COEN 380
ADV. DATABASE SYSTEMS

GROUP1

SHUBHI AGARWAL
NIVEDITA RAO
KARAN MADHWANI
KAPIL VARMA
NILAY PATEL

AGENDA

- Dataset
  - Populating data in Oracle and Hive
- Queries
  - Query plans :- Oracle
  - Runtime comparison :- Oracle v/s Hive
- Comparative Analysis
- Challenges faced
- Conclusion

DATA INSERTION

- To insert data
  - We Converted the data excel file to .csv file.
- For Hive :
  - After creating tables , we loaded the respective csv file with the following query :-
    - Load data local inpath '/home/adbteam01/documents/Performers.csv' overwrite into table performers;
    - Same goes for movies.csv, Classification.csv and cast.csv
- For Oracle :
  - We created tables and than imported the csv file using the import function provided by Oracle sql developer tool.

DATASET

6 Tables:

- MOVIES
- PERFORMER
- WORKSHOP
- DESIGNATION
- CLASSIFICATION
- CAST

TABLES

MOVIES
-film_id -title -year -workshop -Prc -cat -awards

PERFORMER
-performer_id -downstart -downend -birthname -firstname -gender -dateofbirth -dateofdeath -type -origin

WORKSHOP
-workshop_id -workshopname

CAST
-film_id -Title -designation_id -performer_id

DESIGNATION
-desg_id -desgname

CLASSIFICATION
-clcode - class_name

SYSTEM CONFIGURATION

- Oracle DB (Personal system)
  - Version - Oracle DB 12c
  - 16 GB RAM
  - 4 Cores CPU

- Hadoop Hive (SCU Design Center)
  - 24 worker Nodes
  - 96 cores Processors
  - 768GB RAM
  - NameNode, Secondary NameNode, Worker Nodes
  - 4 cores
  - 32GB RAM

QUERY 1: JOIN

Aim: Get all the movies where movies are classified as Actn

Query

```
SELECT title, year
FROM movies m
JOIN classification c ON (c.ctcode = 'Actn' and m.cat='Actn');
```

QUERY 1: ORACLE RESULT

- Fetched Rows: 5
- Time taken: 0.001 seconds

```
20  Select title,year from movies m join classification c
21  ON (c.ctcode = 'Actn' and m.cat='Actn');
```

Script Output | Query Result

SQL | All Rows Fetched: 5 in 0.001 seconds

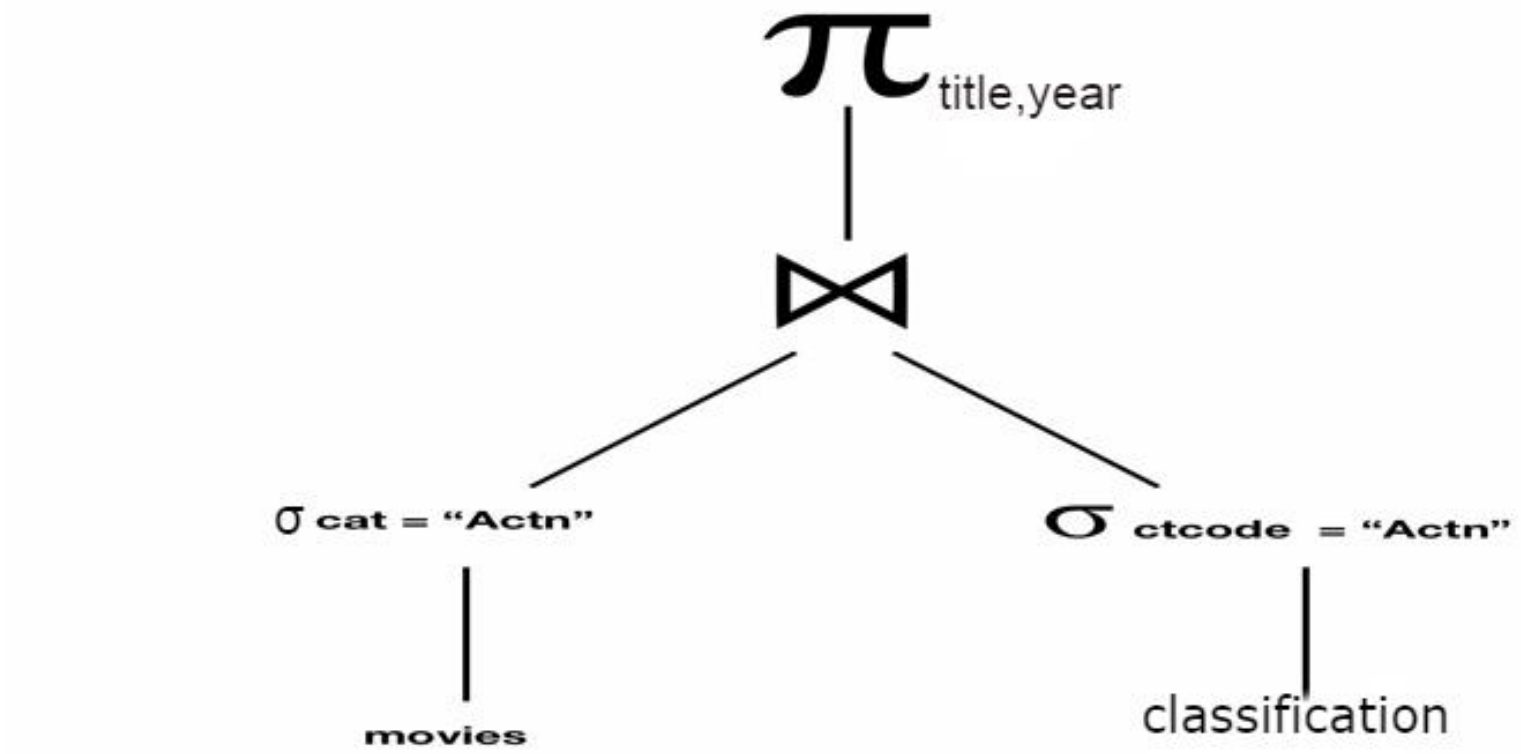
	TITLE	YEAR
1	T:The Three Musketeers	1984
2	T:L'assissinatat du Duc de Guise	1916
3	T:Ivanhoe	1967
4	T:Beau Ideal	1941
5	T:Sands of Iwo Jima	1906

