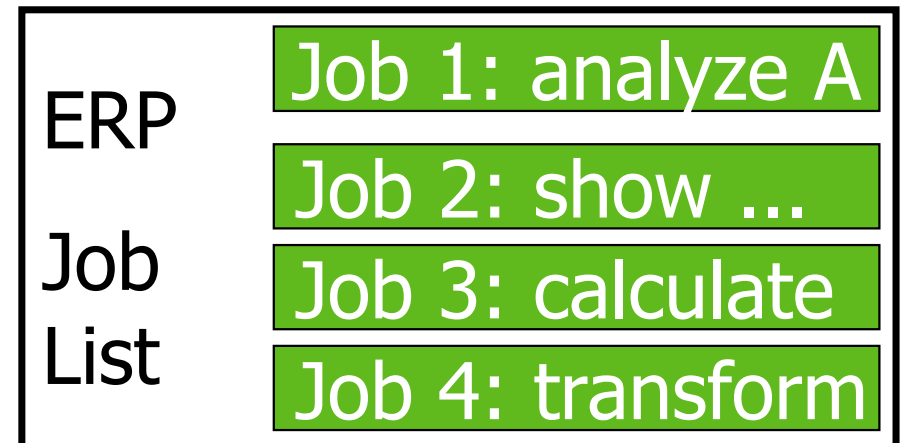
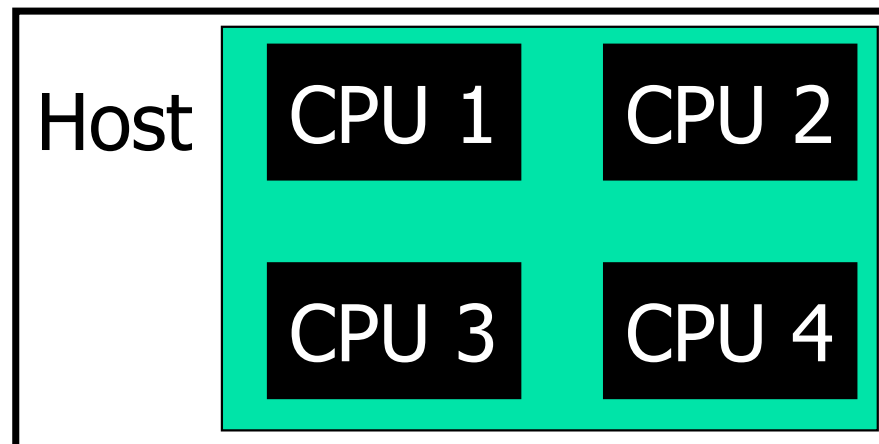
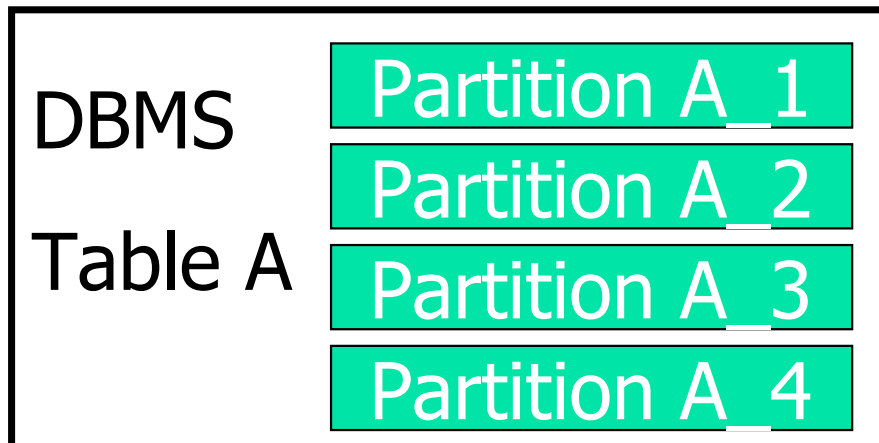
Possible Drawbacks: Conflicts with Parallel Jobs



