

CLAIMS

1. A computer-implemented method for executing a dataflow graph that represents a query on data items in a stream of near real-time data to provide, as the dataflow graph is being executed, intermediate results for the query, the method including:

receiving a stream of near real-time data having data items located in different places in the stream;

between a first time and a second time, intermittently executing the dataflow graph that represents the query multiple times, with the dataflow graph being executed by one or more computer systems in near real-time with respect to receipt of the stream of near real-time data and being executed upon the stream of near real-time data for two or more of the data items, with the dataflow graph including computer code to implement the query, and with the dataflow graph receiving as input query specifications for the query;

generating, during execution of the dataflow graph, one or more query results that satisfy the query;

generating intermediate results from the one or more query results, as the dataflow graph intermittently executes between the first time and the second time, by aggregating the one or more query results with one or more prior query results of one or more prior executions of the dataflow graph on the stream of near real-time data for the two or more of the data items that previously appeared in the stream of near real-time data; and

transmitting to one or more client devices the intermediate results during intermittent execution of the dataflow graph, prior to completion of execution of the dataflow graph.

2. The computer-implemented method of claim 1, further including:
at a subsequent point in time, aggregating the intermediate results with results for executing the dataflow graph at the subsequent point in time to generate final results.

3. The computer-implemented method of claim 2, further including:
storing the final results and discarding the intermediate results.

4. The computer-implemented method of claim 1, wherein intermittently executing the dataflow graph includes:

executing the dataflow graph on a first one of the data items in the stream of near real-time data located in a first portion of the stream of near real-time data, and

executing the dataflow graph on a second one of the data items in the stream of near real-time data located in a second portion of the stream.

5. The computer-implemented method of claim 1, wherein the dataflow graph includes components that represent operations to be performed in execution of the query, and wherein the method further includes:

for a component:

performing a checkpoint operation that saves a local state of the component to enable recoverability of a state of the dataflow graph.

6. The computer-implemented method of claim 1, wherein the dataflow graph is executed on data items that appear in the stream of near real-time data during a period of time the end of which is unknown at a start of executing the dataflow graph.

7. The computer-implemented method of claim 1, wherein an amount of data items in the stream of near real-time data on which the dataflow graph is executed is unknown at a start of executing the dataflow graph.

8. The computer-implemented method of claim 6, further including:
accessing information indicative of user-defined custom operations for data transformation on the aggregated results;

executing the user-defined custom operations on the aggregated results; and
transforming the aggregated results in accordance with the user-defined custom operations.

9. The computer-implemented method of claim 1, further including:

generating, based on the aggregated results, a near real-time alert to alert a user of detection of a pre-defined condition.

10. The computer-implemented method of claim 1, wherein the stream of near real-time data includes a stream of near real-time data in which data items are (i) periodically received at different times or (ii) continuously received at different times.

11. The computer-implemented method of claim 1, further including receiving the stream of near real-time data from a data queue, a data repository, or a data feed.

12. The computer-implemented method of claim 1, wherein the dataflow graph is a first dataflow graph, with the method further including:

selecting a second dataflow graph for execution on two or more of the data items that appear at different locations in the stream of near real-time data; and

executing the first and second dataflow graphs in near real-time with respect to the data items of the stream of near real-time data.

13. The computer-implemented method of claim 1, further including:
generating information for a user interface that when rendered on a display device includes:

input fields for input of information defining dataflow graphs to be executed on the stream of data.

14. The computer-implemented method of claim 1, further including:
accessing, in a control repository, pre-defined dataflow graphs that are candidates for execution on the stream of near real-time data.

15. A system for executing a dataflow graph that represents a query on data items in a stream of near real-time data to provide, as the dataflow graph is being executed, intermediate results for the query, the system including:

one or more processing devices; and

one or more machine-readable hardware storage devices storing instructions that are executable by the one or more processing devices to perform operations including:

- receiving a stream of near real-time data having data items located in different places in the stream;

- between a first time and a second time, intermittently executing the dataflow graph that represents the query multiple times, with the dataflow graph being executed by one or more computer systems in near real-time with respect to receipt of the stream of near real-time data and being executed upon the stream of near real-time data for two or more of the data items, with the dataflow graph including computer code to implement the query, and with the dataflow graph receiving as input query specifications for the query;

- generating, during execution of the dataflow graph, one or more query results that satisfy the query;

- generating intermediate results from the one or more query results, as the dataflow graph intermittently executes between the first time and the second time, by aggregating the one or more query results with one or more prior query results of one or more prior executions of the dataflow graph on the stream of near real-time data for the two or more of the data items that previously appeared in the stream of near real-time data; and

- transmitting to one or more client devices the intermediate results during intermittent execution of the dataflow graph, prior to completion of execution of the dataflow graph.

16. The system of claim 15, wherein the operations further include:
at a subsequent point in time, aggregating the intermediate results with results for executing the dataflow graph at the subsequent point in time to generate final results.

17. The system of claim 16, wherein the operations further include:
storing the final results and discarding the intermediate results.

18. The system of claim 15, wherein intermittently executing the dataflow graph includes:

executing the dataflow graph on a first one of the data items in the stream of near real-time data located in a first portion of the stream of near real-time data, and

executing the dataflow graph on a second one of the data items in the stream of near real-time data located in a second portion of the stream.

19. The system of claim 15, wherein the dataflow graph includes components that represent operations to be performed in execution of the query, and wherein the operations further include:

for a component:

performing a checkpoint operation that saves a local state of the component to enable recoverability of a state of the dataflow graph.

20. The system of claim 15, wherein the dataflow graph is executed on data items that appear in the stream of near real-time data during a period of time the end of which is unknown at a start of executing the dataflow graph.

21. The system of claim 15, wherein an amount of data items in the stream of near real-time data on which the dataflow graph is executed is unknown at a start of executing the dataflow graph.

22. One or more machine-readable hardware storages for executing a dataflow graph that represents a query on data items in a stream of near real-time data to provide, as the dataflow graph is being executed, intermediate results for the query, the one or more machine-readable hardware storages storing instructions that are executable by one or more processing devices to perform operations including:

receiving a stream of near real-time data having data items located in different places in the stream;

between a first time and a second time, intermittently executing the dataflow graph that represents the query multiple times, with the dataflow graph being executed by one or more

computer systems in near real-time with respect to receipt of the stream of near real-time data and being executed upon the stream of near real-time data for two or more of the data items, with the dataflow graph including computer code to implement the query, and with the dataflow graph receiving as input query specifications for the query;

generating, during execution of the dataflow graph, one or more query results that satisfy the query;

generating intermediate results from the one or more query results, as the dataflow graph intermittently executes between the first time and the second time, by aggregating the one or more query results with one or more prior query results of one or more prior executions of the dataflow graph on the stream of near real-time data for the two or more of the data items that previously appeared in the stream of near real-time data; and

transmitting to one or more client devices the intermediate results during intermittent execution of the dataflow graph, prior to completion of execution of the dataflow graph.