

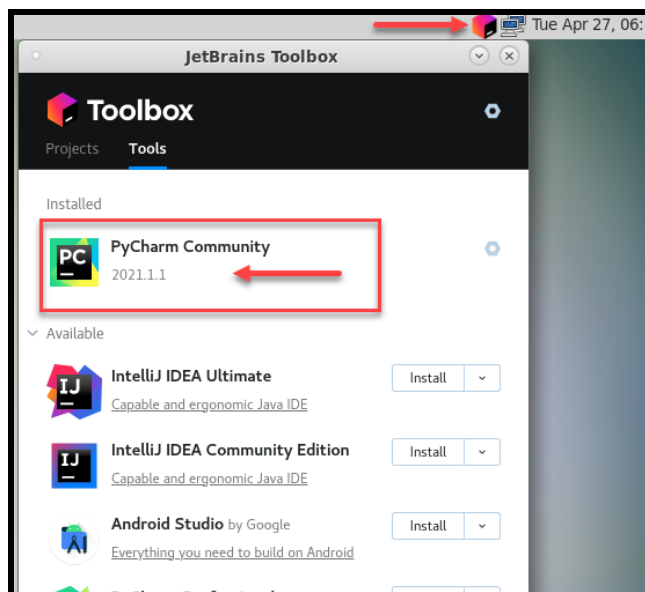
Lab: Implement Broadcast Join

Introduction

This exercise would help you to understand about the internals of the Broadcast Join in Spark. Let's explore it together.

Let's get Started

Run Pycharm using below command



In the existing **JoinDemo** project. Apply the broadcast function to the right side dataframe

```
join_expr = flight_time_df1.id == flight_time_df2.id  
join_df = flight_time_df1.join(broadcast(flight_time_df2), join_expr, "inner")
```

Note: Giving a hint to spark that this dataframe is small enough, and you should consider broadcasting this table.

```
from pyspark.sql.functions import broadcast
```

```
JoinDemo.py x
1 from pyspark.sql import SparkSession
2 from pyspark.sql.functions import broadcast
3
4 from lib.logger import Log4j
5
6 if __name__ == "__main__":
7     spark = SparkSession \
8         .builder \
9         .appName("Join Demo") \
10        .master("local[3]") \
11        .getOrCreate()
12
13     logger = Log4j(spark)
14
15     flight_time_df1 = spark.read.json("data/d1/")
16     flight_time_df2 = spark.read.json("data/d2/")
17
18     spark.conf.set("spark.sql.shuffle.partitions", 3)
19
20     join_expr = flight_time_df1.id == flight_time_df2.id
21     join_df = flight_time_df1.join(broadcast(flight_time_df2), join_expr, "inner")
22
```

Now, Run the program:

```
JoinDemo.py x
1 from pyspark.sql import SparkSession
2 from pyspark.sql.functions import broadcast
3
4 from lib.logger import Log4j
5
6 if __name__ == "__main__":
7     spark = SparkSession \
8         .builder \
9         .appName("Join Demo") \
10        .master("local[3]") \
11        .getOrCreate()
12
13     logger = Log4j(spark)
14
15     flight_time_df1 = spark.read.json("data/d1/")
16     flight_time_df2 = spark.read.json("data/d2/")
17
18     spark.conf.set("spark.sql.shuffle.partitions", 3)
19
20     join_expr = flight_time_df1.id == flight_time_df2.id
21     join_df = flight_time_df1.join(broadcast(flight_time_df2), join_expr, "inner")
22
```



Let's go to the **Spark UI** by visiting the <http://localhost:4040>

spark 3.1.1

JobsStagesStorageEnvironmentExecutorsSQL

Shuffle Join Demo application UI

Spark Jobs (?)

User: training
Total Uptime: 1.7 min
Scheduling Mode: FIFO
Completed Jobs: 4

▶ Event Timeline

▼ Completed Jobs (4)

Page: 1 1 Pages. Jump to 1 . Show 100 items in a page. Go

Job Id (Job Group) ▼	Description	Submitted	Duration	Stages: Succeeded/Total	Tasks (for all stages): Succeeded/Total
3	collect at /home/training/Documents/Spark-Programming-In-Python-master/19-ShuffleJoinDemo/SuffleJoinDemo.py:23	2021/03/19 14:48:52	1 s	1/1	3/3
2 (cfa5a4fb-0704-4bc6-8668-cfa4a950f3b6)	broadcast exchange (runId cfa5a4fb-0704-4bc6-8668-cfa4a950f3b6) collect at /home/training/Documents/Spark-Programming-In-Python-master/19-ShuffleJoinDemo/SuffleJoinDemo.py:23	2021/03/19 14:48:50	1 s	1/1	3/3
1	json at NativeMethodAccessorImpl.java:0 json at NativeMethodAccessorImpl.java:0	2021/03/19 14:48:49	0.6 s	1/1	3/3
0	json at NativeMethodAccessorImpl.java:0 json at NativeMethodAccessorImpl.java:0	2021/03/19 14:48:47	2 s	1/1	3/3

