Optimizing Apache Spark SQL Joins

Vida Ha Solutions Architect



About Me

2005 Mobile Web & Voice Sear 100ge



About Me

2005 Mobile Web & Voice Sear 100ge

2012 Reporting & Analytics





About Me

Mobile Web & Voice Sear (100)

Reporting & Analytics







Evolution of Spark...

2014:

- Spark 1.x
- RDD based API's.
- Everyday I'm Shufflin'

2017:

- Spark 2.x
- Dataframes & Datasets
- Adv SQL Catalyst
- Optimizing Joins



Spark SQL Joins

```
SELECT ...

FROM TABLE A

JOIN TABLE B

ON A.KEY1 = B.KEY2
```

Topics Covered Today

Basic Joins:

- Shuffle Hash Join
 - Troubleshooting
- Broadcast Hash Join
- Cartesian Join

Special Cases:

- Theta Join
- One to Many Join

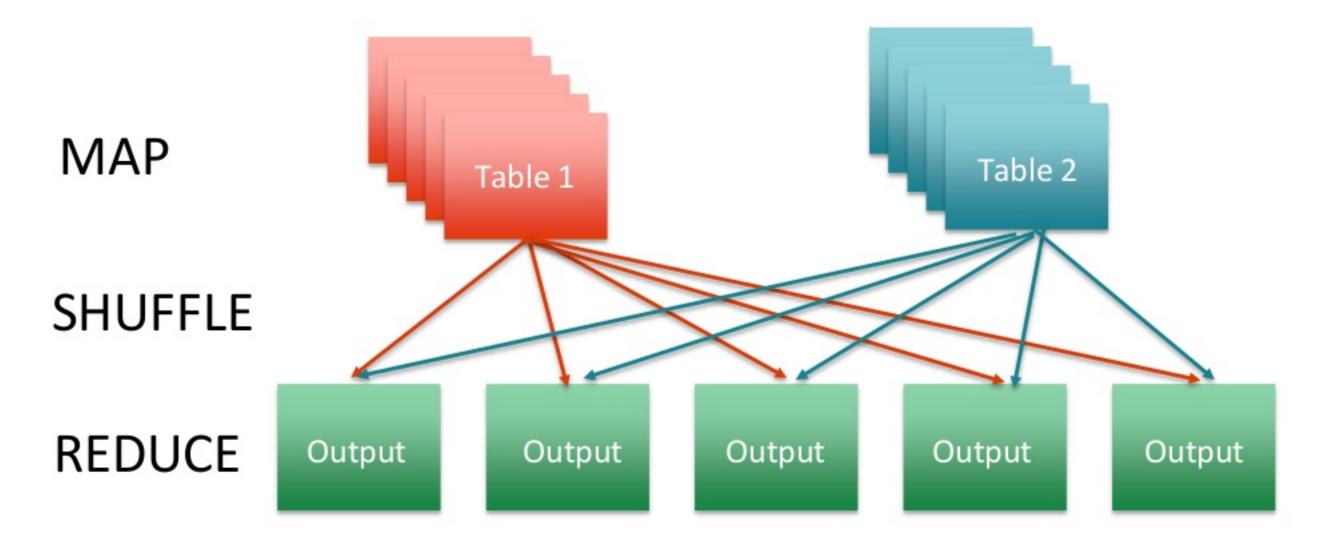


Shuffle Hash Join

A Shuffle Hash Join is the most basic type of join, and goes back to Map Reduce Fundamentals.

- Map through two different data frames/tables.
- Use the fields in the join condition as the output key.
- Shuffle both datasets by the output key.
- In the reduce phase, join the two datasets now any rows of both tables with the same keys are on *databricks* the same machine and are sorted.

