

BÀI THUYẾT TRÌNH

INTRODUCTION TO THE OPTIMIZER (CHAPTER 3)

Giảng viên hướng dẫn: TS. Trần Việt Trung

Nhóm sinh viên thực hiện: nhóm 02

Trương Lộc Bình	<i>2011 1177</i>
Đặng Xuân Trường	<i>2011 2356</i>
Lê Công Thái	<i>2011 2673</i>
Nguyễn Huy Hùng	<i>2011 1520</i>

PHÂN CÔNG CÔNG VIỆC

Họ và tên	Công việc thực hiện
Trương Lộc Bình	✓ Nghiên cứu tài liệu: phần 3 ✓ Thuyết trình bài tập lớn
Đặng Xuân Trường	✓ Nghiên cứu tài liệu: phần 1, 2
Lê Công Thái	✓ Nghiên cứu tài liệu: phần 1, 2
Nguyễn Huy Hùng	✓ Nghiên cứu tài liệu: phần 3 ✓ Tổng hợp, viết báo cáo

OUTLINE

- 1 Sơ lược về kiến trúc cơ sở dữ liệu Oracle
- 2 Câu lệnh SQL
- 3 Tối ưu hóa trong cơ sở dữ liệu Oracle

OUTLINE

1 Sơ lược về kiến trúc cơ sở dữ liệu Oracle

1.1 Các thành phần chính

