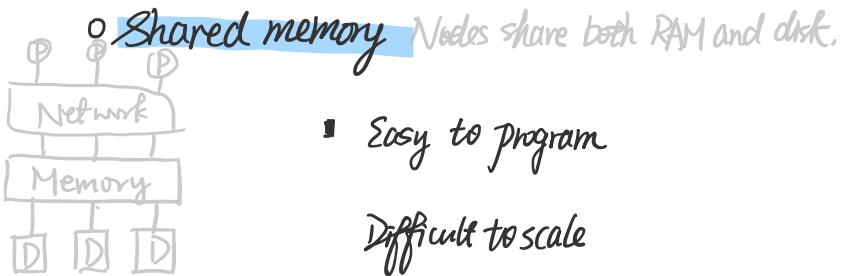


Parallel Database

Friday, May 17, 2019 1:40 PM

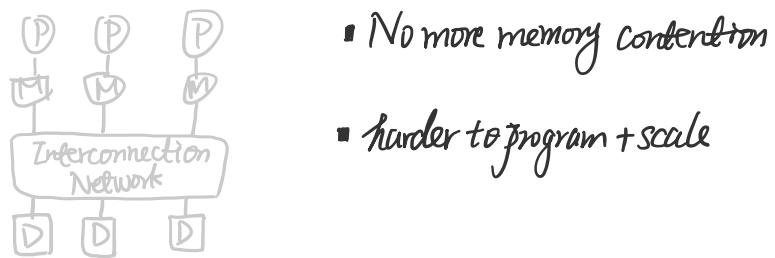
- Architecture



- Easy to program

- Difficult to scale

- **Shared disk**



- No more memory contention

- harder to program + scale

- **Shared nothing** (Spark)

- Easy to maintain and scale

- Most difficult to administer and tune *program*

- Approaches to Parallel Query Evaluation

- o **inter-query parallelism**

- Function per node \rightarrow one node runs through query for a part of data.
- ✓ **transactional workloads**

- o **inter-operator parallelism**

- Operator per node
- ✓ **analytical workloads**

- o **Intra-operator parallelism** \rightarrow most scalable.

- Operator on multiple nodes

- **Distributed data processing**

- o Data is horizontally partitioned on servers

- o Operators may require data shuffling

- **Horizontal Data Partitioning**

- o **Block Partition** guarantee that data is uniform distribution.

- arbitrarily $\text{size}(R_1) \approx \text{size}(R_2) \approx \dots \text{size}(R_n)$

- o **Hash partitioned on attribute A**

- tuple t goes to chunk i , where $i = h(t.A) \bmod P + 1$.

- Calling hash function is free in this case

- o **Range partitioned on attribute A**

- **range partitioned on attribute A**
 - Partition in range of A into $-\infty = v_0 < v_1 < \dots < v_p = \infty$
 - Tuple t goes to chunk i if $v_{i+1} \leq t.A < v_i$.
- **Skewed data** Informally, we say that the data is skewed if one server holds much more data than average.
 - Block, hash on key partition won't produce skewed data.
 - Block partition on attribute A may cause skewed data.

Good hash function

E.g., Group by A

- R is hash partitioned on A \Rightarrow local group by
- R is hash partitioned on key or block partition \Rightarrow Repartition.

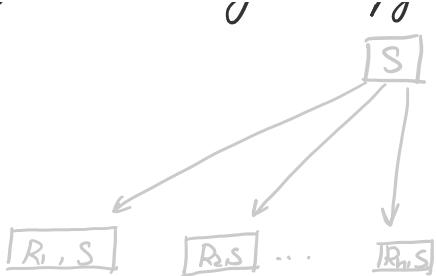
Reshuffle the data on attribute A. then locally group by

- Assume the data is not skewed

nodes doubles run time halves

nodes, # data doubles. run time doesn't change

E.g. 2., Broadcast join : copy S to every single server.



good when S is small.