Page: 1 1 Pages. Jump to 1 . Show 100 items in a page. Go

spark 3.1.1

JobsStagesStorageEnvironmentExecutorsSQL

Shuffle Join Demo application UI

Stages for All Jobs

Completed Stages: 4

▼ Completed Stages (4)

Page: 1 1 Pages. Jump to 1 . Show 100 items in a page. Go

Stage Id ▼	Description	Submitted	Duration	Tasks: Succeeded/Total	Input	Output	Shuffle Read	Shuffle Write
3	collect at /home/training/Documents/Spark-Programming-In-Python-master/19-ShuffleJoinDemo/SuffleJoinDemo.py:23 +details	2021/03/19 14:48:52	1 s	3/3	79.4 MIB			
2	broadcast exchange (runId cfa5a4fb-0704-4bc6-8668-cfa4a950f3b6) collect at /home/training/Documents/Spark-Programming-In-Python-master/19-ShuffleJoinDemo/SuffleJoinDemo.py:23 +details	2021/03/19 14:48:50	1 s	3/3	64.0 MIB			
1	json at NativeMethodAccessorImpl.java:0 +details	2021/03/19 14:48:49	0.6 s	3/3	64.0 MIB			
0	json at NativeMethodAccessorImpl.java:0 +details	2021/03/19 14:48:47	2 s	3/3	79.4 MIB			

Page: 1 1 Pages. Jump to 1 . Show 100 items in a page. Go

Move to the **SQL** tab

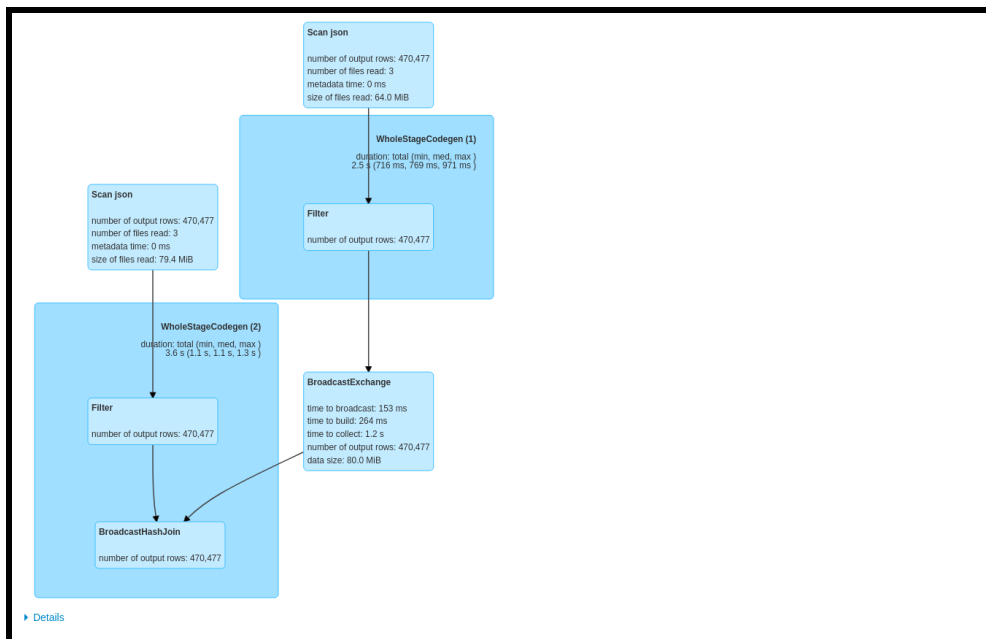


The screenshot shows the Databricks SQL interface. At the top, there are tabs for Jobs, Stages, Storage, Environment, Executors, and SQL. The SQL tab is selected. Below the tabs, it says "Shuffle Join Demo application UI". The main heading is "SQL". Underneath, it says "Completed Queries: 1". A link "Completed Queries (1)" is shown. Below that, there are pagination controls: "Page: 1", "1 Pages. Jump to 1", ". Show 100 items in a page. Go". A table lists the completed queries. The first query is highlighted with a red box. The table has columns: ID, Description, Submitted, Duration, and Job IDs.