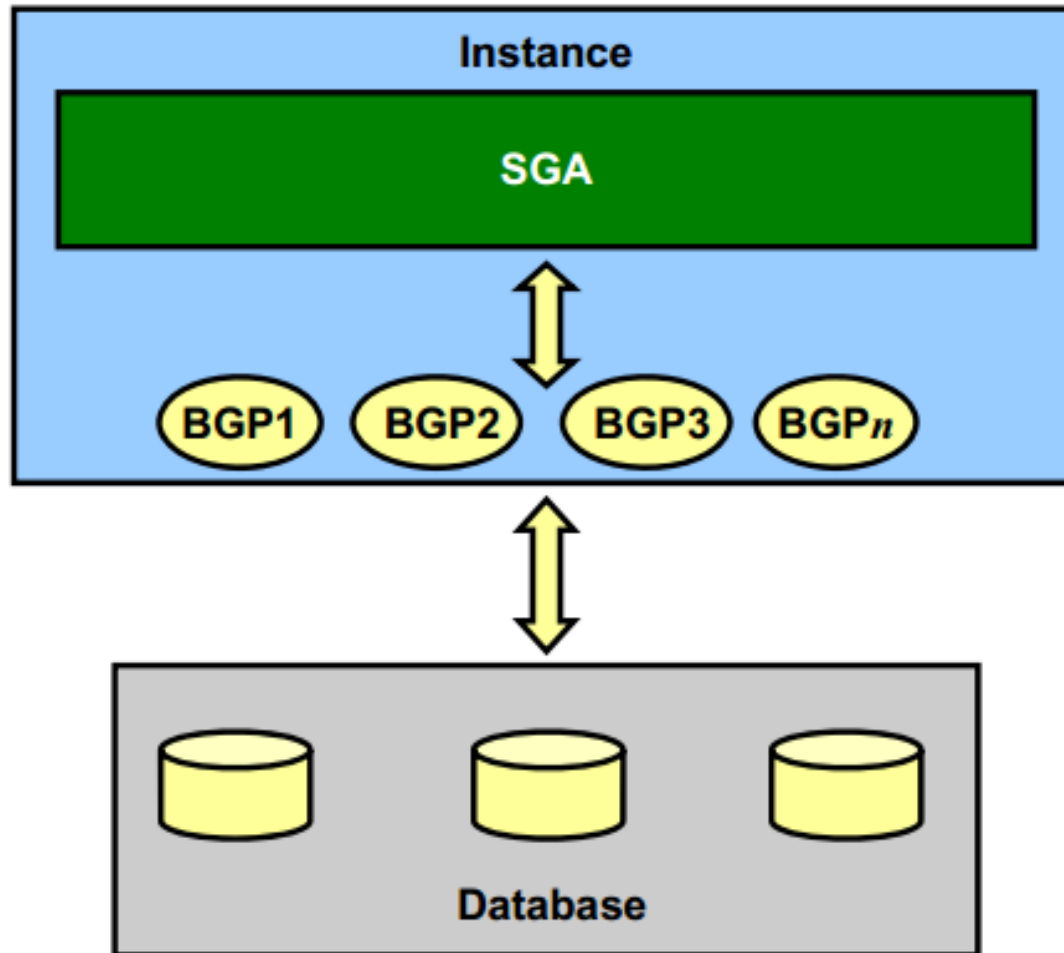
1.2 Cấu trúc bộ nhớ

1.3 Cấu trúc lưu trữ vật lý và logic

2 Câu lệnh SQL

3 Tối ưu hóa trong cơ sở dữ liệu Oracle

TỔNG QUAN KIẾN TRÚC ORACLE DATABASE



OUTLINE

1 Sơ lược về kiến trúc cơ sở dữ liệu Oracle

1.1 Các thành phần chính

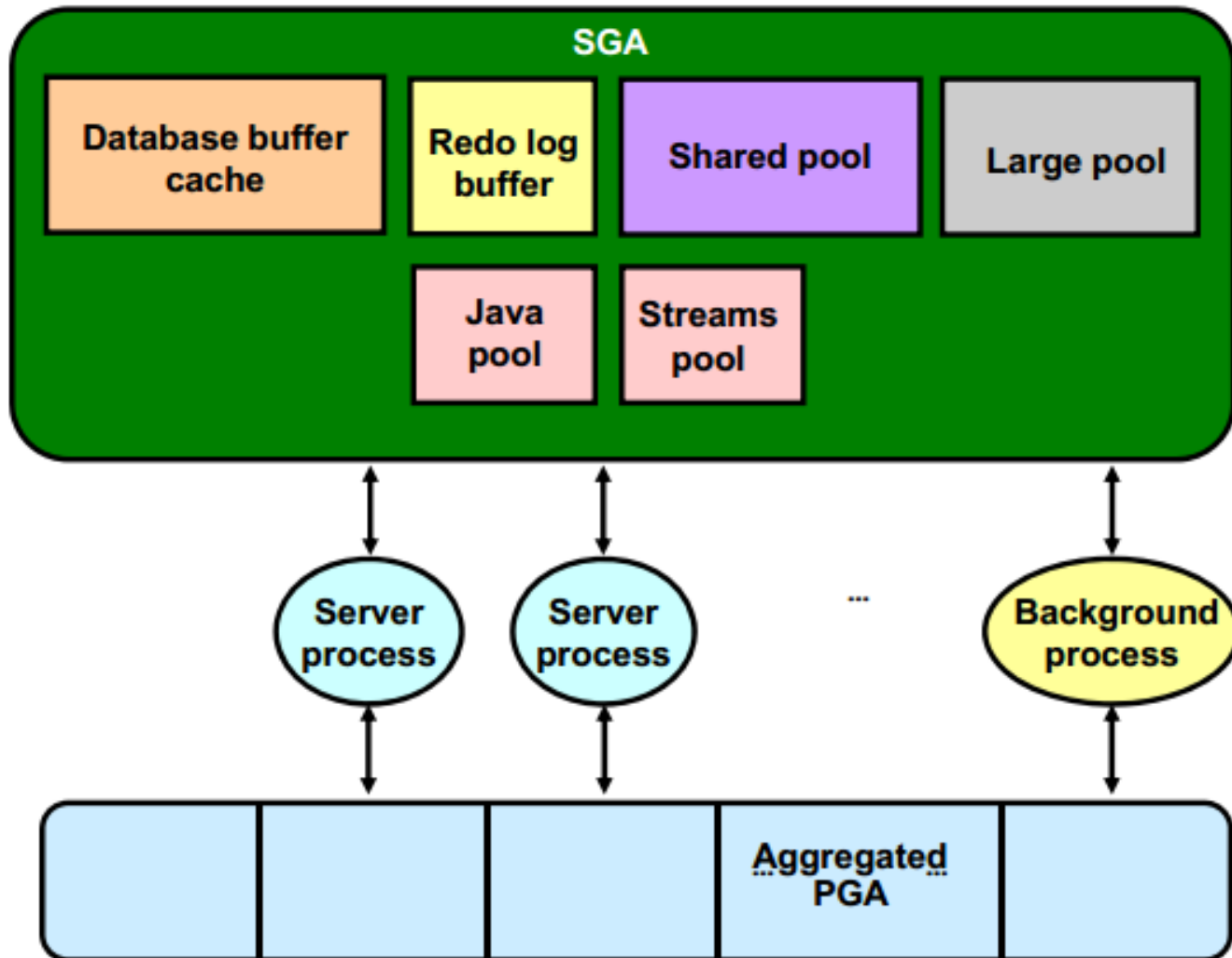
1.2 Cấu trúc bộ nhớ

1.3 Cấu trúc lưu trữ vật lý và logic

2 Câu lệnh SQL

3 Tối ưu hóa trong cơ sở dữ liệu Oracle

KIẾN TRÚC BỘ NHỚ



OUTLINE

1 Sơ lược về kiến trúc cơ sở dữ liệu Oracle

1.1 Các thành phần chính

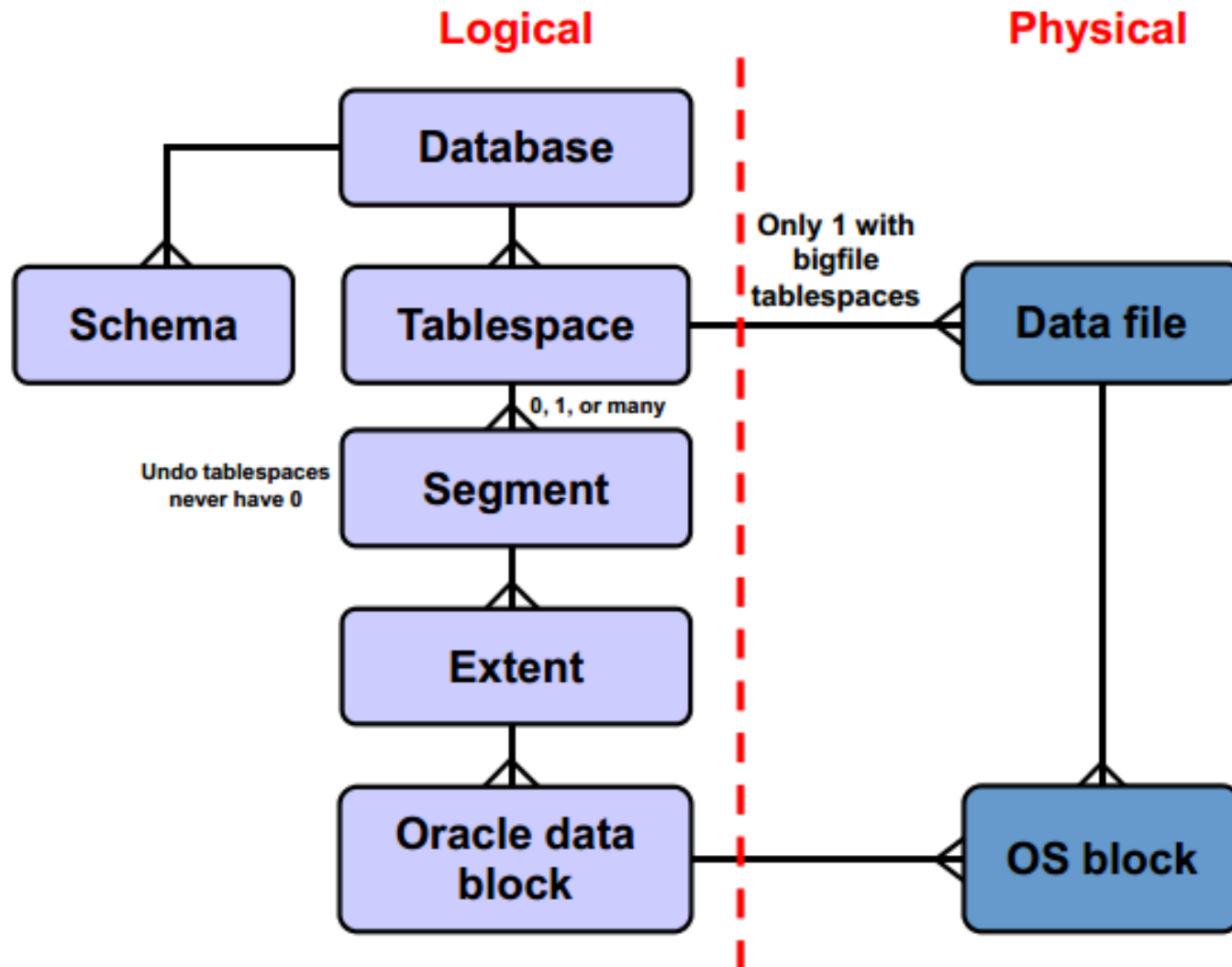
1.2 Cấu trúc bộ nhớ

1.3 Cấu trúc lưu trữ vật lý và logic

2 Câu lệnh SQL

3 Tối ưu hóa trong cơ sở dữ liệu Oracle

CẤU TRÚC LƯU TRỮ VẬT LÝ VÀ LOGIC



OUTLINE

1 Sơ lược về kiến trúc cơ sở dữ liệu Oracle