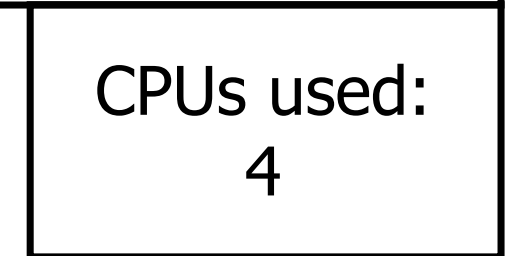
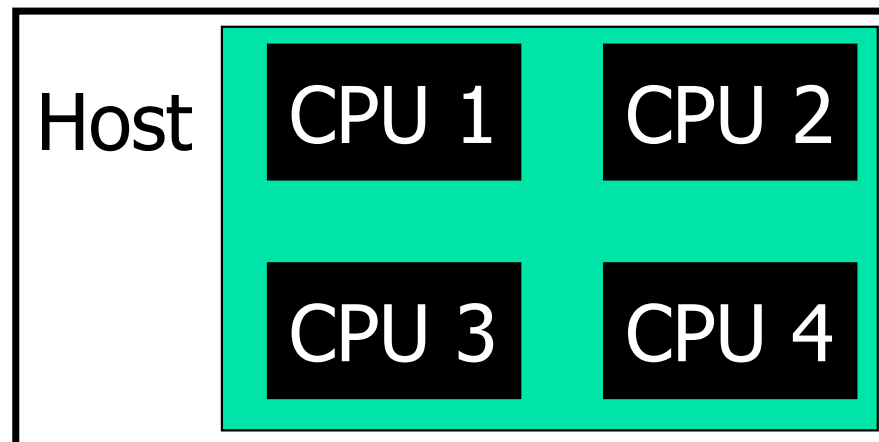
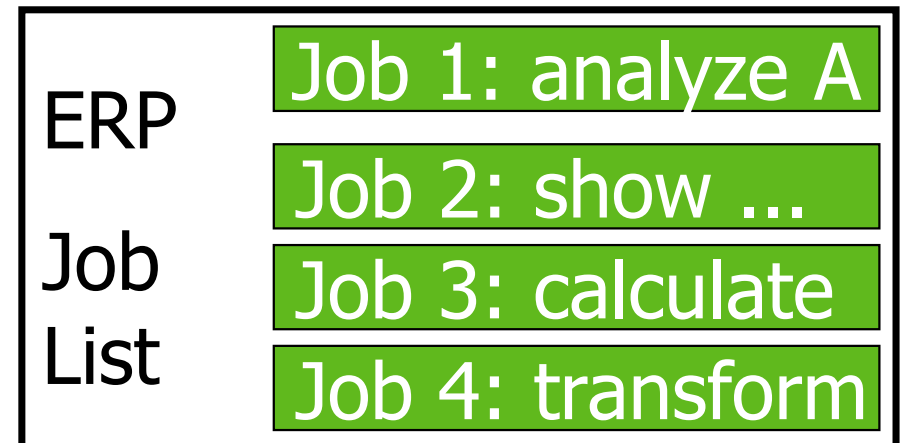
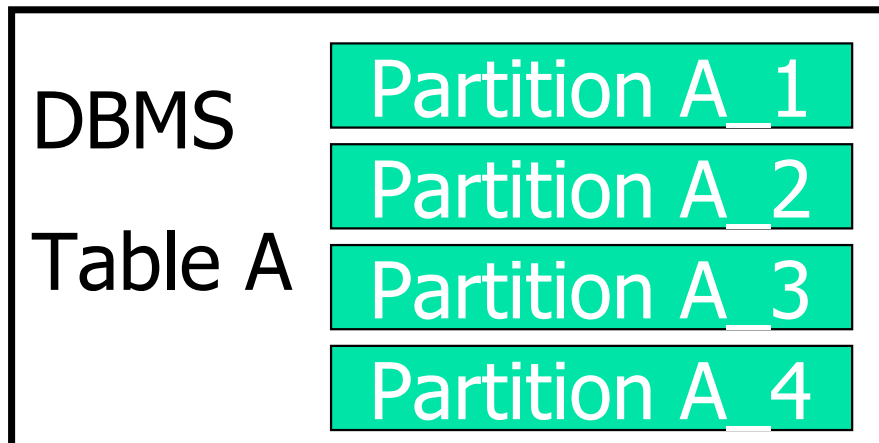
Possible Drawbacks: Conflicts with Parallel Jobs



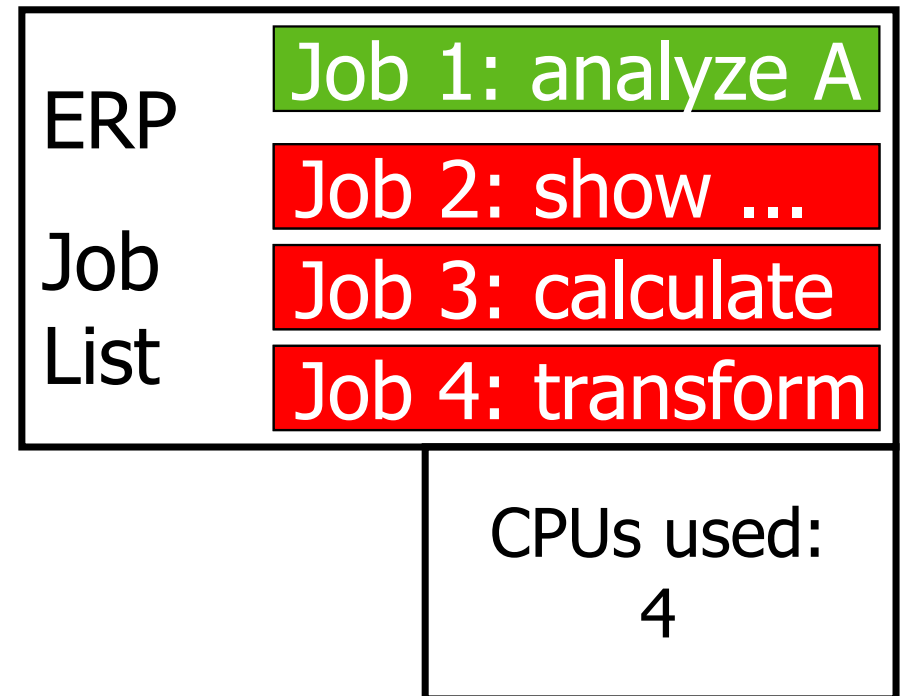
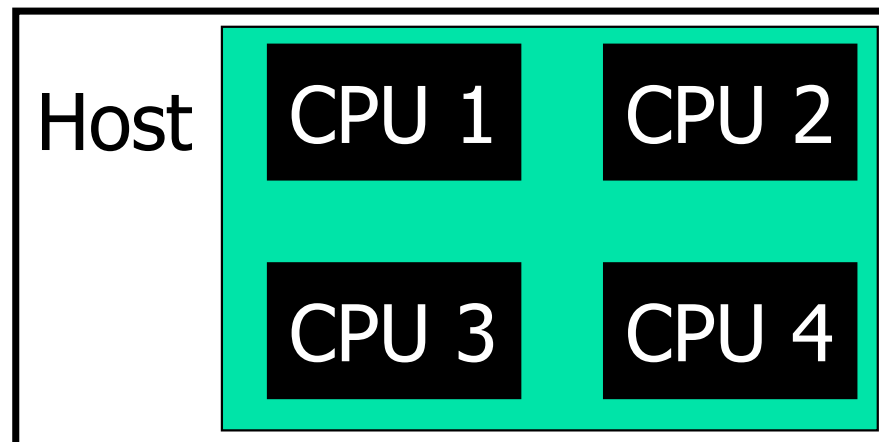
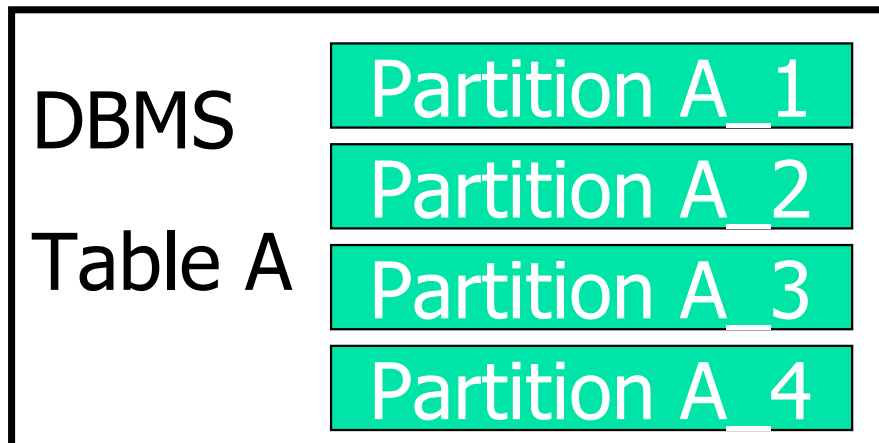
Possible Drawbacks: Conflicts with Parallel Jobs