QUERY 1: HIVE RESULT

- Fetched Rows: 5
- Time taken: 49.172 seconds

```
Total MapReduce CPU Time Spent: 32 seconds 420 msec
OK
T:Ivanhoe       1967
T:The Three Musketeers  1984
T:Beau Ideal    1941
T:L''assissinatat du Duc de Guise       1916
T:Sands of Iwo Jima     1906
Time taken: 49.172 seconds, Fetched: 5 row(s)
hive>
```

PROPOSED QUERY PLAN



QUERY 1 - ORACLE EXECUTION PLAN

```
PLAN_TABLE_OUTPUT
SQL_ID  ar5n6maallymh, child number 0
-------------------------------------
Select title,year from movies m join classification c  ON (c.ctcode =
'Actn' and m.cat='Actn')

Plan hash value: 2488223273

-------------------------------------------------------------------------------
| Id  | Operation          | Name    | Rows  | Bytes | Cost (%CPU)| Time     |
-------------------------------------------------------------------------------
|   0 | SELECT STATEMENT   |         |       |       |     5 (100)|          |
|   1 |  NESTED LOOPS      |         |     5 |   170 |     5   (0)| 00:00:01 |
|*  2 |   INDEX UNIQUE SCAN| CTC_PK  |     1 |     5 |     0   (0)|          |
|*  3 |   TABLE ACCESS FULL| MOVIES  |     5 |   145 |     5   (0)| 00:00:01 |
-------------------------------------------------------------------------------
```

QUERY 2: SELECT ALL

Aim: Get all Records from Cast table

Query

```
SELECT * FROM CASTS;
```

QUERY 2: ORACLE RESULT

- Fetched Rows: 2488
- Time taken: 0.098 seconds

```
26
27  select * from casts;
28
```

Script Output × | Query Result ×

SQL | All Rows Fetched: 2488 in 0.098 seconds

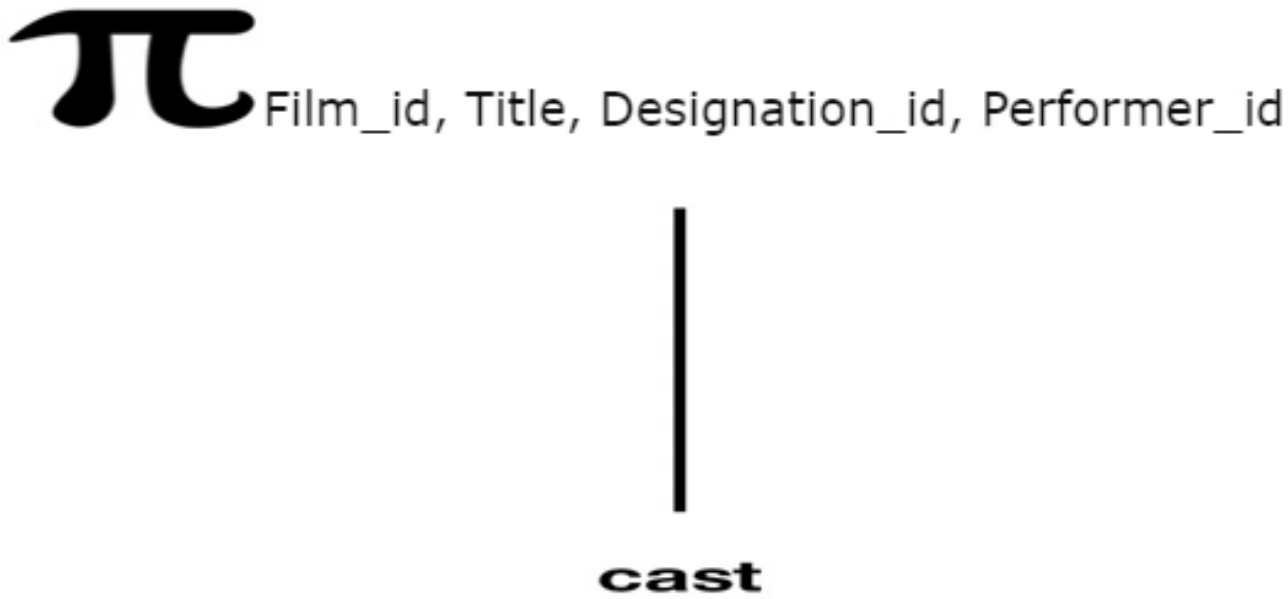
	FILM_ID	TITLE	DESIGNATION_ID	PERFORMER_ID
1	H1	T:Always Tell Your Wife	3	37
2	H1	T:Always Tell Your Wife	2	391
3	H1	T:Always Tell Your Wife	1	428
4	H1	T:Always Tell Your Wife	2	83
5	H1	T:Always Tell Your Wife	1	257
6	H2	T:Number Thirteen	1	301
7	H2	T:Number Thirteen	3	531
8	H2	T:Number Thirteen	3	105
9	H2	T:Number Thirteen	2	340
10	H3	T:Woman to Woman	3	24
11	H3	T:Woman to Woman	3	227
12	H3	T:Woman to Woman	1	406
13	H3	T:Woman to Woman	2	617