2 Câu lệnh SQL

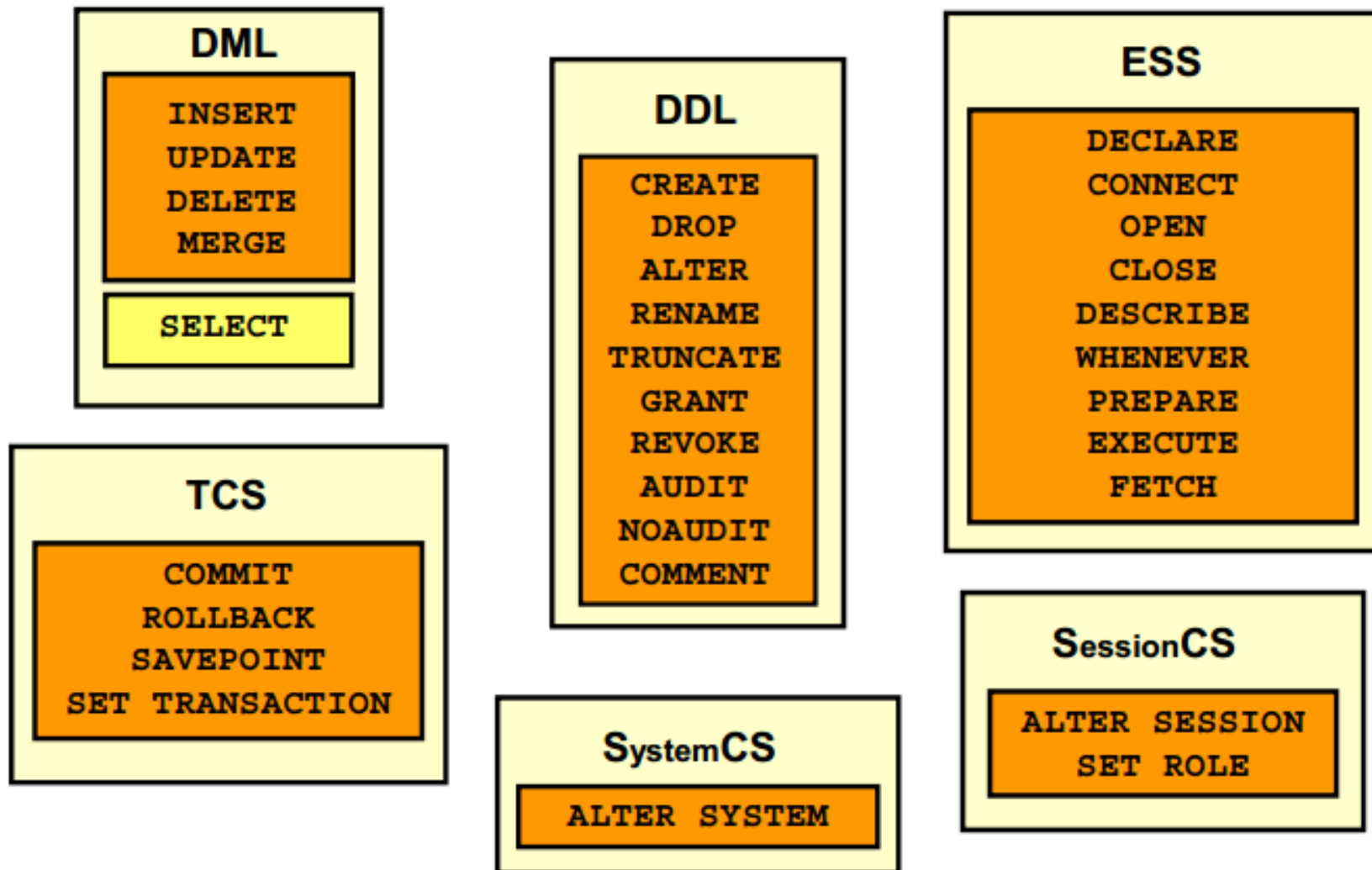
2.1 Phân loại

2.2 Biểu diễn và cài đặt

2.3 Quá trình thực thi

3 Tối ưu hóa trong cơ sở dữ liệu Oracle

6 LOẠI CÂU LỆNH SQL



OUTLINE

1 Sơ lược về kiến trúc cơ sở dữ liệu Oracle

2 Câu lệnh SQL

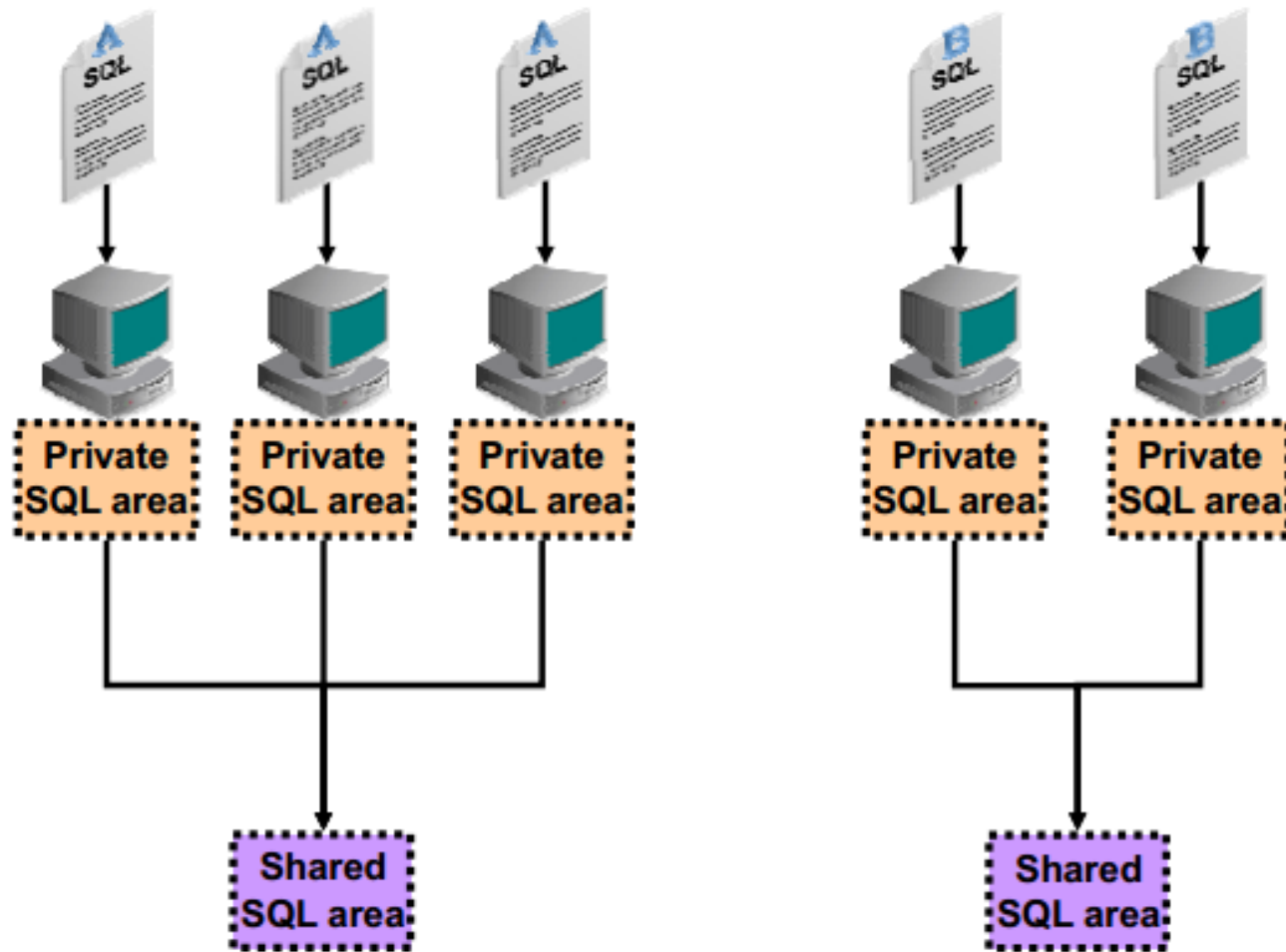
2.1 Phân loại

2.2 Biểu diễn và cài đặt

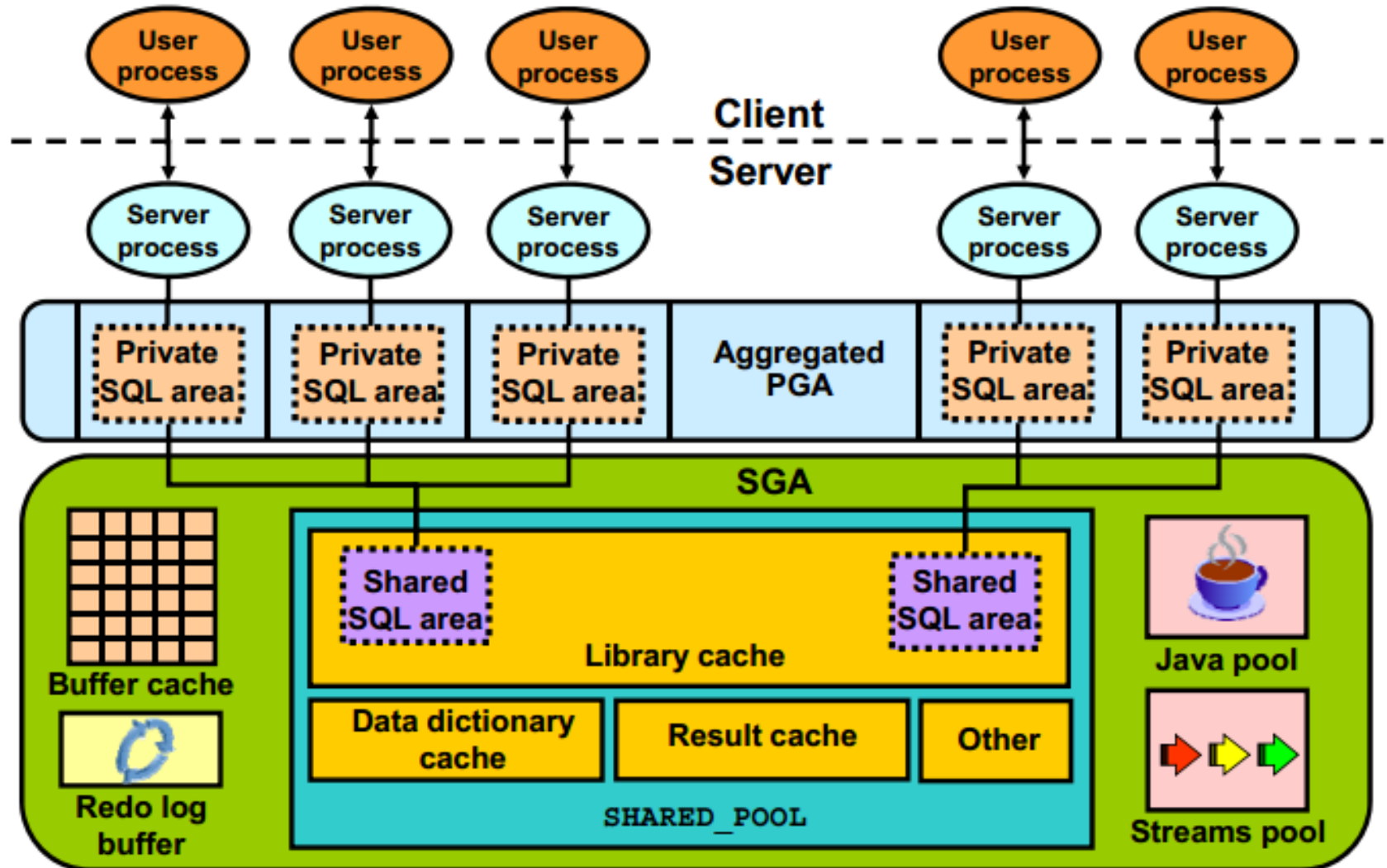
2.3 Quá trình thực thi

3 Tối ưu hóa trong cơ sở dữ liệu Oracle

BIỂU DIỄN CÂU LỆNH SQL



CÀI ĐẶT



OUTLINE

1 Sơ lược về kiến trúc cơ sở dữ liệu Oracle

2 Câu lệnh SQL

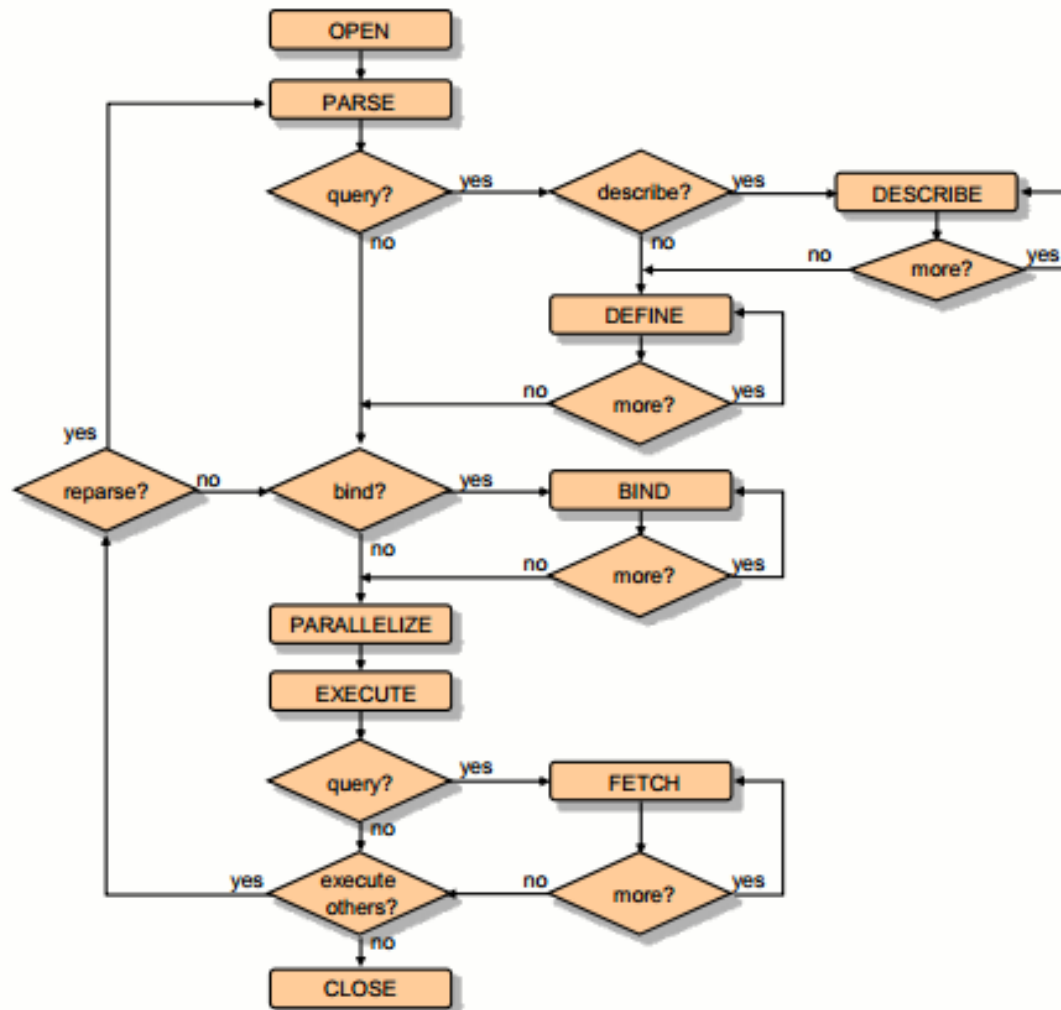
2.1 Phân loại

2.2 Biểu diễn và cài đặt

2.3 Quá trình thực thi

3 Tối ưu hóa trong cơ sở dữ liệu Oracle

QUÁ TRÌNH XỬ LÝ CÂU LỆNH SQL



CÁC BƯỚC XỬ LÝ

1. Create a cursor.
