Query States

A typical continuous query (CQ) has a longer life time. Once installed it can pass through different states.

If a CQ is added to the system it is in *inactive* state. In this state the query is installed but processes no data and causes no system load. With a *start*-transition the query can be moved to the *active* state. Here, the query receives and processes data from sources and sends results to sinks. This is the state where a query typically causes the most system load. Calling *stop* deactivates the query again and moves to inactive. Nevertheless, queries can be removed at any time (independent from their current query state).

A query in active state can be *suspended*. This means, no data is processed anymore but the incoming data is buffered temporally. A suspended query can be resumed without any loss of data. This transition can reduce processing costs, but typically increases memory usage for the input buffering. Although, compression or swapping to external devices can reduce the necessary memory costs, this state should not last to long and should only be used to temporary high load or for input with a lower data rate. If a suspended query is resumed, the buffered data elements are processed first.

Another way to reduce load is called Load Shedding. Here, some of the incoming data elements are discarded after their arrival. In this partial active state the CQ process only a fraction of the data stream. This also reduces the system load, but, depending on the query, the results will be less accurate. Finally, combinations of query states are imaginable (e.g., partially suspended queries).