QUERY 2: HIVE RESULT

- Fetched Rows: 2488
- Time taken: 0.079 seconds

```
ASa50   T:Jack London   2       109
ASa50   T:Jack London   1       573
ASa50   T:Jack London   1       178
ASa50   T:Jack London   3       454
ASa51   T:The Hairy Ape 2       695
ASa51   T:The Hairy Ape 3       72
ASa51   T:The Hairy Ape 1       431
ECa10   T:The Final Judgement   1       272
ECa10   T:The Final Judgement   2       176
ECa10   T:The Final Judgement   1       575
ECa10   T:The Final Judgement   3       696
ECa10   T:The Final Judgement   2       564
ECa10   T:The Final Judgement   1       229
ECa10   T:The Final Judgement   3       104
ECa20   T:The trail to Yesterday        3       463
ECa20   T:The trail to Yesterday        3       278
ECa20   T:The trail to Yesterday        2       51
Time taken: 0.079 seconds, Fetched: 2488 row(s)
hive>
```

PROPOSED QUERY PLAN



QUERY 2 - ORACLE EXECUTION PLAN

```
SQL_ID  9r7jdv6m9vr6h, child number 0
-------------------------------------
select * from casts

Plan hash value: 1755451814

-------------------------------------------------------------------------------
| Id  | Operation         | Name  | Rows  | Bytes | Cost (%CPU)| Time     |
-------------------------------------------------------------------------------
|   0 | SELECT STATEMENT  |       |       |       |     6 (100)|          |
|   1 |  TABLE ACCESS FULL| CASTS |  2488 | 77128 |     6   (0)| 00:00:01 |
-------------------------------------------------------------------------------
```

QUERY 3: UNIQUE RECORDS FROM TABLE: COUNT

Aim: Get Number of unique names by concatenation of Birthname and Firstname

Query

```
SELECT COUNT(DISTINCT CONCAT(birthname,firstname))
FROM performer;
```

QUERY 3: ORACLE RESULT

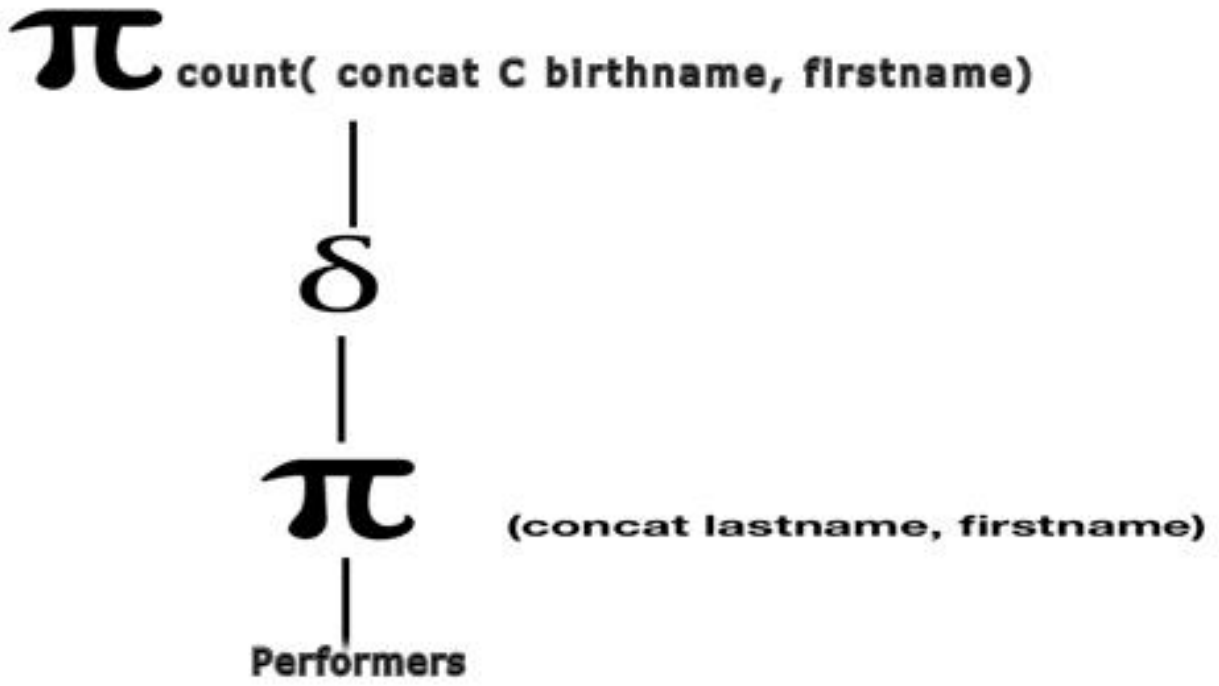
- Fetched Rows: 1
- Time taken: 0.003 seconds

QUERY 3: HIVE RESULT

- Fetched Rows: 1
- Time taken: 4.86 seconds

```
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 4.86 sec   HDFS Rea
Total MapReduce CPU Time Spent: 4 seconds 860 msec
OK
482
Time taken: 26.651 seconds, Fetched: 1 row(s)
```

PROPOSED QUERY PLAN



QUERY 3 - ORACLE EXECUTION PLAN

```
PLAN_TABLE_OUTPUT
SQL_ID  b217nk9cycpxq, child number 0
-------------------------------------
SELECT COUNT(DISTINCT CONCAT(birthname,firstname)) FROM performer

Plan hash value: 1207736318

-------------------------------------------------------------------------------
| Id  | Operation            | Name       | Rows  | Bytes | Cost (%CPU)| Time     |
-------------------------------------------------------------------------------
|   0 | SELECT STATEMENT     |            |       |       |     5 (100)|          |
|   1 |  SORT AGGREGATE      |            |     1 |   102 |            |          |
|   2 |   VIEW               | VW_DAG_0   |   699 | 71298 |     5  (20)| 00:00:01 |
|   3 |    HASH GROUP BY     |            |   699 | 10485 |     5  (20)| 00:00:01 |
|   4 |     TABLE ACCESS FULL| PERFORMER  |   699 | 10485 |     4   (0)| 00:00:01 |
-------------------------------------------------------------------------------
```

QUERY 4: AGGREGATION

Aim: Count number of specific role type for a specific performer

Query

```
SELECT COUNT(DISTINCT(a.type)) FROM performer a
INNER JOIN casts c ON a.performer_id = c.performer_id
INNER JOIN movies m ON m.film_id = c.film_id and m.cat = 'Susp'
AND c.performer_id = 256;
```