2. Parse the statement.
3. Describe query results.
4. Define query output.
5. Bind variables.
6. Parallelize the statement.
7. Execute the statement.
8. Fetch rows of a query.
9. Close the cursor.

Step 1: Create a Cursor

- A cursor is a handle or name for a private SQL area.
- It contains information for statement processing.
- It is created by a program interface call in expectation of a SQL statement.
- The cursor structure is independent of the SQL statement that it contains.

Step 2: Parse the Statement

- Statement passed from the user process to the Oracle instance
- Parsed representation of SQL created and moved into the shared SQL area if there is no identical SQL in the shared SQL area
- Can be reused if identical SQL exists

Steps 3 and 4: Describe and Define

- The describe step provides information about the select list items; it is relevant when entering dynamic queries through an OCI application.
- The define step defines location, size, and data type information required to store fetched values in variables.

Steps 5 and 6: Bind and Parallelize

- Bind any bind values:
 - Enables memory address to store data values
 - Allows shared SQL even though bind values may change
- Parallelize the statement:
 - SELECT
 - INSERT
 - UPDATE
 - MERGE
 - DELETE
 - CREATE
 - ALTER

Steps 7 Through 9

- Execute:
 - Drives the SQL statement to produce the desired results
- Fetch rows:
 - Into defined output variables
 - Query results returned in table format
 - Array fetch mechanism
- Close the cursor.

