Australian National University
School of Computer Science

Computer Science COMP3420 (2010) – Tutorial Three

Question 1. Review the syntax in the definitions of *data cube* and *dimension* tables, can you give a concrete example?

Question 2. What is the lattice of data cube? what is the cuboid? what is the cuboid cells? and what is the base cell or aggregate cell?

Question 3. What is the *iceberg* data cube? what is the *Apriori* property? what is the *iceberg condition*?, and what is the *anti-monotonic* property?

Question 4. Given a data cube consisting four dimensions D_1,D_2, D_3 and D_4, assume that the numbers of levels associated these dimensions are 10, 4, 3, and 15. How many cuboids do the data cube contain?

Question 5. Can you point out the differences between the Multi-way array aggregation algorithm and the BUC algorithm in terms of the data cube computation? in which case, which approach is preferable?

Question 6. When we apply the high OLAP minimal cubing approach for OLAP queries, which important aspects should be considered to trade-off the precomputed sub-data cubes and the on-the-fly datacube computation. Can you list them?

Question 7. Suppose that a base cuboid has three dimensions, A, B, C with the following number of cells: $|A| = 1,000,000$, $|B| = 100$, $|C| = 1,000$. Suppose that each dimension is evenly partitioned into 10 portions for chunking.

- Assuming each dimension has only one level, draw the complete lattice of the cube.
- If each cube cell stores one measure with 4 bytes, what is the total sizes of the computed cube if the cube is dense?
- State the order for computing the chunks in the cube that requires the least amount of space, and compute the total amount of main memory space required for computing the 2-D planes.

Question 8. Suppose that we would like to compute an iceberg cube for the dimensions, A, B, C, and D, where we wish to materialize all cells that satisfy a minimum support count of at least θ, and assume that $|A| < |B| < |C| < |D|$. Show the BUC processing tree (which shows the order in which the BUC algorithm explores the lattice of a data cube, starting from *all*) for the construction of the above iceberg cube.