Shuffle Hash Join





Shuffle Hash Join Performance

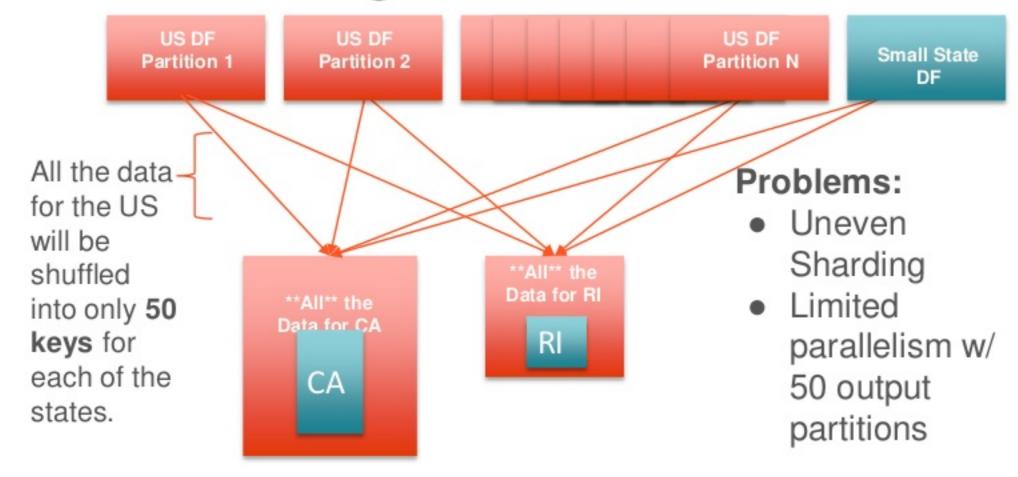
Works best when the DF's:

- Distribute evenly with the key you are joining on.
- Have an adequate number of keys for parallelism.

```
join_rdd = sqlContext.sql("select *
  FROM people_in_the_us
  JOIN states
  ON people_in_the_us.state = states.name")
```

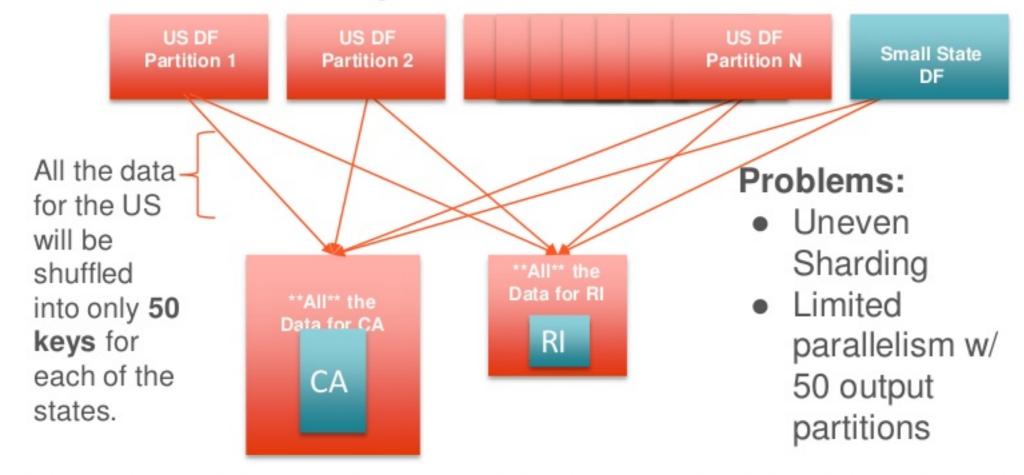


Uneven Sharding & Limited Parallelism,



A larger Spark Cluster will not solve these problems!

Uneven Sharding & Limited Parallelism,



Broadcast Hash Join can address this problem if one DF is small enough to fit in memory.



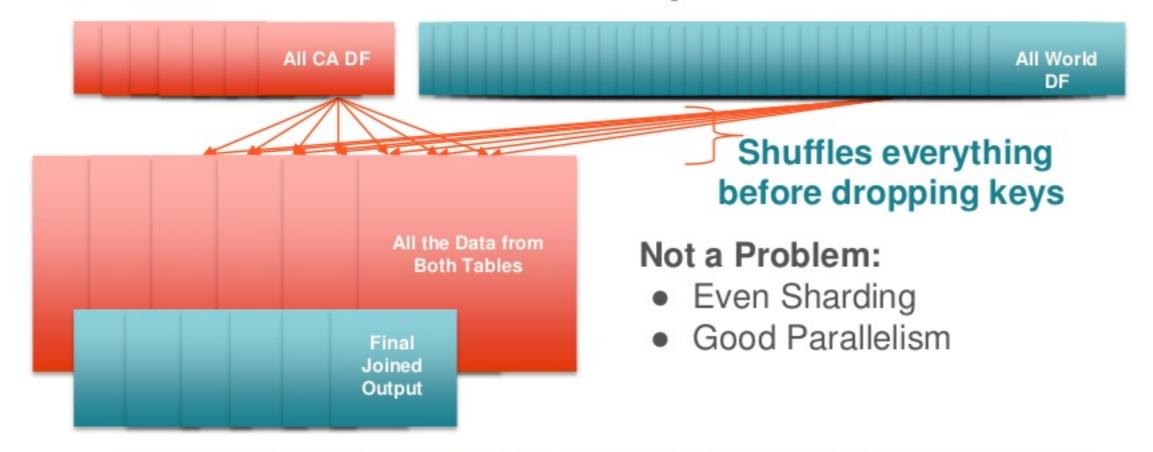
More Performance Considerations

```
join_rdd = sqlContext.sql("select *
   FROM people_in_california
   LEFT JOIN all_the_people_in_the_world
   ON people_in_california.id =
      all_the_people_in_the_world.id")
```