OUTLINE

1 Sơ lược về kiến trúc cơ sở dữ liệu Oracle

2 Câu lệnh SQL

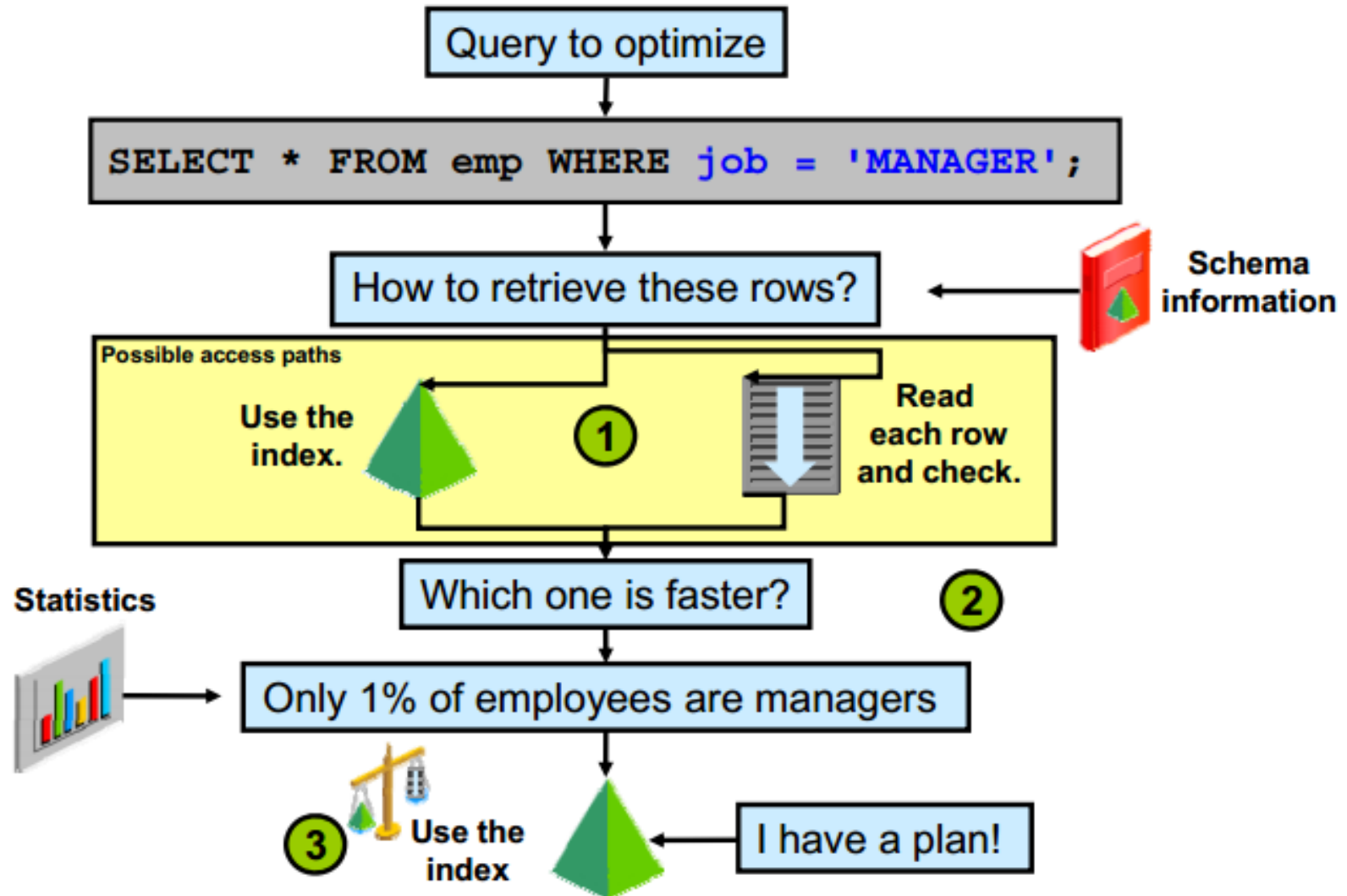
3 Tối ưu hóa trong cơ sở dữ liệu Oracle

3.1 Sự cần thiết của bộ tối ưu hóa

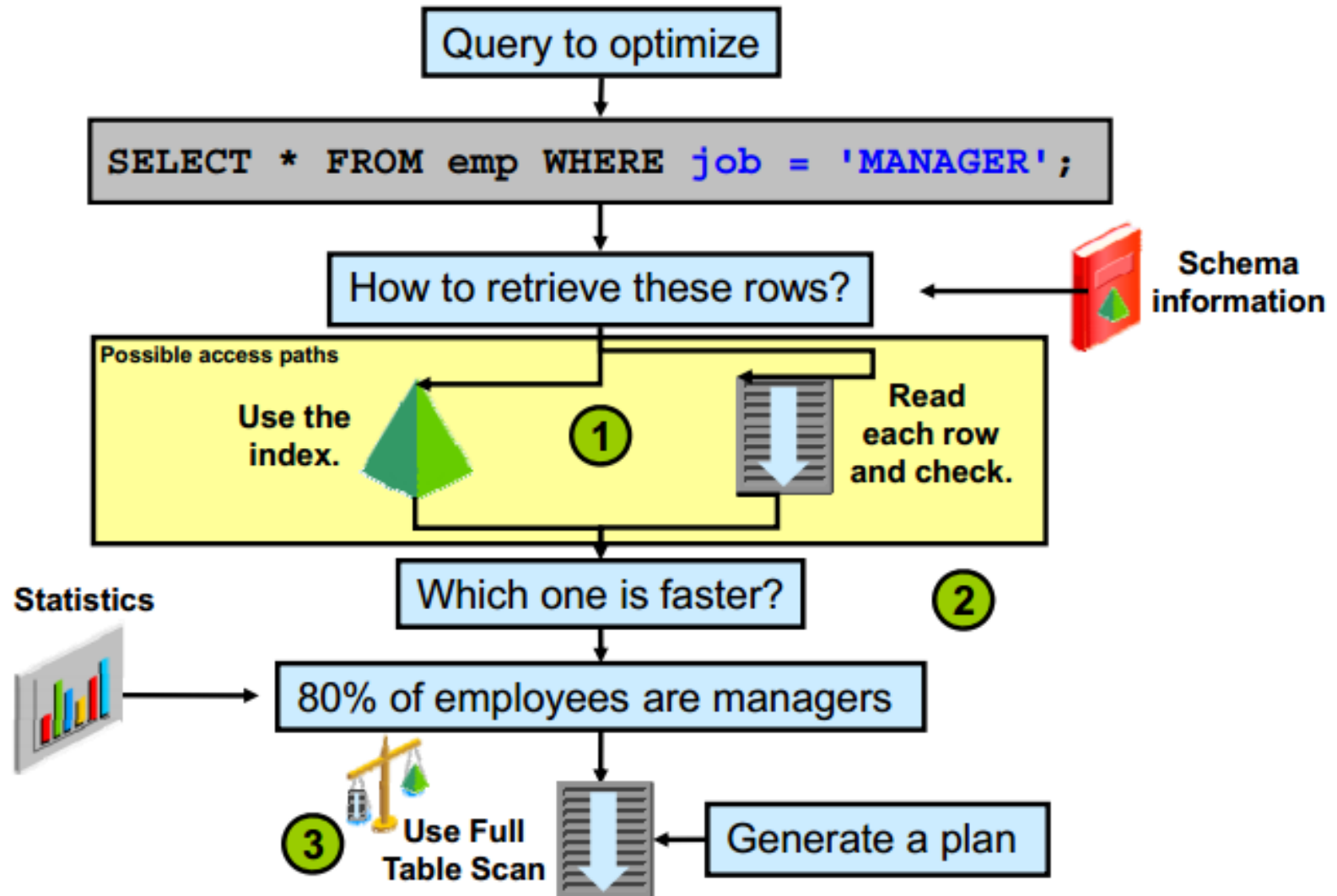
3.2 Các pha của quá trình tối ưu

3.3 Điều khiển hành vi của bộ tối ưu hóa

TẠI SAO CẦN CÓ BỘ TỐI ƯU HÓA?



