Let's use the query SELECT SUM(total\_sales) FROM sales\_data WHERE region = 'EAST'; to explain Snowflake's three caches in order.

**First Execution (The "Cold" Run)**

When you run this query for the very first time on a new virtual warehouse, Snowflake checks its caches in a specific order:

**Result Cache:** Snowflake's Service Layer first checks the Result Cache for an identical query result. Since this is the first time, it finds nothing.

**Local Disk Cache:** The query is then sent to the Compute Layer (your virtual warehouse). This warehouse is "cold" and has no data from the sales\_data table in its Local Disk Cache.

**Remote Disk:** The virtual warehouse has no choice but to go to the Remote Disk Storage to get the raw data files for the sales\_data table. This is the slowest part of the process.

While executing the query, Snowflake will read the necessary data from the Remote Disk, put a copy of it in the Local Disk Cache, and then perform the SUM() calculation. Finally, it will store the result in the Result Cache for future use.

**Second Execution (The "Warm" Run)**

Now, suppose you run a slightly different query on the same warehouse right after the first one: SELECT COUNT(\*) FROM sales\_data WHERE region = 'EAST';

**Result Cache:** Snowflake checks the Result Cache, but the query is not an exact match (COUNT(\*) vs SUM(total\_sales)), so it doesn't find a cached result.

**Local Disk Cache:** The query then goes to the Compute Layer. It needs data from the same sales\_data table. Since that data was just put into the Local Disk Cache during the first run, the virtual warehouse can now read the data from its high-speed SSDs. This is significantly faster than going back to the Remote Disk.

The Local Disk Cache allows for quick reuse of the underlying data, even if the query logic changes.

**Third Execution (The "Hot" Run)**

Finally, suppose you run the original query again (exactly the same as the first one): SELECT SUM(total\_sales) FROM sales\_data WHERE region = 'EAST';

**Result Cache:** This time, Snowflake checks the Result Cache and finds a perfect match. The query is identical, and the underlying data hasn't changed. The result is returned instantly from the cache in milliseconds, without using any compute resources.

This is the fastest and most efficient way to get your results in Snowflake.