Final output keys = # of people in CA, so don't need a huge Spark cluster, right?



Left Join - Shuffle Step

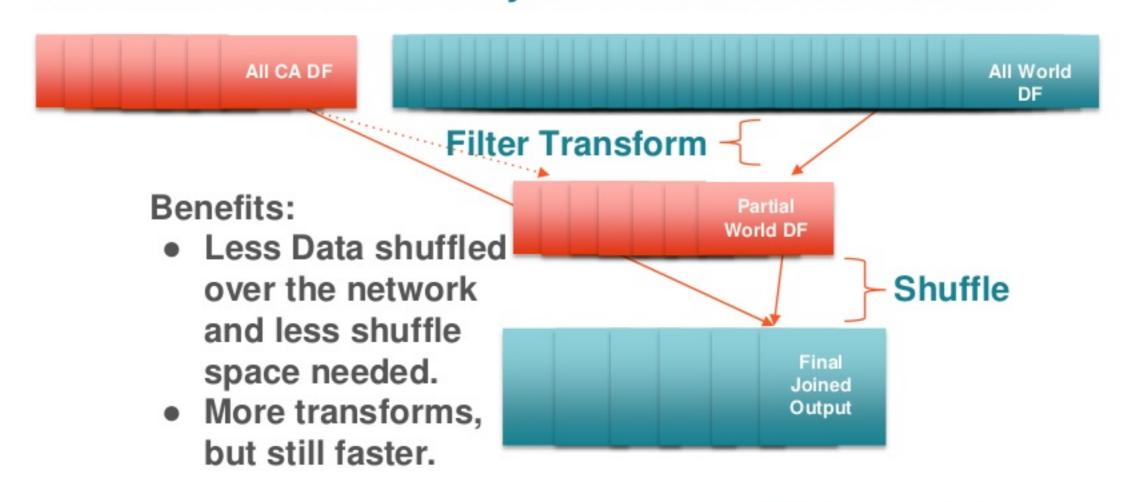


The Size of the Spark Cluster to run this job is limited by the Large table rather than the Medium Sized Table.



A Better Solution

Filter the World DF for only entries that match the CA ID





What's the Tipping Point for Huge?

- Can't tell you.
- There aren't always strict rules for optimizing.
- If you were only considering two small columns from the World RDD in Parquet format, the filtering step may not be worth it.

You should understand your data and it's unique properties in order to best optimize your Spark Job.