TẠI SAO CẦN CÓ BỘ TỐI ƯU HÓA? (tiếp)



OUTLINE

1 Sơ lược về kiến trúc cơ sở dữ liệu Oracle

2 Câu lệnh SQL

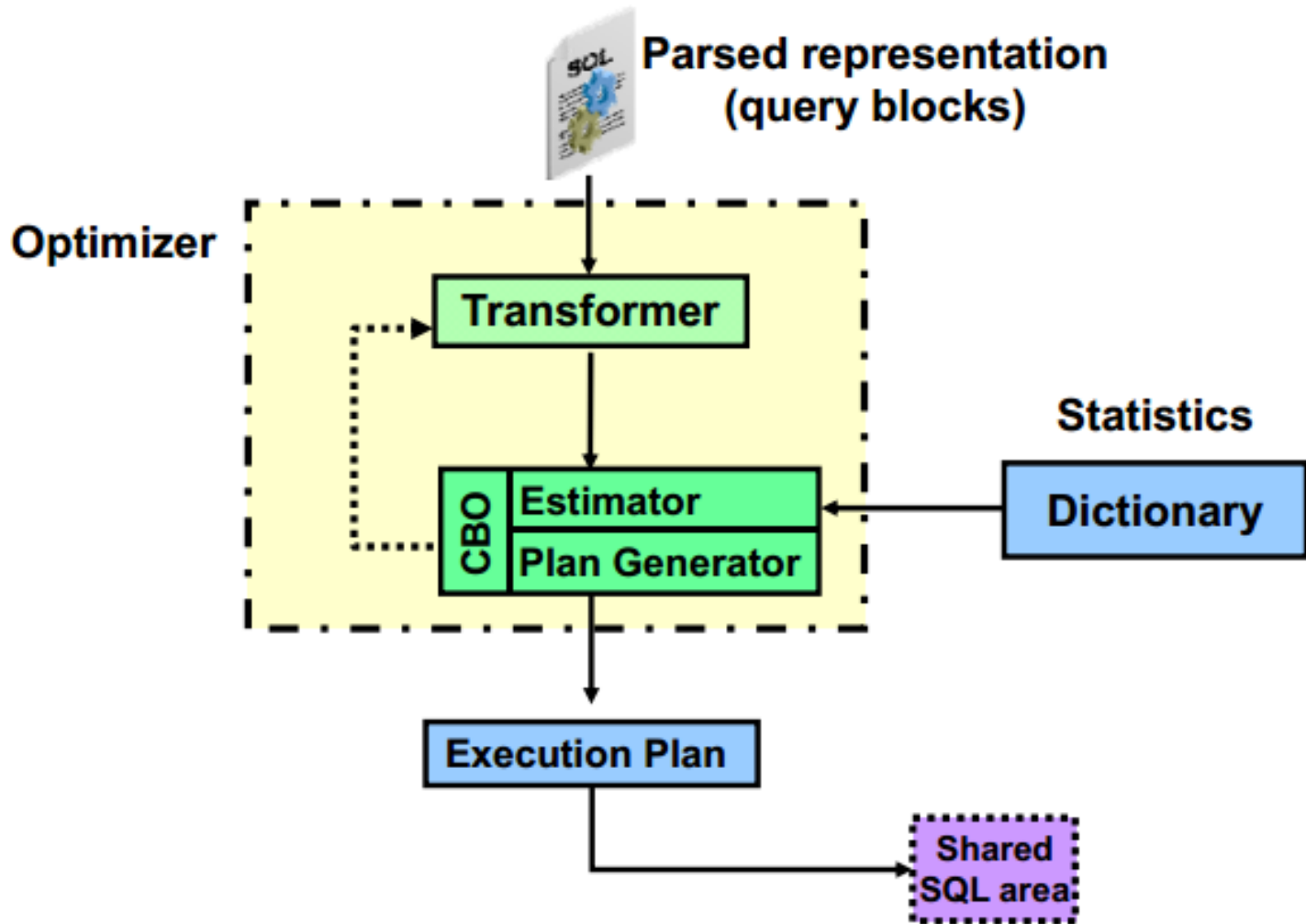
3 Tối ưu hóa trong cơ sở dữ liệu Oracle

3.1 Sự cần thiết của bộ tối ưu hóa

3.2 Các pha của quá trình tối ưu


3.3 Điều khiển hành vi của bộ tối ưu hóa

CÁC PHA CỦA QUÁ TRÌNH TỐI ƯU



BỘ CHUYỂN ĐỔI: phép OR mở rộng

- Original query:

 B*-tree Index

```
SELECT *  
  FROM emp  
 WHERE job = 'CLERK' OR deptno = 10;
```

- Equivalent transformed query:

```
SELECT *  
  FROM emp  
 WHERE job = 'CLERK'  
UNION ALL  
SELECT *  
  FROM emp  
 WHERE deptno = 10 AND job <> 'CLERK';
```

BỘ CHUYỂN ĐỔI: truy vấn con không lồng nhau

- Original query:

```
SELECT *  
  FROM accounts  
 WHERE custno IN  
        (SELECT custno FROM customers);
```

- Equivalent transformed query:

```
SELECT accounts.*  
  FROM accounts, customers  
 WHERE accounts.custno = customers.custno;
```

↑
Primary or unique key

BỘ CHUYỂN ĐỔI: trộn khung nhìn

- Original query:

 Index

```
CREATE VIEW emp_10 AS  
  SELECT empno, ename, job, sal, comm, deptno  
  FROM emp  
  WHERE deptno = 10;
```

```
SELECT empno FROM emp_10 WHERE empno > 7800;
```

- Equivalent transformed query:

```
SELECT empno  
  FROM emp  
  WHERE deptno = 10 AND empno > 7800;
```

BỘ CHUYỂN ĐỔI: vị từ mở rộng

- Original query:

 Index

```
CREATE VIEW two_emp_tables AS
SELECT empno, ename, job, sal, comm, deptno FROM emp1
UNION
SELECT empno, ename, job, sal, comm, deptno FROM emp2;
```

```
SELECT ename FROM two_emp_tables WHERE deptno = 20;
```

- Equivalent transformed query:

```
SELECT ename
FROM ( SELECT empno, ename, job, sal, comm, deptno
      FROM emp1 WHERE deptno = 20
      UNION
      SELECT empno, ename, job, sal, comm, deptno
      FROM emp2 WHERE deptno = 20 );
```