QUERY 4: ORACLE RESULT

- Fetched Rows: 1
- Time taken: 0.005 seconds

```
--Query 4: Inner Join Unique
SELECT COUNT(DISTINCT(a.type))
from performer a
INNER JOIN casts c ON a.performer_id = c.performer_id
INNER JOIN movies m ON m.film_id = c.film_id and m.cat = 'Susp'
AND c.performer_id = 256;
```

Query Result

SQL | All Rows Fetched: 1 in 0.005 seconds

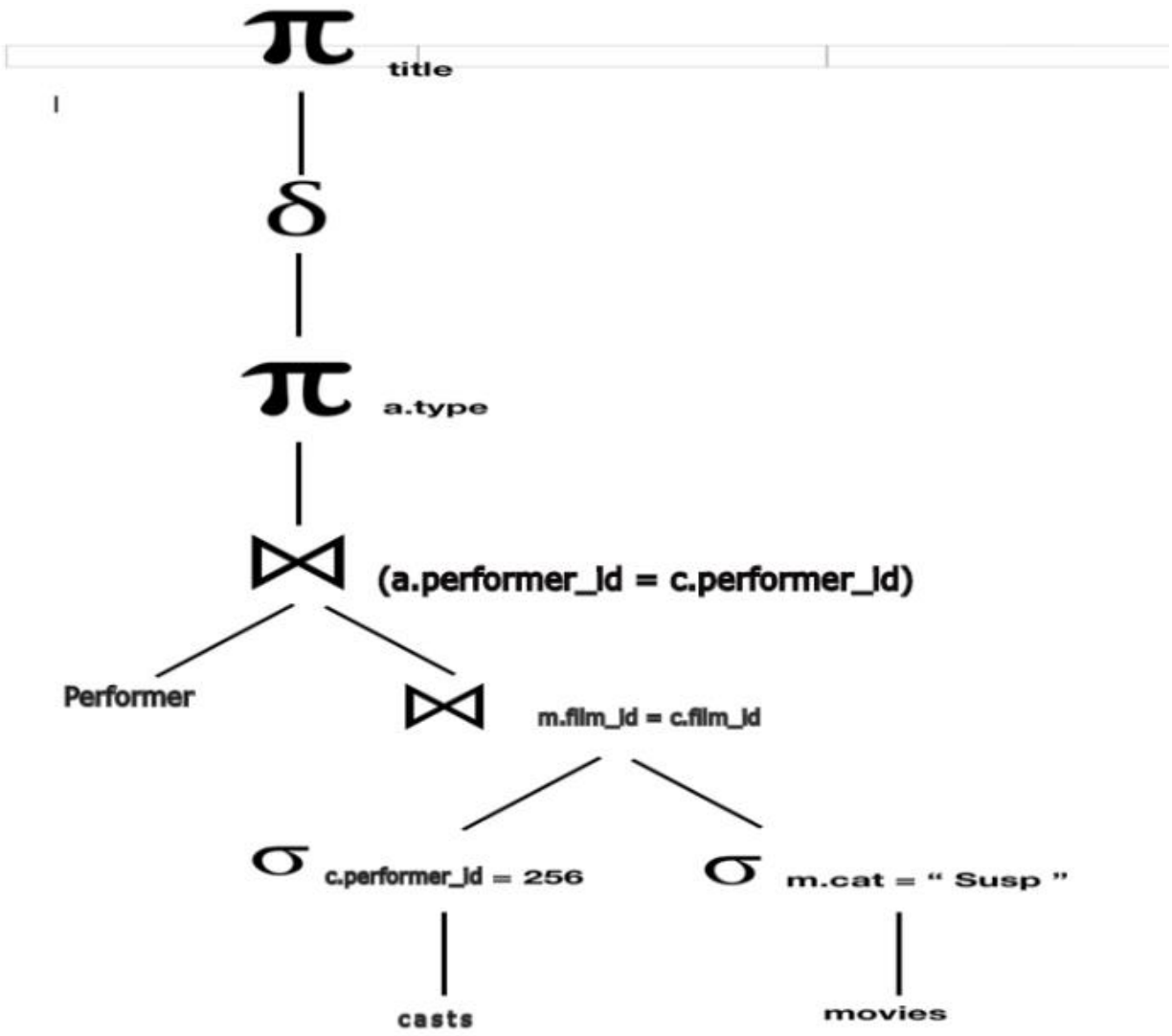
	COUNT(DISTINCT(A.TYPE))
1	1

QUERY 4: HIVE RESULT

- Fetched Rows: 1
- Time taken: 53.109 seconds

```
MapReduce Total cumulative CPU time: 2 seconds 230 msec
Ended Job = job_1574050829610_0574
MapReduce Jobs Launched:
Stage-Stage-3: Map: 1  Reduce: 1   Cumulative CPU: 2.23 sec   HDFS Read: 94193 HDFS Write: 2 SUCCESS
Total MapReduce CPU Time Spent: 2 seconds 230 msec
OK
1
Time taken: 53.109 seconds, Fetched: 1 row(s)
hive>
```