In Practice: Detecting Shuffle Problems

Danks																
Name of Street		-	Italia	Lacarity Lacar	Summittee this Print	Laureth, Tirle	Derbe	Strington String	Tank Decembelson	Si Tora	Result Seriphotos Yang	destina Seculi	rest		Shaffe Mile	
	25		N-00000	PRODOSE LOCAL	3 - qu-10 -0 -101 -100-us- sont-2 -computer commit	2014/00/00 20:00:00	**	12 ins	S rise	274	l-ma	Sires.	SCO ME	10	3.1 MB	
	ŞN		BUDDESE.	PROCESS LOGAL	A Fay-10-0-187 Hillion- well-Boardura Fransi	2016/00/00 2016/00/00	**	12:00	7.00	013	0.04	S rea	SALE ME.	14	47 MB	
	38		8000088	PRODURE LOCAL	A Facility of the Hillians	2015/50/02 2015/30/00	**	Him	0 may	27.6	0 mm	S'rea	34.3 MB		43 MB	
		0	NUOCOS	PRESCRIBE LOCAL	BY SHIPS SHIPS FROM WARE Extrapolar Promise	SELECTION OF THE PERSON NAMED IN	**	11 (46)	tria	tia	D-ma	2100	SAIL NO. Produced	11	0.4360	
•	20		Notes	PH00096.100%	3/9-100101105us sectlumous/secu	20 CD-60	**	fee	7100	279	D TOTAL	S'ree	SHE MA	11 200	41.00	
	.26		5000000	PRODUCES LOCAL	9 / 20 10 0 / RF / RELIGIONS week bownsome critical	2015/10/03 2016/40	**	Res	0.000	21.6	Trees.	Sea	SST MA Printers	10	43 MB	
	n		BL000BB	PREDCER LOCAL	\$7 (a. 10 to 181 185 cal- sees 3 removal internal	201 \$152,000 501-00-41	.74	fee	3.000	274	Des.	504	SLEME Framed	10	4.6 MB	
*	=		BUCCOSS	PRECEDELLOCAL	87 to 10-0-187 FBLor- well 8 compute interest	DETERMINED DECEMENT	**	700	1116	514	Des	Sime	SAA NB. Swings		43 MB	
	×	9	RUNNING	99000B.100K	\$1 (p-10-0-197-190-us- seed \$ compute internal	2015/20/60 20109-97	**	time	See		(-ma	den.				
,	#	0	HIPPRIS	PRECIOE, LOCAL	07 ip-10-0-197-185 on weer 0 upropoles internal	2016/20/03 20:00:40	**	Des	S res		2-we	See				
*	**		PLANAGE	H0008,1004L	973-193-197-195-se- ent 6 company money	SHAMESTA STATES	5+	0 ===	li rev		5-10	See.				
71	þe	0	RUMMO	MADOWN TOOM	0 / 5x 10-0:107-185-ss- week 2 compute internal	201 M/02/03 20/00/40	**	0 mu	0 mm		2 ma	2100				

Check the Spark UI pages for task level detail about your Spark job.

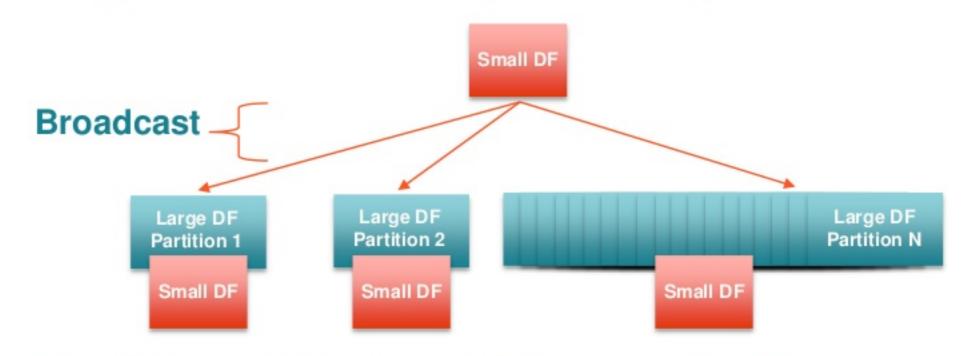
Things to Look for:

- Tasks that take much longer to run than others.
- Speculative tasks that are launching.
- Shards that have a lot more input or shuffle output.



Broadcast Hash Join

Optimization: When one of the DF's is small enough to fit in memory on a single machine.



Parallelism of the large DF is maintained (n output partitions), and shuffle is not even needed.



Broadcast Hash Join

- Often optimal over Shuffle Hash Join.
- Use "explain" to determine if the Spark SQL catalyst hash chosen Broadcast Hash Join.
- Should be automatic for many Spark SQL tables, may need to provide hints for other types.



Cartesian Join

 A cartesian join can easily explode the number of output rows.

 $100,000 \times 100,000 = 10 \text{ Billion}$

- Alternative to a full blown cartesian join:
 - Create an RDD of UID by UID.
 - Force a Broadcast of the rows of the table.
 - Call a UDF given the UID by UID to look up the table rows and perform your calculation.
- Time your calculation on a sample set to size your cluster.



One To Many Join

- A single row on one table can map to many rows on the 2nd table.
- Can explode the number of output rows.
- Not a problem if you use parquet the size of the output files is not that much since the duplicate data encodes well.



Theta Join

```
join_rdd = sqlContext.sql("select *
  FROM tableA
  JOIN tableB
  ON (keyA < keyB + 10)")</pre>
```

- Spark SQL consider each keyA against each keyB in the example above and loop to see if the theta condition is met.
- Better Solution create buckets for keyA and KeyB can be matched on.



Thank you



Questions?