Possible Drawbacks: Conflicts with Parallel Jobs



Possible Drawbacks: Conflicts with Parallel Jobs





Outline

- Brief Overview of SAP R/3
- Motivation
- Related Work
- Traditional Performance Tuning Techniques
- Exploiting Horizontal Partitioning for Tuning Purposes
 - Partitioning Scenarios/Techniques and their Pros and Cons
 - Possible Benefits and Drawbacks of Partitioning
- **Performance Analysis**
- **Conclusion**



Analyzed Scenarios

- Flat: table and index are not partitioned
- Global Index: the table is partitioned but the index is not
- Partitioned Index: the table and the index are partitioned using the same partitioning algorithm (range partitioning)



Used Data and Hardware (1)

Copies of SAP R/3 standard tables (Material Management):

	indexed key columns	av. row length	# rows	table size
MARC	mandt, matnr, werks	496 Byte	5 million	2.5 GB
MARD	mandt, matnr, werks, lgort	142 Byte	25 million	3.5 GB



Used Data and Hardware (2)

- range partitioned according to value of *werks* → 100 partitions
- anonymized data from a productive SAP R/3 system

Hardware:

- SUN Enterprise 450 with four 400 MHz CPUs and 4 GB RAM
- SUN A1000 500 GB RAID with RAID level 5
- the R/3 system and the DBMS used 512 MB main memory each



Analyzed Statements

- selects (single, for all entries, up to n rows, parallel selects,...)
- inserts, updates, and deletes
- joins
- parallel jobs at application level
- administrative tasks
- ...

