Joiner Hybrid Hash-Join Algorithm

Author: Heiko Hofer
Date: March 08, 2010

Input: Two tables where the first will be referred to as the left table and the second as the right table.
Output: A single table, the joined table.

The Joiner combines rows from the left table and the right table. Two rows are combined and added to the joined table when they satisfy the joining criterion. In the simple case two cells must match.

The input table with less number of rows is referred as the inner table and the other is referred as the outer table. The Joiner uses the Hybrid Hash-Join algorithm.

Step 1: Read rows of the inner table read in n partitions where information for joining and the indices the rows are stored. The partitioning is done in a way that a row in the i-th partition of the inner table does not match with any row of the k-th partition of the outer table for $i \neq k$. This allows to join partitions independently.
Partitions are skipped for reading when free space on the heap becomes low. When even one partition does not fit in main memory, the number of partitions is raised which leads to smaller partitions.

Step 2: Join the read partitions with the outer table. The indices of the joined rows are stored on disk. This step is optimized to have a small memory footprint. Since a low memory condition in this step is fatal.

Step 3: Proceed with Step 1 reading the skipped partitions.

Step 4: The joined rows are read, sorted and merged to the joined table.

Remarks on the Ordering: The rows in the joined table are ordered with the following order criteria:
1. The type of join: Inner Join, Left Outer Join, Right Outer Join.
2. The order of the left table.
3. The order of the right table.