BỘ CHUYỂN ĐỔI: tính chất bắc cầu

- Original query:

 Index

```
SELECT *  
FROM emp, dept  
WHERE emp.deptno = 20 AND emp.deptno = dept.deptno;
```

- Equivalent transformed query:

```
SELECT *  
FROM emp, dept  
WHERE emp.deptno = 20 AND emp.deptno = dept.deptno  
AND dept.deptno = 20;
```


TỐI ƯU HÓA DỰA TRÊN CHI PHÍ

- Piece of code:
 - Estimator
 - Plan generator
- Estimator determines cost of optimization suggestions made by the plan generator:
 - Cost: Optimizer's best estimate of the number of standardized I/Os made to execute a particular statement optimization
- Plan generator:
 - Tries out different statement optimization techniques
 - Uses the estimator to cost each optimization suggestion
 - Chooses the best optimization suggestion based on cost
 - Generates an execution plan for best optimization

BỘ ƯỚC LƯỢNG: tính chọn lọc

$$\text{Selectivity} = \frac{\text{Number of rows satisfying a condition}}{\text{Total number of rows}}$$

- Selectivity is the estimated proportion of a row set retrieved by a particular predicate or combination of predicates.
- It is expressed as a value between 0.0 and 1.0:
 - High selectivity: Small proportion of rows
 - Low selectivity: Big proportion of rows
- Selectivity computation:
 - If no statistics: Use dynamic sampling
 - If no histograms: Assume even distribution of rows
- Statistic information:
 - `DBA_TABLES` and `DBA_TAB_STATISTICS` (`NUM_ROWS`)
 - `DBA_TAB_COL_STATISTICS` (`NUM_DISTINCT`, `DENSITY`, `HIGH/LOW_VALUE`,...)

BỘ ƯỚC LƯỢNG: lực lượng

Cardinality = Selectivity * Total number of rows

- Expected number of rows retrieved by a particular operation in the execution plan
- Vital figure to determine join, filters, and sort costs
- Simple example:

```
SELECT days FROM courses WHERE dev_name = 'ANGEL';
```

- The number of distinct values in DEV_NAME is 203.
- The number of rows in COURSES (original cardinality) is 1018.
- Selectivity = $1/203 = 4.926 \times 10^{-3}$
- Cardinality = $(1/203) \times 1018 = 5.01$ (rounded off to 6)

BỘ ƯỚC LƯỢNG: chi phí

- Cost is the optimizer's best estimate of the number of standardized I/Os it takes to execute a particular statement.
- Cost unit is a standardized single block random read:
 - 1 cost unit = 1 SRds
- The cost formula combines three different costs units into standard cost units.

$$\text{Cost} = \frac{\begin{array}{|c|} \hline \text{Single block I/O cost} \\ \hline \#SRds * sreadtim \\ \hline \end{array} + \begin{array}{|c|} \hline \text{Multiblock I/O cost} \\ \hline \#MRds * mreadtim \\ \hline \end{array} + \begin{array}{|c|} \hline \text{CPU cost} \\ \hline \#CPUCycles / cpuspeed \\ \hline \end{array}}{sreadtim}$$

#SRds: Number of single block reads

#MRds: Number of multiblock reads

#CPUCycles: Number of CPU Cycles

Sreadtim: Single block read time

Mreadtim: Multiblock read time

Cpuspeed: Millions instructions per second

BỘ SINH KẾ HOẠCH THỰC THI

```
select e.last_name, d.department_name
from   employees e, departments d
where  e.department_id = d.department_id;
```

```
Join order[1]:  DEPARTMENTS[D]#0  EMPLOYEES[E]#1
NL Join:   Cost: 41.13  Resp: 41.13  Degree: 1
SM cost: 8.01
HA cost: 6.51
```

Best:: JoinMethod: Hash

Cost: 6.51 Degree: 1 Resp: 6.51 Card: 106.00

```
Join order[2]:  EMPLOYEES[E]#1  DEPARTMENTS[D]#0
NL Join:   Cost: 121.24  Resp: 121.24  Degree: 1
SM cost: 8.01
HA cost: 6.51
```

Join order aborted

Final cost for query block SEL\$1 (#0)

All Rows Plan:

Best join order: 1

+-----+-----+-----+-----+-----+-----+-----+						
Id	Operation	Name	Rows	Bytes	Cost	
+-----+-----+-----+-----+-----+-----+-----+						
0	SELECT STATEMENT				7	
1	HASH JOIN		106	6042	7	
2	TABLE ACCESS FULL	DEPARTMENTS	27	810	3	
3	TABLE ACCESS FULL	EMPLOYEES	107	2889	3	
+-----+-----+-----+-----+-----+-----+-----+						

OUTLINE

1 Sơ lược về kiến trúc cơ sở dữ liệu Oracle

2 Câu lệnh SQL

3 Tối ưu hóa trong cơ sở dữ liệu Oracle

3.1 Sự cần thiết của bộ tối ưu hóa

3.2 Các pha của quá trình tối ưu

3.3 Điều khiển hành vi của bộ tối ưu hóa

ĐIỀU KHIỂN HÀNH VI BỘ TỐI ƯU HÓA

- `CURSOR_SHARING`: `SIMILAR`, `EXACT`, `FORCE`
- `DB_FILE_MULTIBLOCK_READ_COUNT`
- `PGA_AGGREGATE_TARGET`
- `STAR_TRANSFORMATION_ENABLED`
- `RESULT_CACHE_MODE`: `MANUAL`, `FORCE`
- `RESULT_CACHE_MAX_SIZE`
- `RESULT_CACHE_MAX_RESULT`
- `RESULT_CACHE_REMOTE_EXPIRATION`

ĐIỀU KHIỂN HÀNH VI BỘ TỐI ƯU HÓA (tiếp)

- OPTIMIZER_INDEX_CACHING
- OPTIMIZER_INDEX_COST_ADJ
- OPTIMIZER_FEATURES_ENABLED
- OPTIMIZER_MODE: ALL_ROWS, FIRST_ROWS, FIRST_ROWS_n
- OPTIMIZER_CAPTURE_SQL_PLAN_BASELINES
- OPTIMIZER_USE_SQL_PLAN_BASELINES
- OPTIMIZER_DYNAMIC_SAMPLING
- OPTIMIZER_USE_INVISIBLE_INDEXES
- OPTIMIZER_USE_PENDING_STATISTICS

**XIN CẢM ƠN THẦY VÀ CÁC BẠN
ĐÃ LẮNG NGHE!**