PROPOSED QUERY PLAN



QUERY 4 - ORACLE EXECUTION PLAN

```
PLAN_TABLE_OUTPUT
SELECT COUNT(DISTINCT(a.type)) from performer a INNER JOIN casts c ON
a.performer_id = c.performer_id INNER JOIN movies m ON m.film_id =
c.film_id and m.cat = 'Susp' AND c.performer_id = 256

Plan hash value: 360332323

-------------------------------------------------------------------------------------------
| Id  | Operation                     | Name       | Rows  | Bytes | Cost (%CPU)| Time     |
-------------------------------------------------------------------------------------------
|   0 | SELECT STATEMENT              |            |       |       |    14 (100)|          |
|   1 |  SORT AGGREGATE               |            |     1 |    52 |            |          |
|   2 |   VIEW                        | VW_DAG_0   |     4 |   208 |    14   (8)| 00:00:01 |
|   3 |    HASH GROUP BY              |            |     4 |   156 |    14   (8)| 00:00:01 |
|   4 |     NESTED LOOPS              |            |     4 |   156 |    13   (0)| 00:00:01 |
|   5 |      NESTED LOOPS             |            |     4 |   156 |    13   (0)| 00:00:01 |
|*  6 |       HASH JOIN SEMI          |            |     4 |    84 |     9   (0)| 00:00:01 |
|*  7 |        TABLE ACCESS FULL      | CASTS      |     4 |    40 |     6   (0)| 00:00:01 |
|*  8 |        TABLE ACCESS FULL      | MOVIES     |    64 |   704 |     3   (0)| 00:00:01 |
|*  9 |       INDEX UNIQUE SCAN       | ACTOR_PK   |     1 |       |     0   (0)|          |
|  10 |      TABLE ACCESS BY INDEX ROWID| PERFORMER |     1 |    18 |     1   (0)| 00:00:01 |
-------------------------------------------------------------------------------------------
```

QUERY 5: GROUP BY STATEMENT

Aim: Select count of movies under each category

Query

```
SELECT class_name AS category, categorycount
FROM (SELECT cat, count(*) AS categorycount
FROM movies GROUP BY cat) A
INNER JOIN (SELECT * FROM CLASSIFICATION) B
ON A.cat = B.ctcode;
```

QUERY 5: ORACLE RESULT

- Fetched Rows: 10
- Time taken: 0.002 seconds

```
SELECT class_name as category, categorycount
FROM (SELECT cat, count(*) as categorycount
FROM movies GROUP BY cat) A
INNER JOIN (SELECT * FROM CLASSIFICATION) B
ON A.cat = B.ctcode;
```

Query Result

SQL | All Rows Fetched: 10 in 0.002 seconds

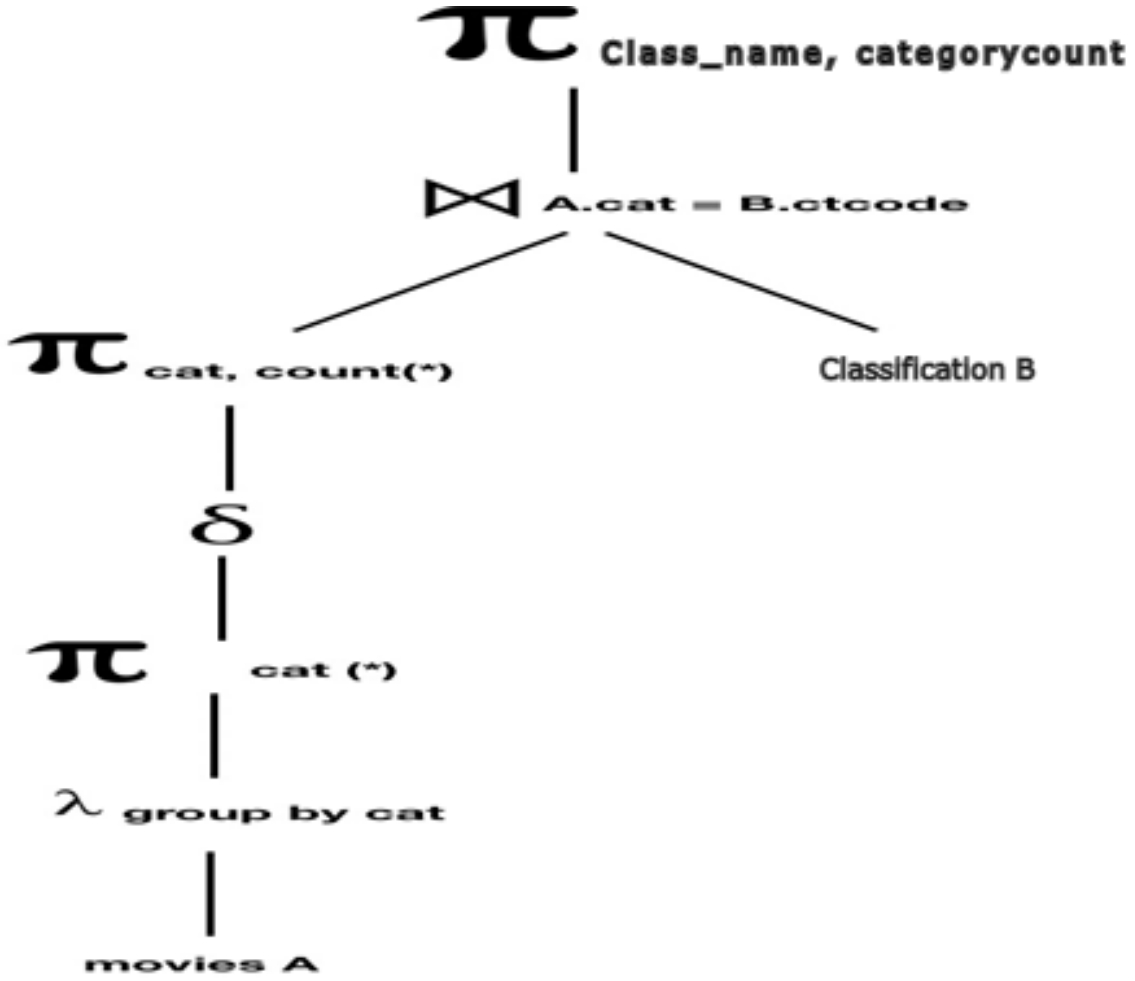
	CATEGORY	CATEGORYCOUNT
1	Action	6
2	Adventure	15
3	Biopic	9
4	Western	26
5	Suspense	64
6	Sci-Fi	2
7	Romantic	34
8	Mystery	9
9	Comedy	199
10	Drama	132

QUERY 5: HIVE RESULT

- Fetched Rows: 10
- Time taken: 60.992 seconds

```
Total MapReduce CPU Time Spent: 9 seconds 900 msec
OK
Action  5
Adventure       14
Biopic  9
Comedy  197
Drama   131
Mystery 9
Romantic        34
Sci-Fi  2
Suspense        46
Western 25
Time taken: 60.992 seconds, Fetched: 10 row(s)
```

