

Map-Reduce-Merge

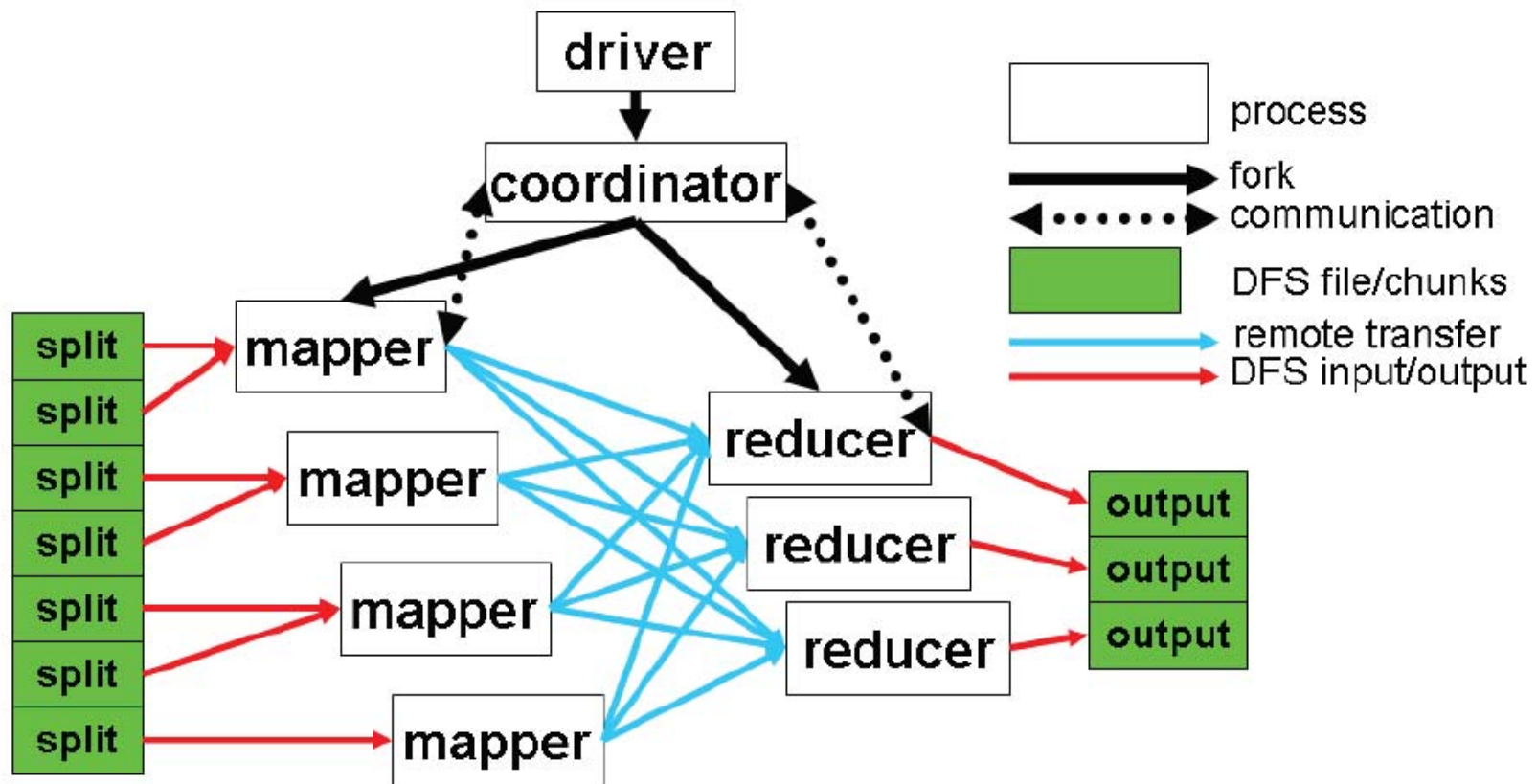
Zhang Kun-Long

zhangkl@tju.edu.cn

2007-12-05

MapReduce

1. Map: $(k1, v1) \rightarrow [(k2, v2)]$
2. Reduce: $(k2, [v2]) \rightarrow [v3]$



MapReduce and SELECT

SELECT 目标列表达式

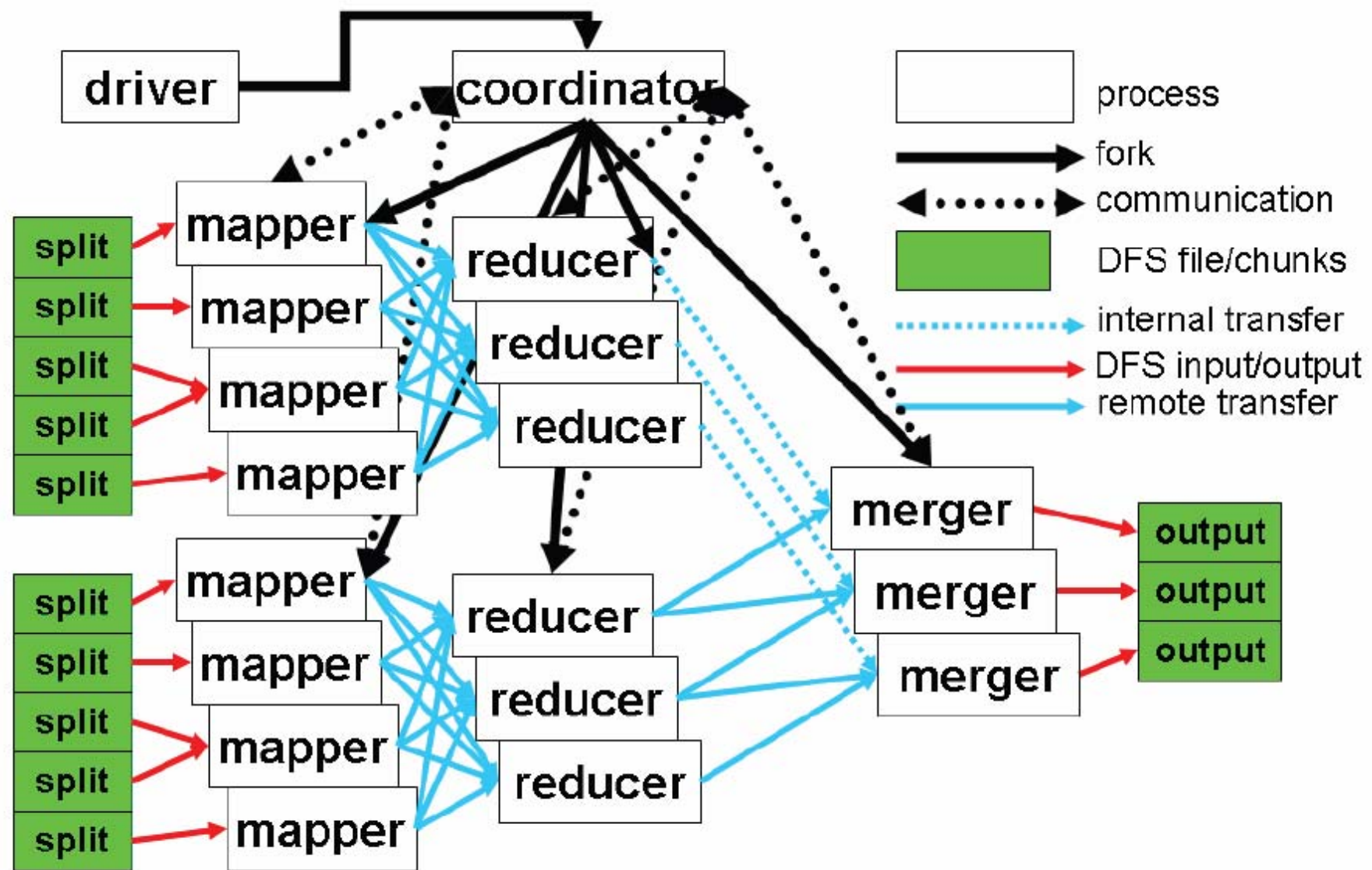
FROM 表名

WHERE 条件表达式

GROUP BY 列名

HAVING 条件表达式

ORDER BY 列名



Map-Reduce-Merge

1. Map: $(k1, v1) \rightarrow [(k2, v2)]$
2. Reduce: $(k2, [v2]) \rightarrow (k2, [v3])$
3. Merge: $((k2, [v3]), (k3, [v4])) \rightarrow (k4, [v5])$

Map-Reduce-Merge: Extending MapReduce

1. Change to reduce phase
2. Merge phase
3. Additional user-definable operations
 - a. partition selector - which data should go to which merger?
 - b. Processor - process data on an individual source
 - c. Merger - define logic to do the merge
 - d. configurable iterators - how to step through each of the lists as you merge

emp-id	emp-info: dept-id	emp-info: bonus
1	B	innovation award (\$100)
1	B	hard worker award (\$50)
2	A	NULL (\$0)