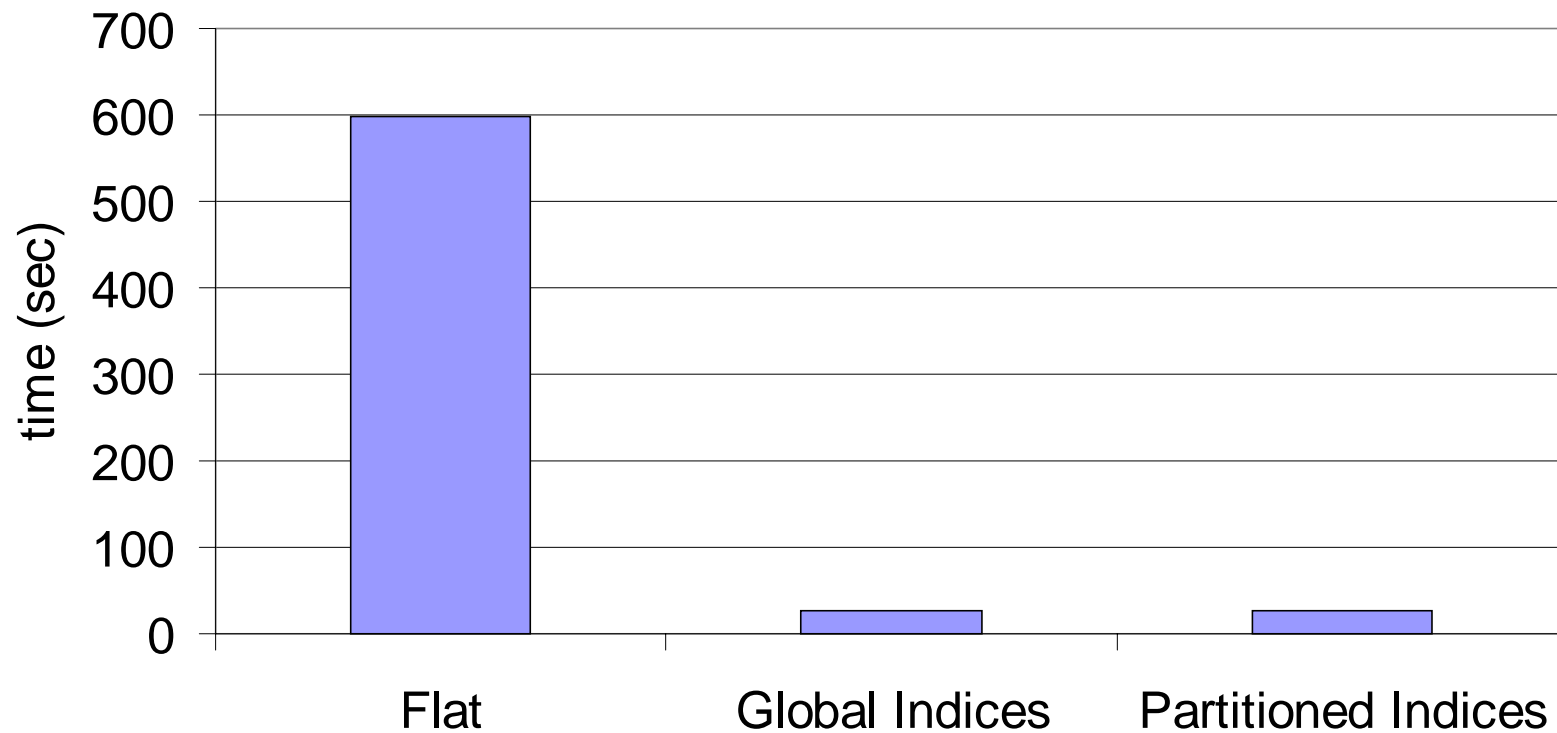


Evaluated Dimensions

- number of processed rows
- set orientated and one-record-at-a-time approach
- number of commits
- number of indices
- parallel jobs: processing data partition wise and
across partition borders

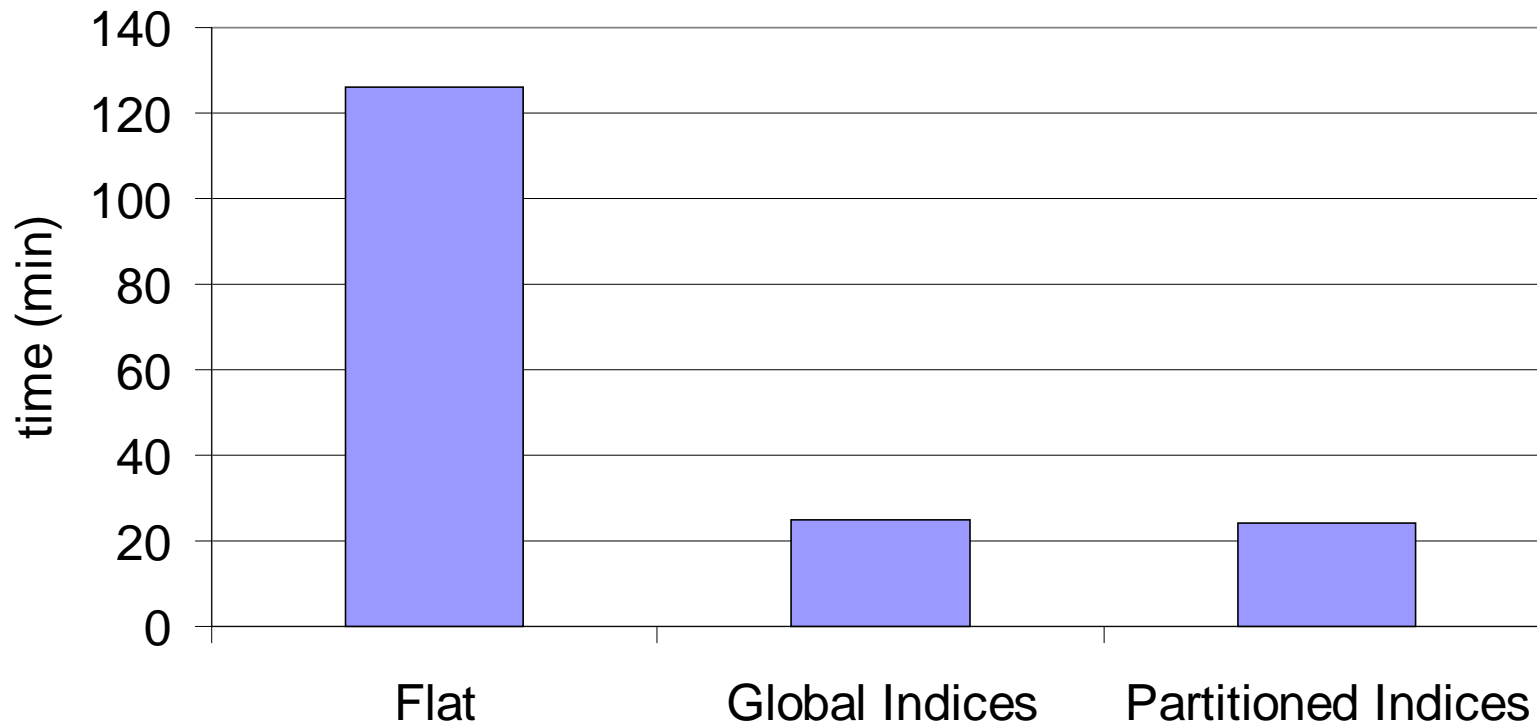
Joining Single Plants (Hash)

Join where mard.matnr = marc.matnr and mard.werks = marc.werks and mard.werks = '0001'