PROPOSED QUERY PLAN



QUERY 5 - ORACLE EXECUTION PLAN

```
SELECT class_name as category, categorycount  FROM (SELECT cat,
count(*) as categorycount  FROM movies GROUP BY cat) A  INNER JOIN
(SELECT * FROM CLASSIFICATION) B  ON A.cat = B.ctcode

Plan hash value: 1344490878

-------------------------------------------------------------------------------
| Id  | Operation            | Name           | Rows  | Bytes | Cost (%CPU)| Time     |
-------------------------------------------------------------------------------
|   0 | SELECT STATEMENT     |                |       |       |     6 (100)|          |
|*  1 |  HASH JOIN           |                |    10 |   780 |     6  (17)| 00:00:01 |
|   2 |   VIEW               |                |    10 |   650 |     4  (25)| 00:00:01 |
|   3 |    HASH GROUP BY     |                |    10 |    50 |     4  (25)| 00:00:01 |
|   4 |     TABLE ACCESS FULL| MOVIES         |   496 |  2480 |     3   (0)| 00:00:01 |
|   5 |   TABLE ACCESS FULL  | CLASSIFICATION |    11 |   143 |     2   (0)| 00:00:01 |
-------------------------------------------------------------------------------
```

QUERY 6: INNER JOIN

Aim: Finding cast of a movie

Query

```
SELECT a.firstname, c.title
FROM performer a INNER JOIN casts c
ON (c.performer_id = a.performer_id)
WHERE c.film_id
in (SELECT m.film_id FROM movies m WHERE m.title = 'T:Number   Thirteen');
```

QUERY 6: ORACLE RESULT

- Fetched Rows: 4
- Time taken: 0.007 seconds

```
SELECT a.firstname, c.title
FROM performer a INNER JOIN casts c
ON (c.performer_id = a.performer_id)
WHERE c.film_id in (SELECT m.film_id FROM movies m WHERE m.title = 'T:Number Thirteen');
```

Query Result

SQL | All Rows Fetched: 4 in 0.007 seconds

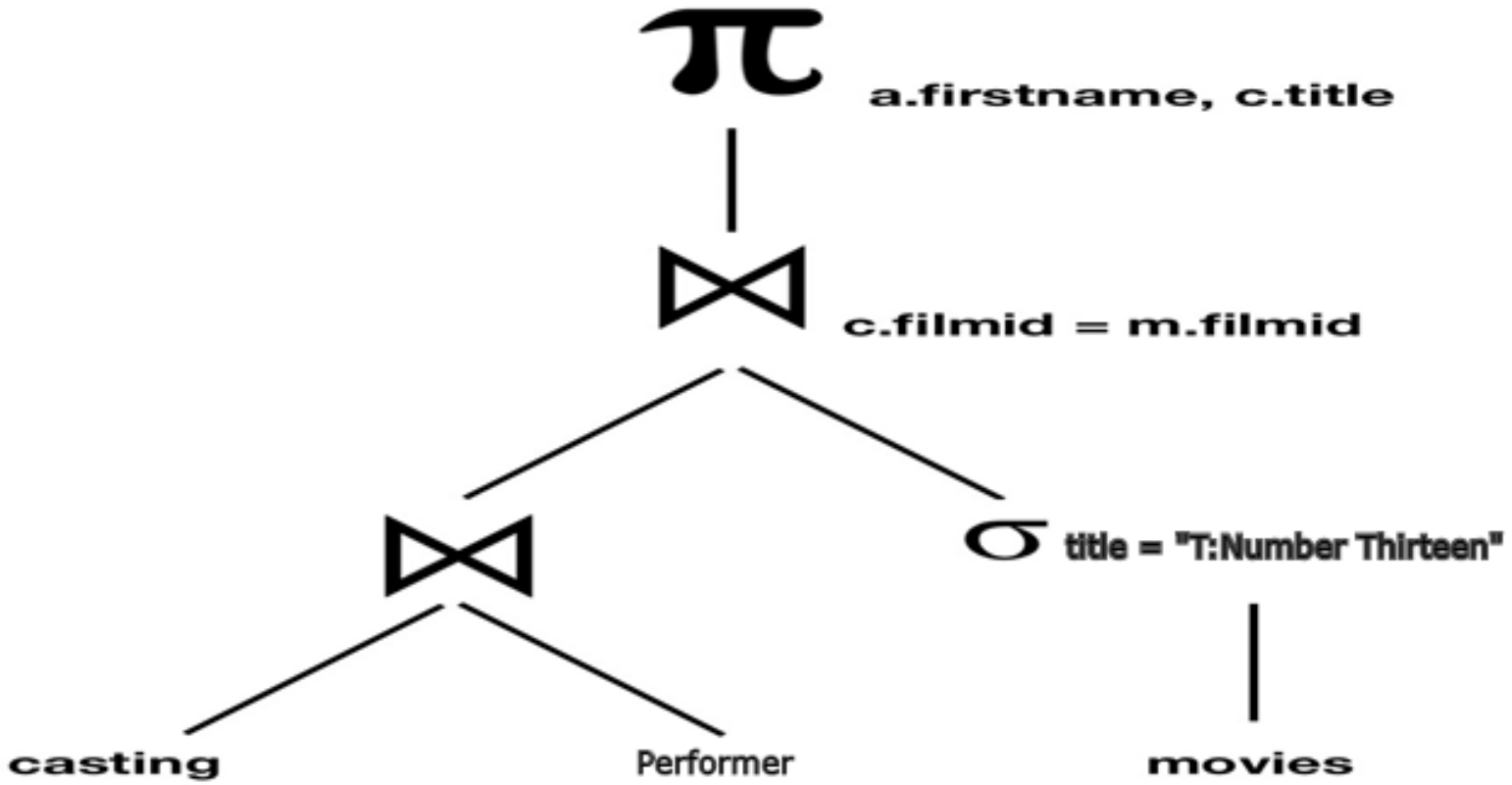
	FIRSTNAME	TITLE
1	Patrick	T:Number Thirteen
2	Joss	T:Number Thirteen
3	William	T:Number Thirteen
4	Walter	T:Number Thirteen

