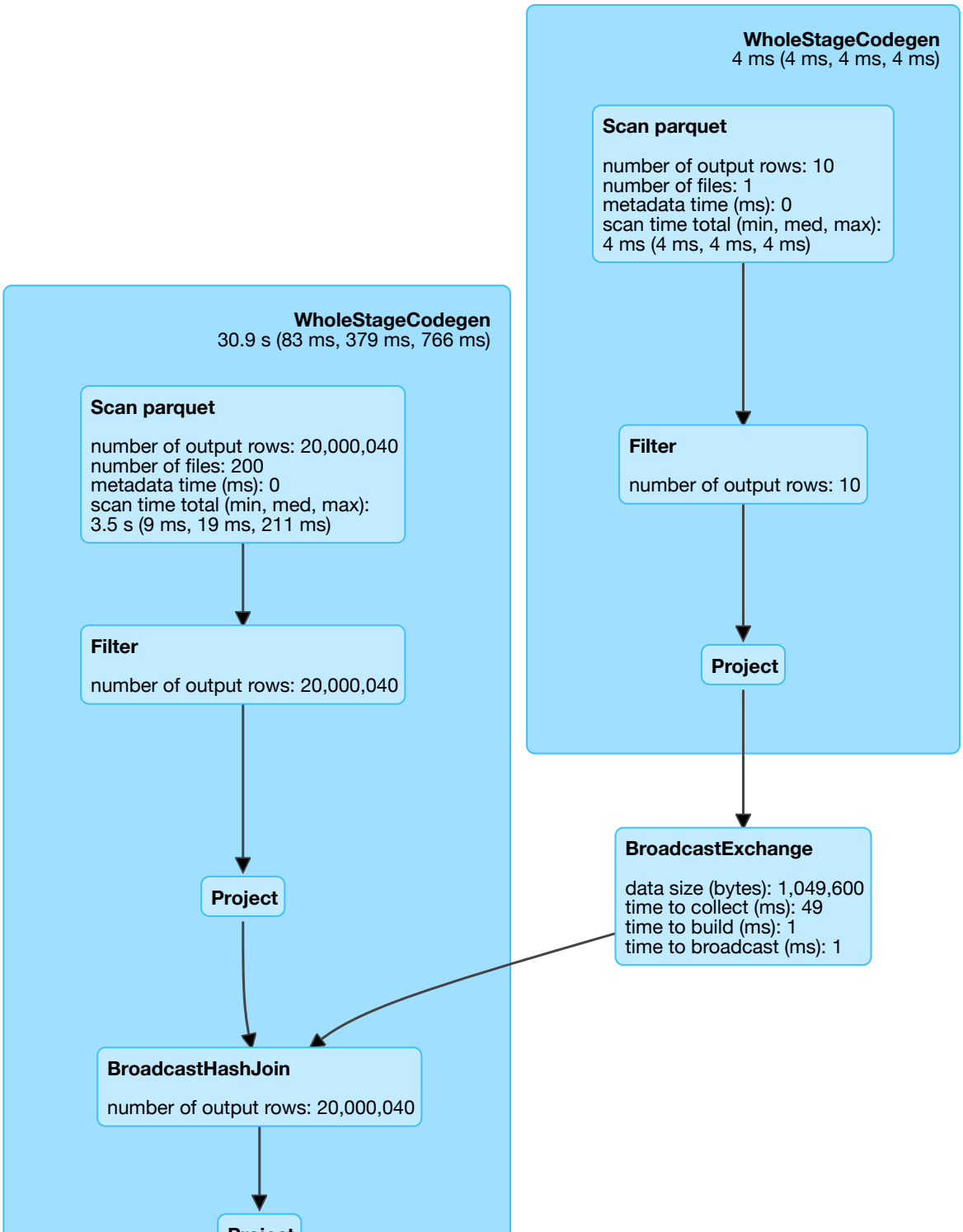
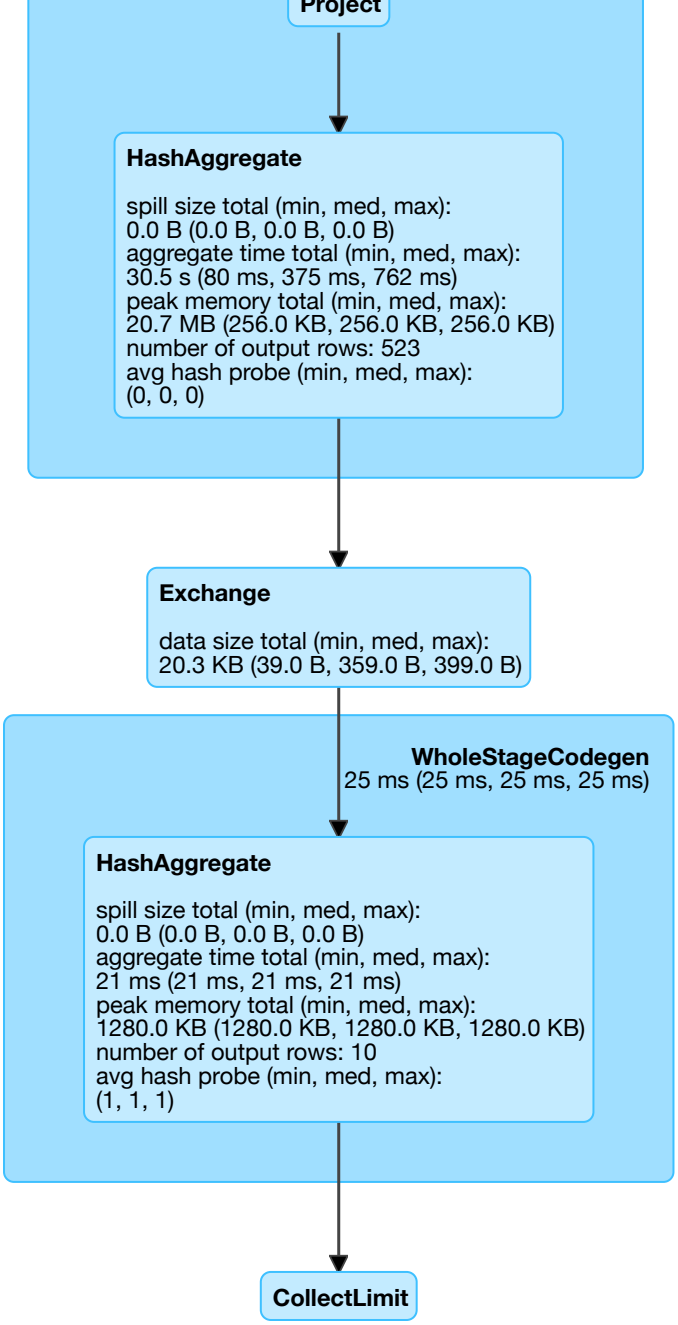
Details for Query 4

Submitted Time: 2020/05/20 22:35:07
Duration: 3 s
Succeeded Jobs: 5 (/jobs/job/?id=5) 6 (/jobs/job/?id=6)




Project
HashAggregate
spill size total (min, med, max):
0.0 B (0.0 B, 0.0 B, 0.0 B)
aggregate time total (min, med, max):
30.5 s (80 ms, 375 ms, 762 ms)
peak memory total (min, med, max):
20.7 MB (256.0 KB, 256.0 KB, 256.0 KB)
number of output rows: 523
avg hash probe (min, med, max):
(0, 0, 0)
Exchange
data size total (min, med, max):
20.3 KB (39.0 B, 359.0 B, 399.0 B)
WholeStageCodegen
25 ms (25 ms, 25 ms, 25 ms)
HashAggregate
spill size total (min, med, max):
0.0 B (0.0 B, 0.0 B, 0.0 B)
aggregate time total (min, med, max):
21 ms (21 ms, 21 ms, 21 ms)
peak memory total (min, med, max):
1280.0 KB (1280.0 KB, 1280.0 KB, 1280.0 KB)
number of output rows: 10
avg hash probe (min, med, max):
(1, 1, 1)
CollectLimit

▼ Details

```
== Parsed Logical Plan ==
GlobalLimit 21
+- LocalLimit 21
   +- Project [cast(seller_id#8 as string) AS seller_id#45, cast(percent_contribution#39 as string) AS
percent_contribution#46]
      +- Aggregate [seller_id#8], [seller_id#8, avg((cast(num_pieces_sold#10 as double) /
cast(daily_target#20 as double))) AS percent_contribution#39]
         +- Join Inner, (seller_id#8 = seller_id#18)
            :- SubqueryAlias `o`
            :  +- SubqueryAlias `orders`
            :     +-
Relation[order_id#6,product_id#7,seller_id#8,date#9,num_pieces_sold#10,bill_raw_text#11] parquet
            +- SubqueryAlias `s`
               +- SubqueryAlias `sellers`