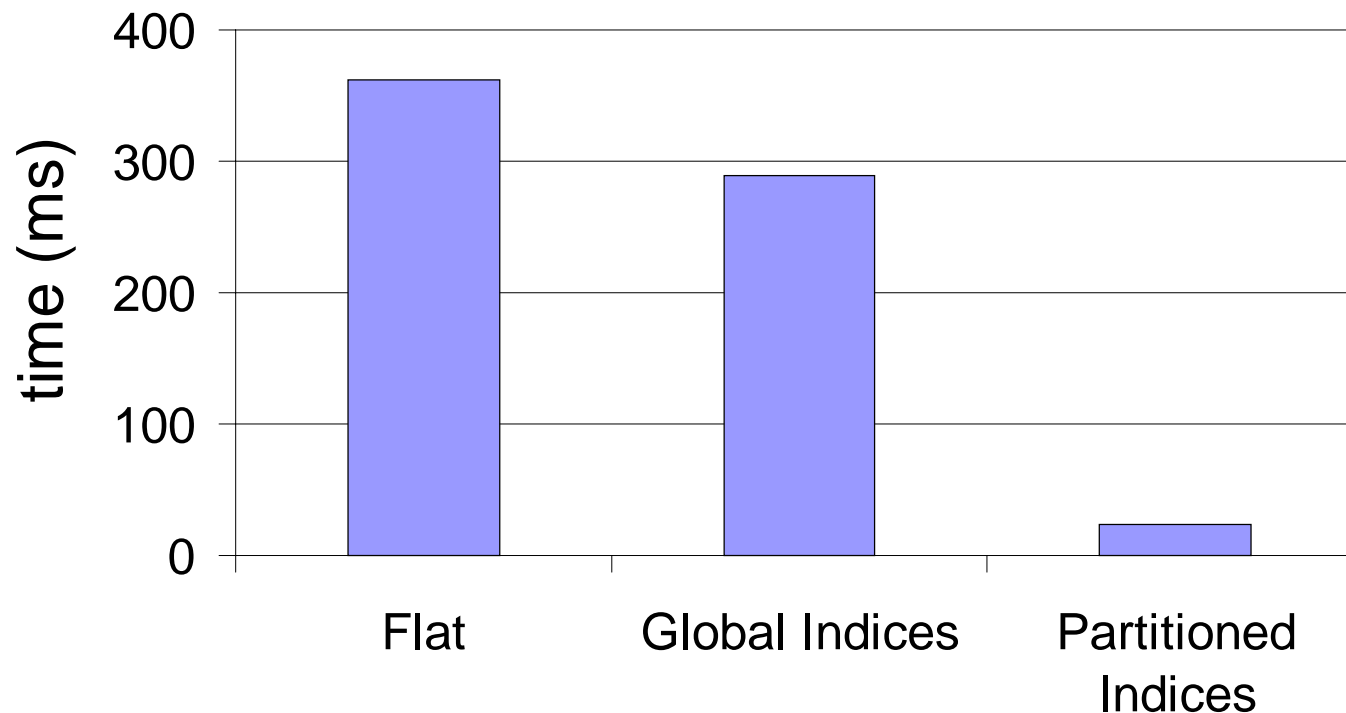
Joining Whole Tables (Hash)

Join where mard.matnr = marc.matnr and mard.werks = marc.werks



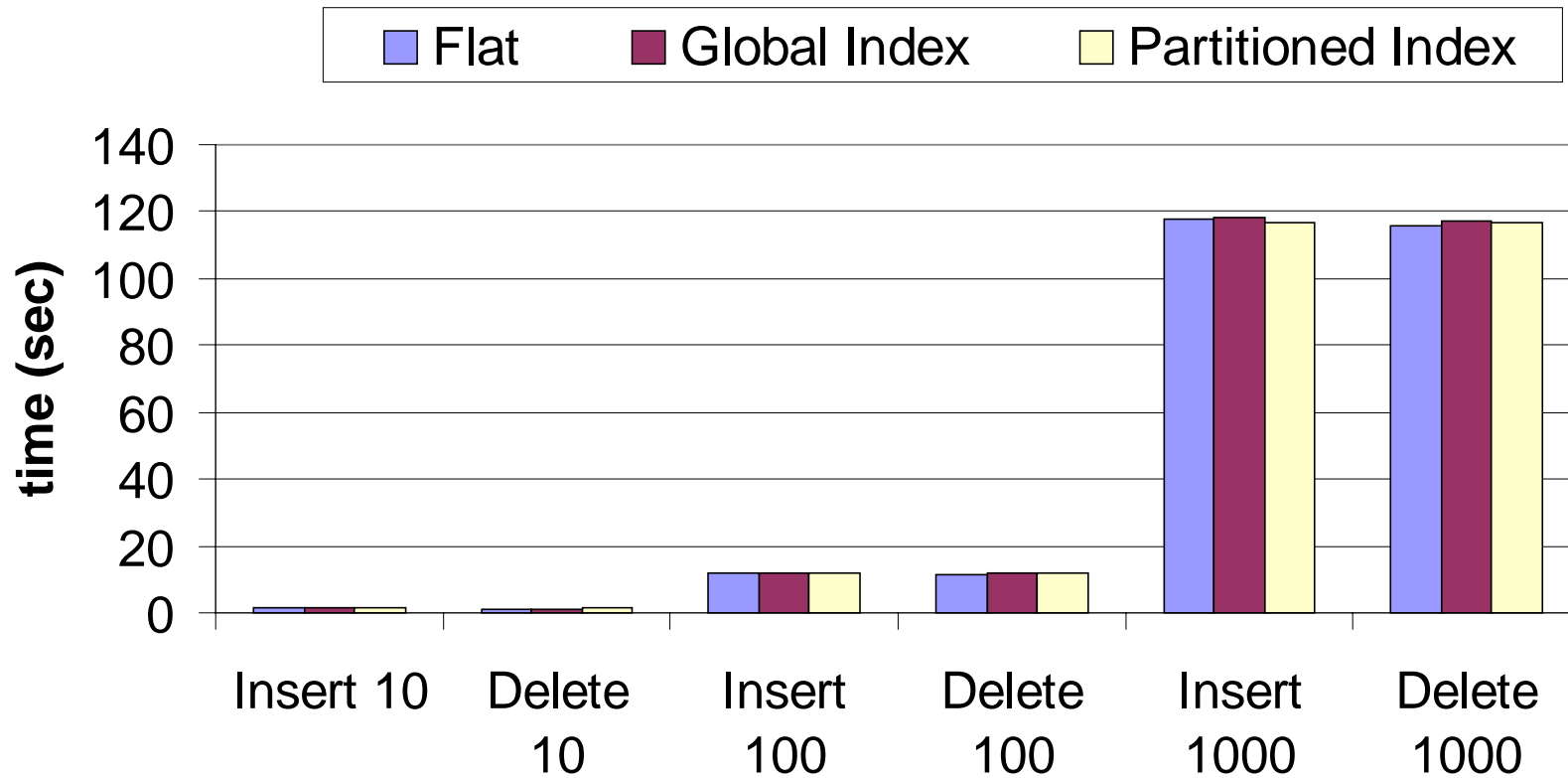
Joining Single Plants (Merge)

```
select /*+use_merge(c,d) index(c) index(d) */  
      c1.abcin,c2.speme  
from   marc c, mard d  
where  c.mandt = d.mandt and c.werks = d.werks  
       and c.matnr = d.matnr and c.werks = '0001'
```