QUERY 6: HIVE RESULT

- Fetched Rows: 4
- Time taken: 51.162 seconds

```
Total MapReduce CPU Time Spent: 1 seconds 380 msec
OK
William T:Number Thirteen
Patrick T:Number Thirteen
Walter  T:Number Thirteen
Joss    T:Number Thirteen
Time taken: 51.162 seconds, Fetched: 4 row(s)
hive>
```

PROPOSED QUERY PLAN



QUERY 6 - ORACLE EXECUTION PLAN

```
SELECT a.firstname, c.title  FROM performer a INNER JOIN casts c  ON
(c.performer_id = a.performer_id) WHERE c.film_id in (SELECT m.film_id
FROM movies m WHERE m.title = 'T:Number Thirteen')

Plan hash value: 1184942852

-------------------------------------------------------------------------------
| Id  | Operation             | Name      | Rows  | Bytes | Cost (%CPU)| Time     |
-------------------------------------------------------------------------------
|   0 | SELECT STATEMENT      |           |       |       |    13 (100)|          |
|*  1 |  HASH JOIN            |           |     5 |   335 |    13   (0)| 00:00:01 |
|   2 |   MERGE JOIN CARTESIAN|           |   699 | 25863 |     7   (0)| 00:00:01 |
|*  3 |    TABLE ACCESS FULL  | MOVIES    |     1 |    26 |     3   (0)| 00:00:01 |
|   4 |    BUFFER SORT        |           |   699 |  7689 |     4   (0)| 00:00:01 |
|   5 |     TABLE ACCESS FULL | PERFORMER |   699 |  7689 |     4   (0)| 00:00:01 |
|   6 |   TABLE ACCESS FULL   | CASTS     |  2488 | 74640 |     6   (0)| 00:00:01 |
-------------------------------------------------------------------------------
```

QUERY 7: SUBQUERY

Aim: Get all Records from Cast table

Query

```
SELECT performer_id
FROM (SELECT performer_id,row_number() over (order by count(*) desc) AS rn
FROM casts GROUP BY performer_id) WHERE rn = 3;
```

QUERY 7: ORACLE RESULT

- Fetched Rows: 1
- Time taken: 0.006 seconds

```
63  -- Query 7: Subquery
64  SELECT performer_id FROM
65  (SELECT performer_id,row_number() over (order by count(*) desc) as rn
66    from casts group by performer_id) where rn = 3;
67
```

Script Output | Query Result

SQL | All Rows Fetched: 1 in 0.006 seconds

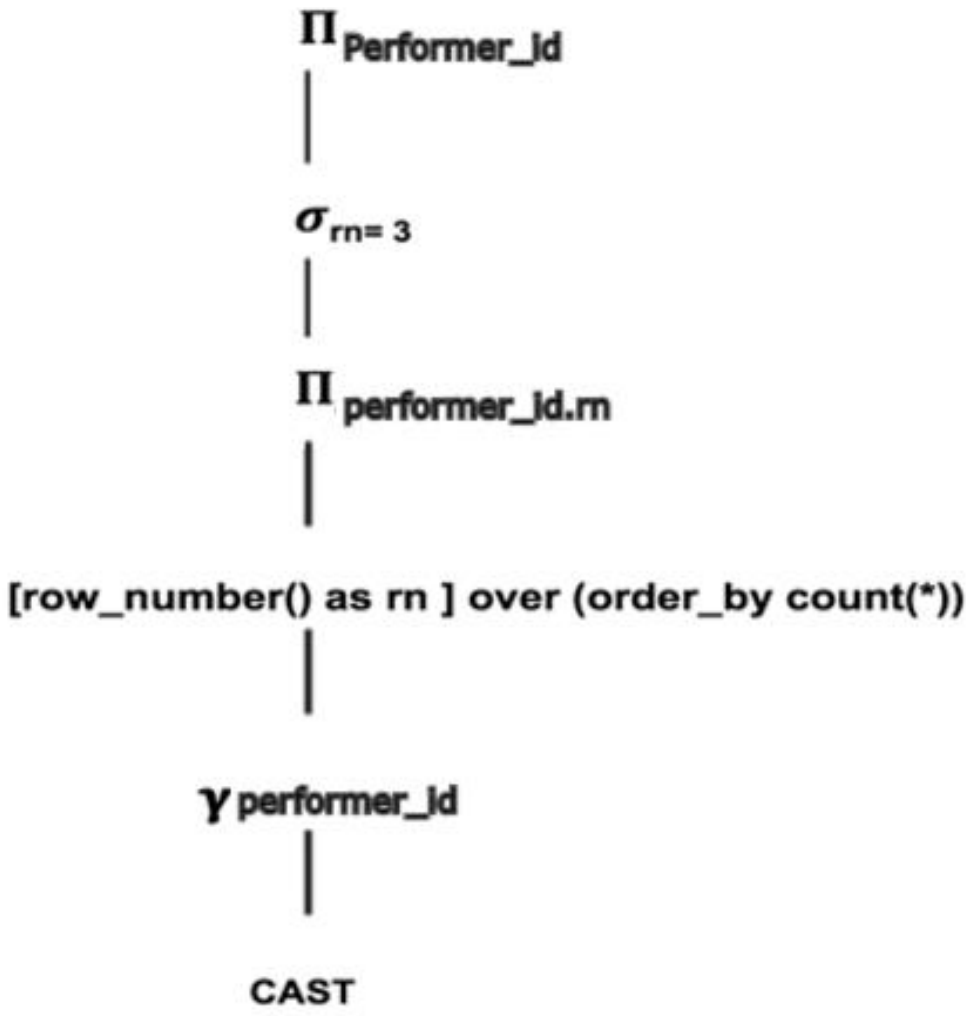
	PERFORMER_ID
1	397

QUERY 7: HIVE RESULT

- Hive does not support subqueries:

```
        at org.apache.hadoop.util.RunJar.run(RunJar.java:221)
        at org.apache.hadoop.util.RunJar.main(RunJar.java:136)
FAILED: ParseException line 1:130 cannot recognize input near 'where' 'rn' '=' in subquery source
hive>
```