                  +- Relation[seller_id#18,seller_name#19,daily_target#20] parquet

== Analyzed Logical Plan ==
seller_id: string, percent_contribution: string
GlobalLimit 21
+- LocalLimit 21
   +- Project [cast(seller_id#8 as string) AS seller_id#45, cast(percent_contribution#39 as string) AS
percent_contribution#46]
      +- Aggregate [seller_id#8], [seller_id#8, avg((cast(num_pieces_sold#10 as double) /
cast(daily_target#20 as double))) AS percent_contribution#39]
         +- Join Inner, (seller_id#8 = seller_id#18)
            :- SubqueryAlias `o`
            :  +- SubqueryAlias `orders`
            :     +-
Relation[order_id#6,product_id#7,seller_id#8,date#9,num_pieces_sold#10,bill_raw_text#11] parquet
            +- SubqueryAlias `s`
               +- SubqueryAlias `sellers`
                  +- Relation[seller_id#18,seller_name#19,daily_target#20] parquet

== Optimized Logical Plan ==
GlobalLimit 21
+- LocalLimit 21
   +- Aggregate [seller_id#8], [seller_id#8, cast(avg((cast(num_pieces_sold#10 as double) /
cast(daily_target#20 as double))) as string) AS percent_contribution#46]
      +- Project [seller_id#8, num_pieces_sold#10, daily_target#20]
         +- Join Inner, (seller_id#8 = seller_id#18)
            :- Project [seller_id#8, num_pieces_sold#10]
            :  +- Filter isnotnull(seller_id#8)
            :     +-
Relation[order_id#6,product_id#7,seller_id#8,date#9,num_pieces_sold#10,bill_raw_text#11] parquet
            +- Project [seller_id#18, daily_target#20]
               +- Filter isnotnull(seller_id#18)
                  +- Relation[seller_id#18,seller_name#19,daily_target#20] parquet

