Nagarro Data Engineer Interview Guide – Experienced 3+

Round 1 & 2 – Technical

1. **Introduce Yourself**

   Begin with a concise summary of your professional background, emphasizing key experiences, tools, and projects related to data engineering. Highlight your expertise in big data frameworks, cloud platforms, and programming languages.

   Example:
   "I am a Data Engineer with over 5 years of experience designing and implementing scalable data pipelines using Spark, Python, and Azure. I've worked extensively with batch and streaming data solutions, focusing on performance optimization and cost-efficiency for large-scale enterprise data platforms."

2. **Explain Your Project Architecture**

   Describe the high-level architecture, data flow, and key components. Example for a cloud-based data pipeline:

   - Data Ingestion: Using Kafka and Azure Data Factory.
   - Data Processing: Utilizing Spark on Databricks.
   - Storage: Data lakes on Azure Blob Storage and structured data in Delta Lake.
   - Reporting: Power BI for dashboards. Explain key design choices, challenges faced, and optimizations implemented.

3. **How to Upsert Your Data Daily Using Spark**

   Upsert (update + insert) in Spark can be achieved using merge statements with Delta Lake or manually combining dataframes:

```
from delta.tables import DeltaTable

delta_table = DeltaTable.forPath(spark, "/path/to/delta")
delta_table.alias("target").merge(
    updates.alias("source"),
    "target.id = source.id"
).whenMatchedUpdate(set={"name": "source.name", "age": "source.age"})
.whenNotMatchedInsert(values={"id": "source.id", "name": "source.name", "age": "source.age"})
.execute()
```

© Shubham Wadekar

For personal use only. Redistribution or resale is prohibited. © Shubham Wadekar

4. **How to Perform SCD Type 3 Using Spark**
   Type 3 SCD tracks only the current and previous values.

```
from pyspark.sql import functions as F

df = df.withColumn("previous_value", F.col("current_value"))

df = df.withColumn("current_value", F.when(condition,
new_value).otherwise(F.col("current_value")))
```

5. **What is Shuffle and How to Handle It in Spark**
   Shuffle refers to redistributing data across partitions, causing costly operations involving disk I/O and network transfers. Occurs during wide transformations (e.g., groupByKey, join).

   Optimizations:

   - Use reduceByKey instead of groupByKey.
   - Apply salting for skewed keys.
   - Enable Adaptive Query Execution (AQE) to dynamically optimize shuffle partitions.

6. **What is Broadcast Join and Why is It Required?**

   A broadcast join sends a small dataset to all worker nodes to avoid shuffle. It improves performance for joins between a large and small table.

```
small_df = spark.read.parquet("small_table").broadcast()

large_df = spark.read.parquet("large_table")

result = large_df.join(small_df, "key_column")
```

7. **What is Predicate Pushdown and AQE with Example**
   Predicate pushdown reduces data read by applying filters at the data source level.

```
df = spark.read.option("pushDownPredicate", True).parquet("/data").filter("col > 100")
```

AQE dynamically adjusts partitions and joins at runtime to optimize performance.

8. **PySpark Code for Broadcast Join and Conditional Aggregation by Location**

```
from pyspark.sql import functions as F

result = df.join(broadcast(dim_table), "key")
result.groupBy("location").agg(F.max(F.avg("salary"))).show()
```

© Shubham Wadekar

For personal use only. Redistribution or resale is prohibited. © Shubham Wadekar

9. **SQL Query for Best of 3 Marks and Average in a Student Table**

```
SELECT student_id, AVG(mark)
FROM (
    SELECT student_id, mark
    FROM student_marks
    ORDER BY mark DESC
    LIMIT 3
) AS top3
GROUP BY student_id;
```

10. **How to Handle Null in Spark** Use fill, drop, or replace functions:

df.fillna({"column_name": 0}).dropna().replace("null_value", "replacement")

11. **SCD Implementation in ETL**
    Use **merge** for Delta tables or a combination of union and filter for traditional tables.

12. **Converting SCD0 to SCD3** Add columns for current and previous values, updating only the most recent records.

13. **Facts and Dimension Tables Properties**

- **Fact Table**: Contains measurements, often with numeric values (e.g., sales, revenue).
- **Dimension Table**: Contains descriptive attributes (e.g., product, date).

14. **Handling Large-Scale Data Ingestion in AWS Pipelines** Use **S3 triggers** with Lambda or Glue jobs. Automate with **Step Functions** for scheduling and retries.

15. **Challenges with Spark Jobs and Resolutions**

- **Memory errors**: Use correct partitioning.
- **Shuffle issues**: Apply salting and broadcast joins.

16. **Data Shuffling Causes and Techniques** Causes: Wide transformations and data skew. Techniques: Repartition, salting, and AQE.

17. **Narrow vs. Wide Transformations**

- **Narrow**: Data from one partition (e.g., map).
- **Wide**: Data from multiple partitions (e.g., groupBy).

© Shubham Wadekar

For personal use only. Redistribution or resale is prohibited. © Shubham Wadekar

18. **ReduceByKey vs. GroupByKey**

- reduceByKey aggregates data in each partition first, reducing shuffle.
- groupByKey sends all data to a single key, causing higher shuffle costs.

19. **Data Volume in Pipelines and Scalability Solutions** Use partitioning, caching, and efficient file formats (like Parquet) for large datasets.

20. **Monitoring and Orchestrating Spark Jobs** Tools: **Airflow, Oozie, or Azure Data Factory**. Use Spark UI to track stages and task execution.

21. **Features of NoSQL Databases**

- Schema flexibility, scalability, and high availability.
- Types: Key-value stores, document stores, column-family stores.

22. **Graph Databases** Designed for handling relationships. Example: Neo4j for social network analysis.

Glassdoor Nagarro Review –

https://www.glassdoor.co.in/Reviews/Nagarro-Reviews-E240077.htm

Nagarro Careers –

https://www.nagarro.com/en/careers

Subscribe to my YouTube Channel for Free Data Engineering Content –

https://www.youtube.com/@shubhamwadekar27

Connect with me here –

https://bento.me/shubhamwadekar

Checkout more Interview Preparation Material on –

https://topmate.io/shubham_wadekar

© Shubham Wadekar

For personal use only. Redistribution or resale is prohibited. © Shubham Wadekar