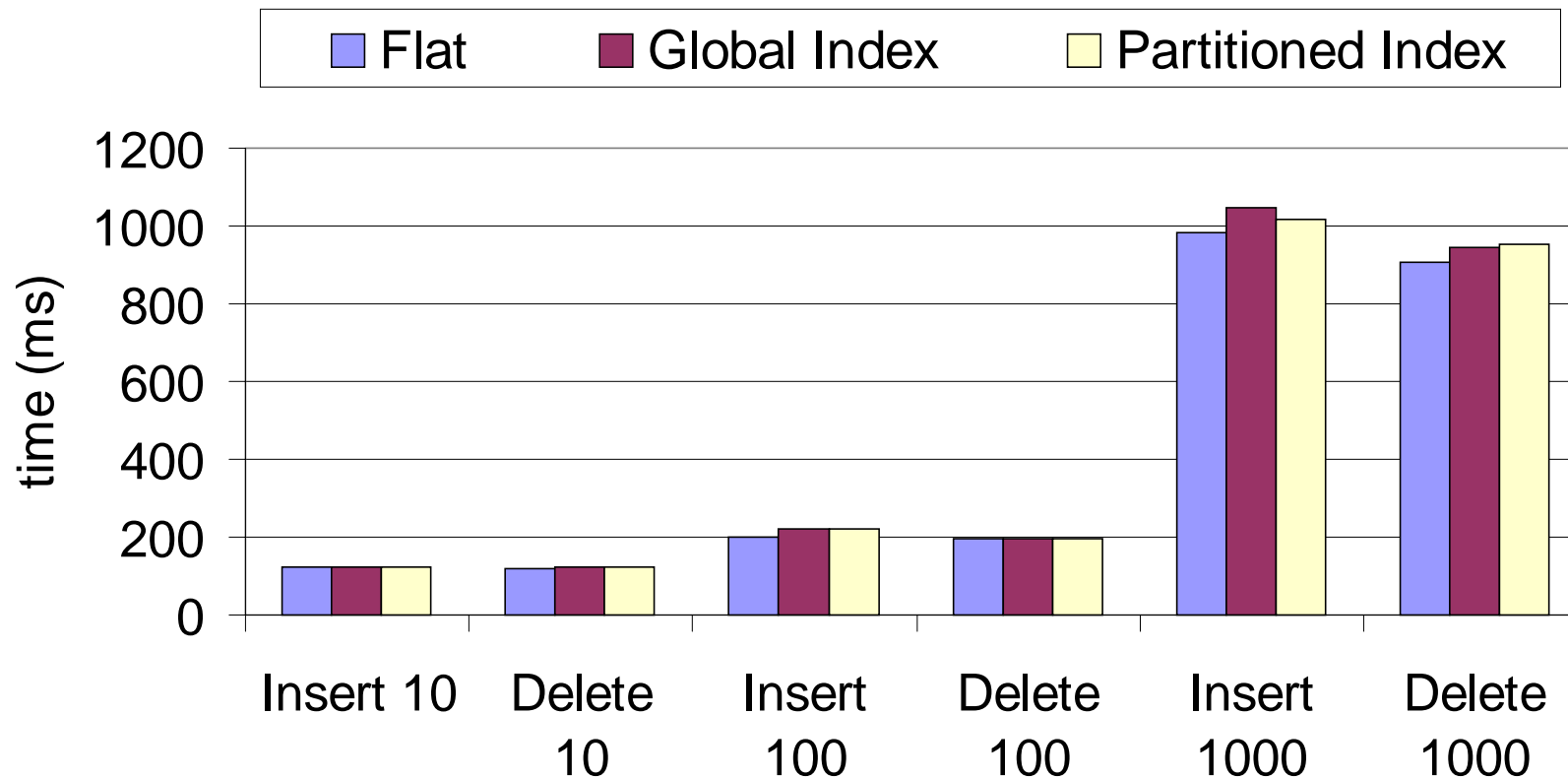
Insertions and Deletions

MARD, with commits

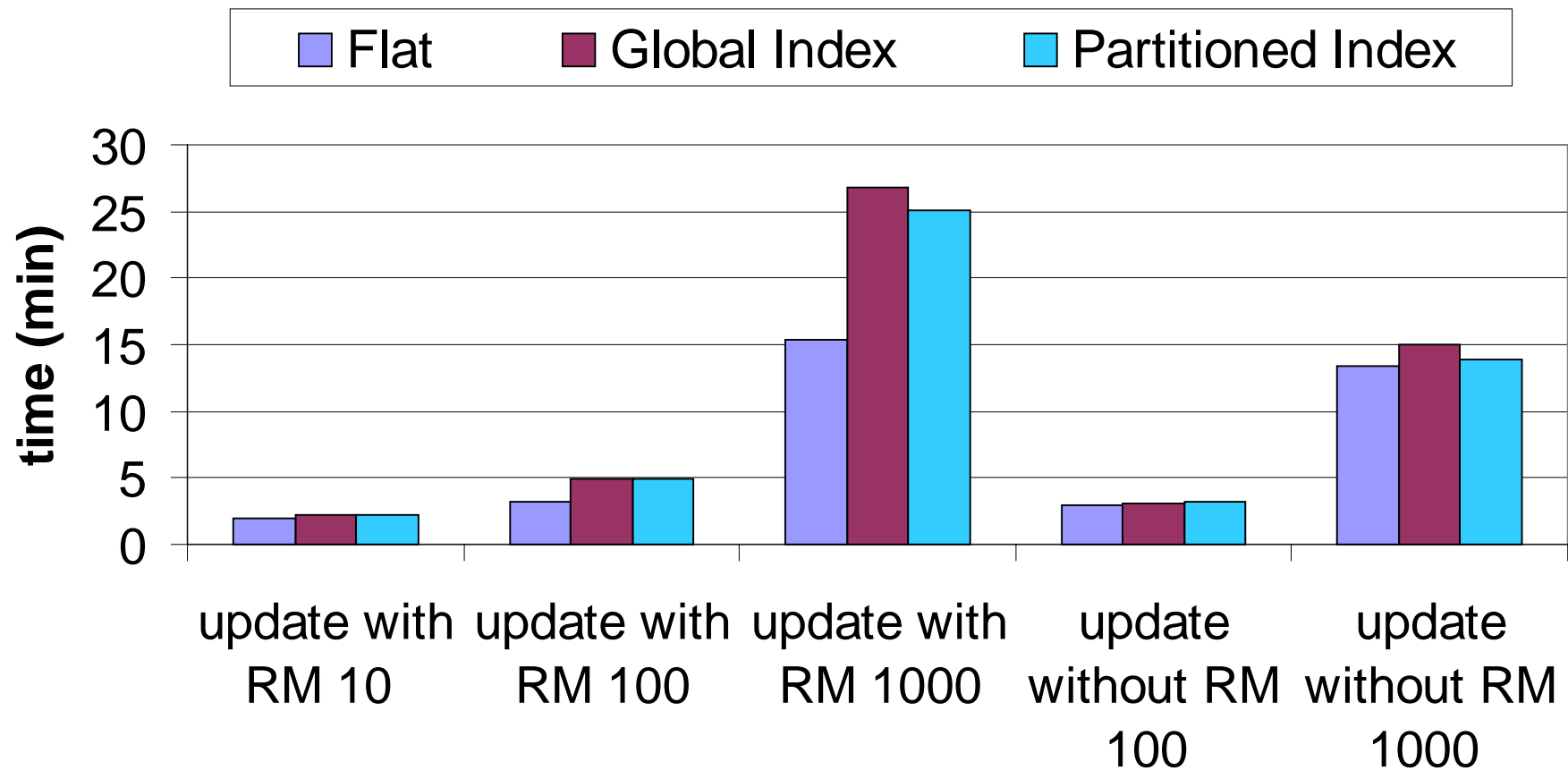


Insertions and Deletions

MARD, without commits

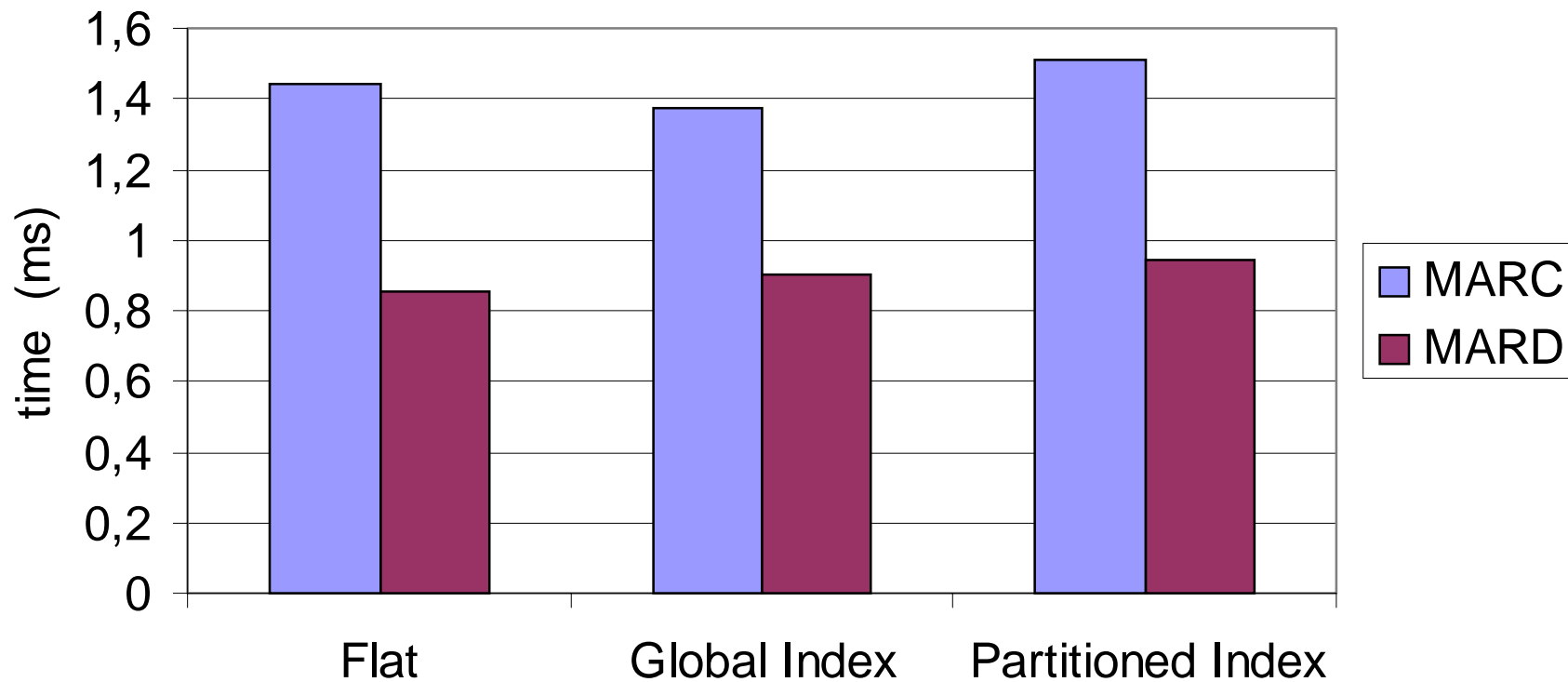


