

1. Recall that **histograms** can be used for constructing load-balanced range partitions. Suppose you have a histogram where values are between 1 and 100, and are partitioned into 10 ranges, 1–10, 11–20, ..., 91–100, with frequencies 15, 5, 20, 10, 10, 5, 5, 20, 5, and 5, respectively. Construct a **balanced range partitioning vector** to divide the values into 5 partitions.

A partitioning vector which gives 5 partitions with 20 tuples in each partition is: [21, 31, 51, 76].

2. In a **range selection** on a **range-partitioned attribute**, it is possible that only *one* node may need to be accessed. Describe the benefit and drawback of this possible case.

- If there are only a few tuples in the queried range, then the query can be processed quickly on a single node. This allows parallel execution of other queries on other nodes, resulting in higher throughput while maintaining good response time.
- On the other hand, if there are many tuples in the queried range, many tuples have to be retrieved from a single node, resulting in an I/O bottleneck (hot-spot) at the node. The query takes a long time to execute as there is no parallelism within its execution. This is an example of **execution skew**: all processing occurs in one or only a few partitions.

Assignment Project Exam Help

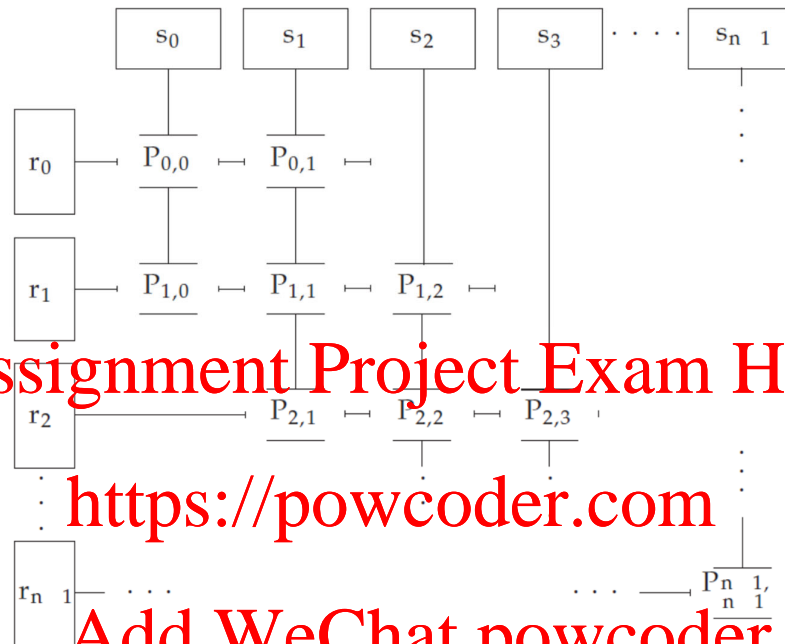
3. Consider the join of two relations r and s using **partitioned join**, where r and s occupy 1000 and 500 blocks, respectively. Let there be a 10-node shared-nothing parallel database and r and s are distributed uniformly among all the disks of the 10 nodes using **round robin**. What is the total number of blocks that are required to ship among the nodes in performing the partitioning step?

We need to send each tuple of r and s to one of the 10 nodes, using range partitioning or hash partitioning. Since there are 150 blocks per node, each node will have about 15 blocks of data for each other node. So, each node ships 135 blocks to the other 9 nodes. The total number of blocks that are required to ship among the nodes is 1350 blocks.

4. Consider join processing using *symmetric fragment and replicate* with *range partitioning*. How can you optimize the evaluation if the join condition is of the form $|r.A - s.B| \leq k$, where k is a small constant?

Here, $|x|$ denotes the absolute value of x . A join with such a join condition is called a *band join*.

- Relation r is partitioned into n partitions, r_0, r_1, \dots, r_{n-1} , and s is also partitioned into n partitions, s_0, s_1, \dots, s_{n-1} using the *same* range-partitioning vector.
- The partitions are replicated and assigned to node as shown below.



- Each fragment is replicated on 3 nodes only, unlike in the general case where it is replicated on n nodes.
- The number of nodes required is now approximately $3n$, instead of n^2 in the general case.
- Therefore given the same number of nodes, we can partition the relations into more fragments with this optimization, thus making each local join faster.