3	A	high-performer (\$150)
3	A	Innovation award (\$100)

LHS mapper computes emp bonuses

<u>emp-id</u>	<u>dept-id</u>	bonus
1	B	\$100
1	B	\$50
2	A	\$0
3	A	\$150
3	A	\$100

LHS reducer sorts on (dept-id, emp-id) pair and sums up emp bonuses

<u>emp-id</u>	<u>dept-id</u>	bonus-sum
2	A	\$0
3	A	\$250
1	B	\$150

match keys on dept-id

<u>emp-id</u>	bonus
2	\$0
3	\$237.5
1	\$172.5

dept-id	dept-info: bonus adjustment
B	1.1
A	0.9

RHS mapper retrieves bonus adjustment

<u>dept-id</u>	bonus adjustment
B	1.15
A	0.95

RHS reducer modified bonus adjustment and sorts on dept-id

<u>dept-id</u>	bonus adjustment
A	0.95
B	1.15

A sort-merge merger joins LHS and RHS reduced outputs, then computes final emp bonuses.

Implementing Relational Algebra Operations

1. Projection
2. Aggregation
3. Selection
4. Cartesian Product
5. Joins
6. Set Union
7. Set Intersection
8. Set Difference
9. Rename

References

- H. Yang et. al., “Map-reduce-merge: simplified relational data processing on large clusters”, SIGMOD 2007.