Update Statements



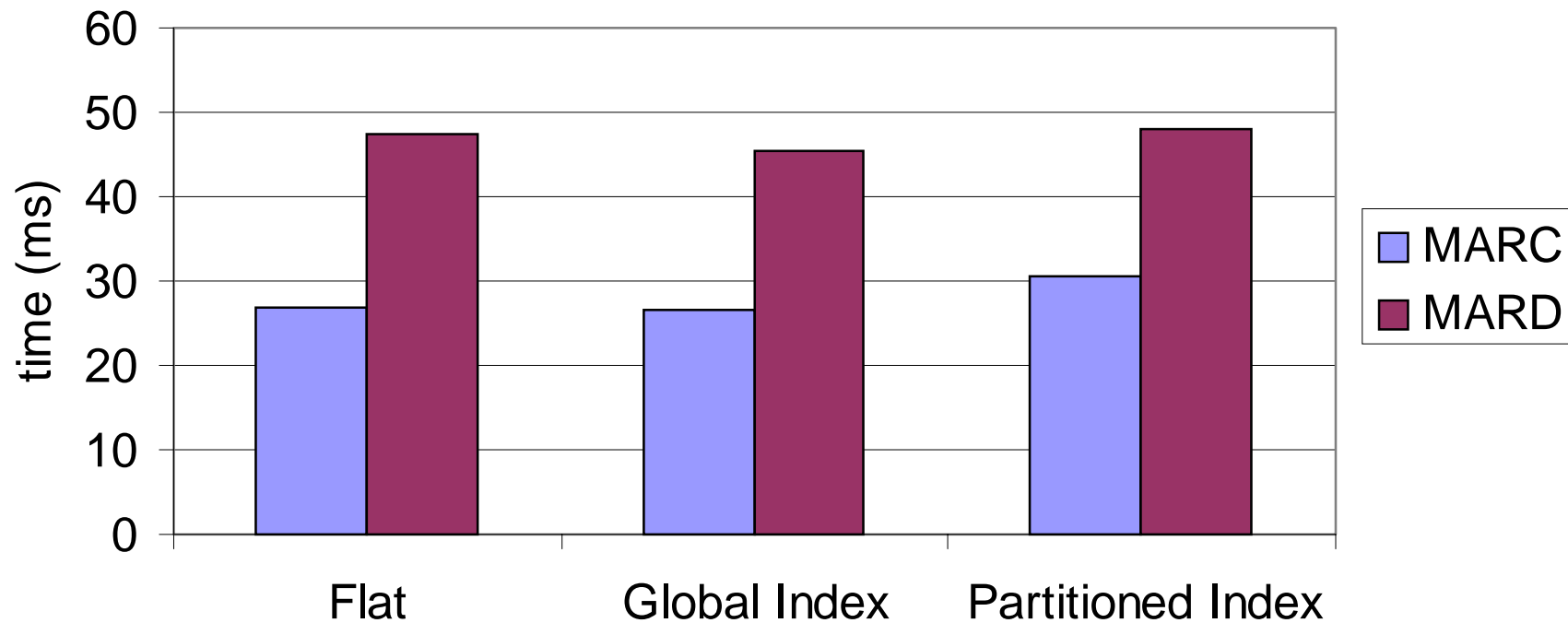
Select Statements

Select one record



Select Statements

Select across partition borders
(10 records from each partition)





Conclusion

- Results show: horizontal partitioning is applicable (with tolerable costs)
 - especially joins, administrative task, parallel selects benefit greatly
- horizontal partitioning is already used in some large-scale system



Thank you for your attention !

Questions ?