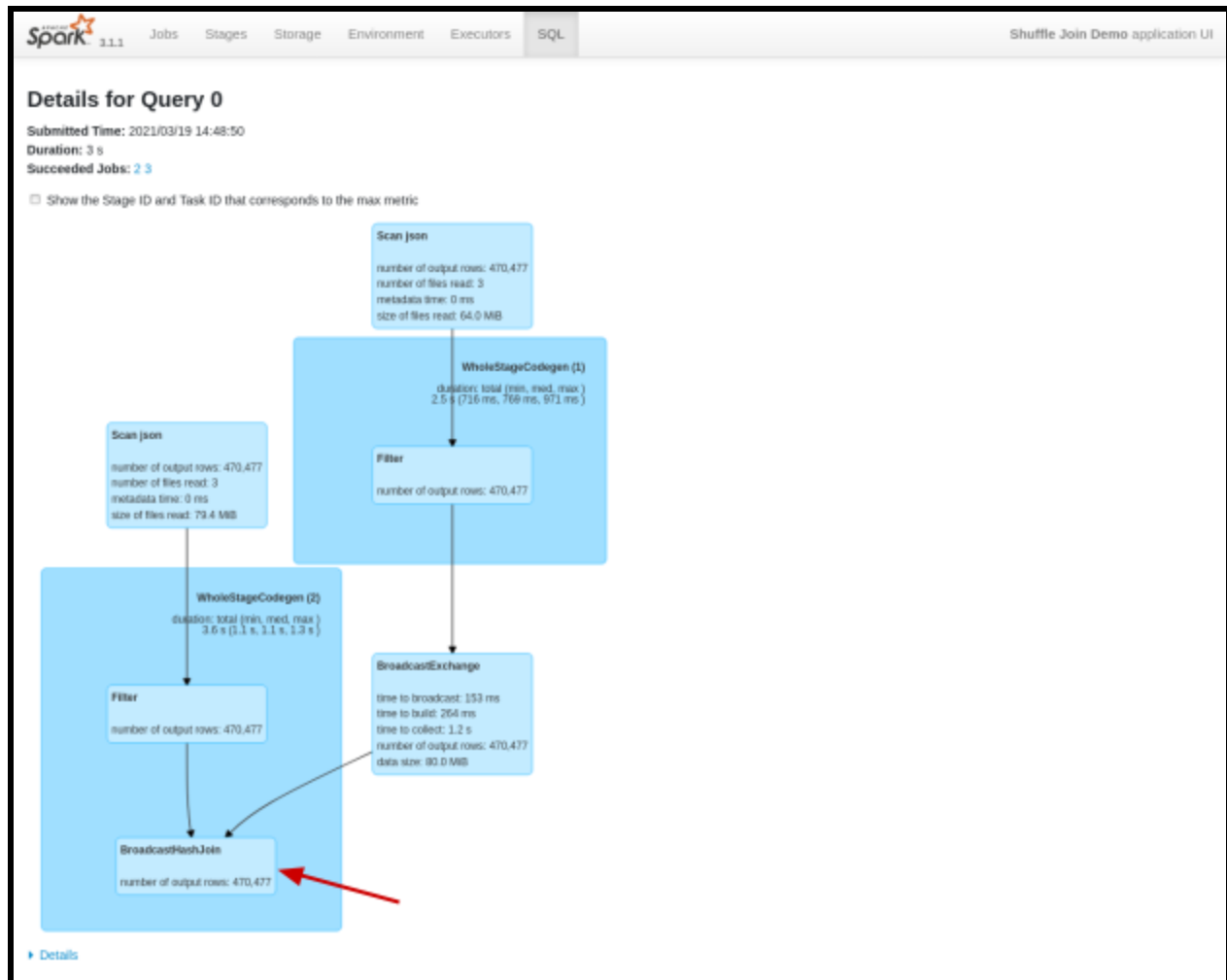
ID	Description	Submitted	Duration	Job IDs
0	collect at /home/training/Documents/Spark-Programming-in-Python-master/19-ShuffleJoinDemo/SqlJoinDemo.py:23	2021/03/19 14:48:50	3 s	[2] [3]

Below the table, there are more pagination controls: "Page: 1", "1 Pages. Jump to 1", ". Show 100 items in a page. Go".

You see the broadcast exchange



And here you can see that, we have a broadcast hash join.



Great! You have learned about the broadcast join and how to implement it.

Voila!! We have successfully completed this exercise.