== Physical Plan ==
CollectLimit 21
+- *(3) HashAggregate(keys=[seller_id#8], functions=[avg((cast(num_pieces_sold#10 as double) /
cast(daily_target#20 as double)))], output=[seller_id#8, percent_contribution#46])
   +- Exchange hashpartitioning(seller_id#8, 1)
      +- *(2) HashAggregate(keys=[seller_id#8], functions=[partial_avg((cast(num_pieces_sold#10 as
double) / cast(daily_target#20 as double)))], output=[seller_id#8, sum#51, count#52L])
         +- *(2) Project [seller_id#8, num_pieces_sold#10, daily_target#20]
            +- *(2) BroadcastHashJoin [seller_id#8], [seller_id#18], Inner, BuildRight
               :- *(2) Project [seller_id#8, num_pieces_sold#10]
```

```
               :  +- *(2) Filter isnotnull(seller_id#8)
               :     +- *(2) FileScan parquet [seller_id#8,num_pieces_sold#10] Batched: true, Format:
Parquet, Location: InMemoryFileIndex[file:/Users/o60774/Downloads/product-sales/sales_parquet],
PartitionFilters: [], PushedFilters: [IsNotNull(seller_id)], ReadSchema:
struct<seller_id:string,num_pieces_sold:string>
               +- BroadcastExchange HashedRelationBroadcastMode(List(input[0, string, true]))
                  +- *(1) Project [seller_id#18, daily_target#20]
                     +- *(1) Filter isnotnull(seller_id#18)
                        +- *(1) FileScan parquet [seller_id#18,daily_target#20] Batched: true, Format:
Parquet, Location: InMemoryFileIndex[file:/Users/o60774/Downloads/product-sales/sellers_parquet],
PartitionFilters: [], PushedFilters: [IsNotNull(seller_id)], ReadSchema:
struct<seller_id:string,daily_target:string>
```