PROPOSED QUERY PLAN



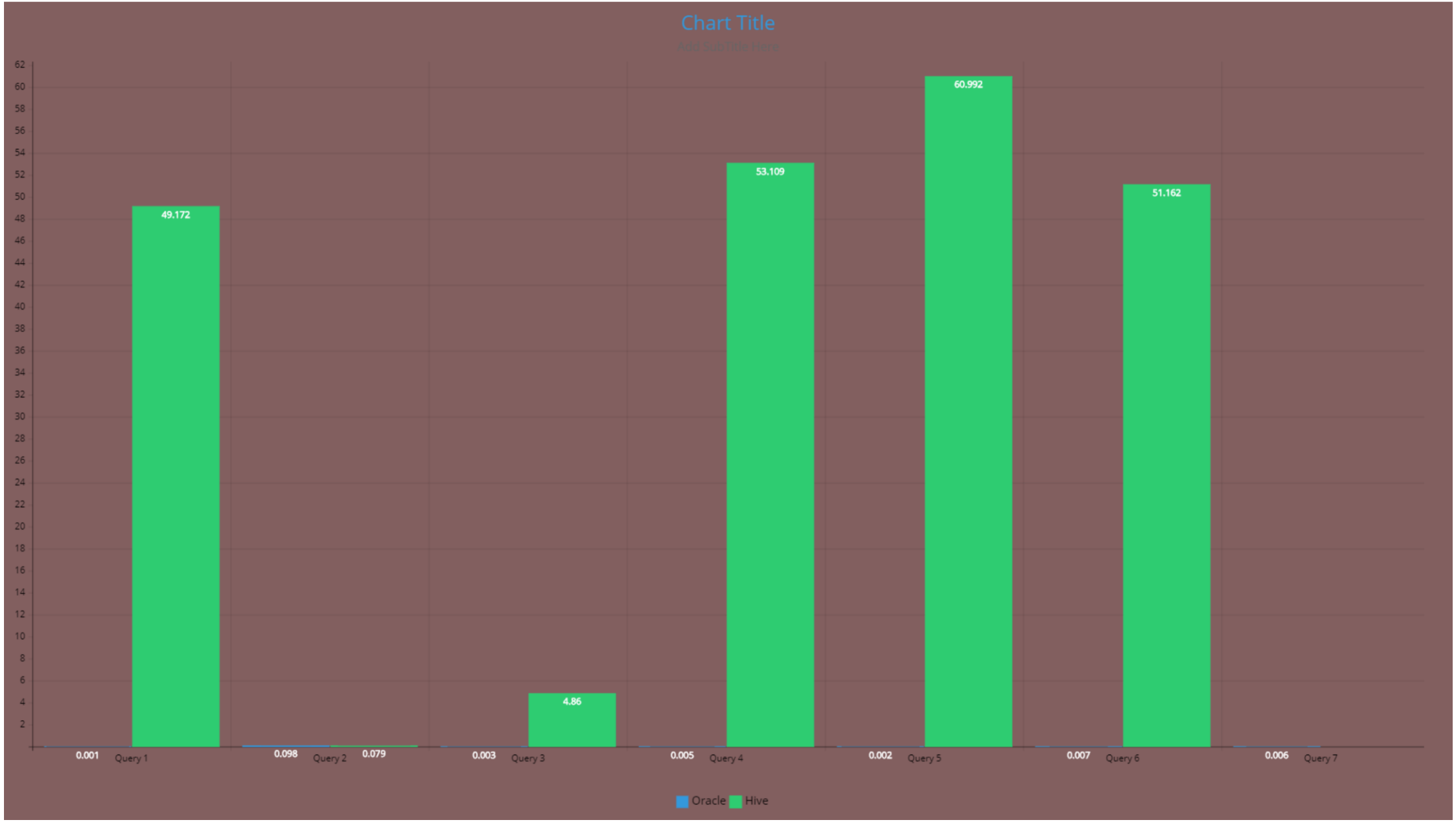
QUERY 7 - ORACLE EXECUTION PLAN

```
SQL_ID  8synca5j4hpag, child number 0
-------------------------------------
SELECT performer_id FROM  (SELECT performer_id,row_number() over (order
by count(*) desc) as rn    from casts group by performer_id) where rn =
3

Plan hash value: 1283376857

-------------------------------------------------------------------------------
| Id  | Operation                | Name  | Rows  | Bytes | Cost (%CPU)| Time     |
-------------------------------------------------------------------------------
|   0 | SELECT STATEMENT         |       |       |       |     8 (100)|          |
|*  1 |  VIEW                    |       |     3 |   195 |     8  (25)| 00:00:01 |
|*  2 |   WINDOW SORT PUSHED RANK|       |   677 |  2708 |     8  (25)| 00:00:01 |
|   3 |    HASH GROUP BY         |       |   677 |  2708 |     8  (25)| 00:00:01 |
|   4 |     TABLE ACCESS FULL    | CASTS |  2488 |  9952 |     6   (0)| 00:00:01 |
-------------------------------------------------------------------------------
```

PERFORMANCE EVALUATION



CHALLENGES:

- Starting with Hive
- Proposed query plan creation was difficult especially for complicated queries in comparison to Oracle Query Plan

CONCLUSION

- Hive takes a heavy blow during join queries because of the index-non supporting architecture
- Hive can be used better for OLAP(On-Line analytical Processing)
- Oracle on the other hand aced through the joins and is very effective to be used for OLTP(On-Line transaction Processing)
- Hive doesn't support subqueries unless used with a from clause