Filter Pushdown vs Column Pruning vs Data Skipping

Karthik Kondpak
9989454737

Day 3 — Spark Optimization Topic

3. Filter Pushdown vs Column Pruning vs Data Skipping

(3powerful optimizations everyPySparkdeveloper must master)

These three optimizations reduce I/O and speed up your Spark jobs, but each one works differently.

This is one of the most asked topics in 20–40 LPA interviews.

1. Filter Pushdown

Spark pushesfilters downtothe storage engine (Parquet/Delta) so **only matching rows are read**.

Example

```
df = spark.read.parquet("/delta/sales") result =
df.filter("state = 'Karnataka'")
```

What happens internally

- Parquet/Delta reads only row groups for Karnataka
- Less data scanned
- Smaller shuffle

- Faster processing

Best Formats

Parquet, ORC, Delta

(Not effective for CSV/JSON)

2. Column Pruning

Spark reads **only the required columns**, not the entire file.

Example

```
df = spark.read.parquet("/delta/users")
selected = df.select("user_id", "city")
```

Benefits

- Saves massive I/O
- Reduces memory footprint
- Reduces shuffle and CPU usage
- Useful for tables with 100+ columns

3. Data Skipping

Delta Lake stores **min-max statistics** for each column in each file.

During query execution, Spark **skips entire files** that cannot match the filter condition.

Example

Dataset file stats:

File	amount_min	amount_max
F1	10	500
F2	501	900
F3	900	1500

Query:

```
df.filter("amount > 1000")
```

Delta will skip F1 and F2 → read only F3.

This is a major optimization for 1B+ row tables.

Scenario — Flipkart Orders

Dataset: /delta/flipkart_orders

Rows: 450M

Columns: 45

Partitioned by: city

Query

```
df =
spark.read.format("delta").load("/delta/zoma
to_orders")

result = df.select("order_id", "city",
"payment_type")
```

What happens:

1. Partition pruning → Reads only Delhi partitions
2. Filter pushdown → Reads only rows for Delhi
3. Column pruning → Reads only order_id and amount
4. Data skipping → Delta skips files where city or amount cannot match

This reduces the scan from **450M rows → a few million**.

Let's build your Data Engineering journey together!

Call us directly at: 9989454737

https://seekhobigdata.com/