WHAT IS CLAIMED IS:

1. A computer-implemented method for determining a characteristic of a data pipeline, comprising:

receiving, with at least one processor, metadata parameters for a data pipeline, wherein the metadata parameters for the data pipeline comprise at least one of the following: data associated with one or more input datasets of a data pipeline, data associated with transformation logic of a data pipeline, data associated with a mapping of an input to an output of a data pipeline, data associated with one or more output datasets of a data pipeline, or any combination thereof;

storing, with at least one processor, the metadata parameters in a data repository;

generating, with at least one processor, a logical representation of the data pipeline based on the metadata parameters;

using, with at least one processor, the metadata parameters to instantiate the data pipeline in a distributed computing system;

executing, with at least one processor, an application programming interface (API) to determine whether an input dataset of the data pipeline is available;

executing, with at least one processor, the data pipeline based on the metadata parameters of the data pipeline and based on determining that the input dataset of the data pipeline is available;

modeling, with at least one processor, the data pipeline using the logical representation of the data pipeline;

monitoring, with at least one processor, activity of the data pipeline after the data pipeline is executed, wherein monitoring the activity of the data pipeline comprises:

receiving checkpoints associated with activity of the data pipeline after the data pipeline is executed;

performing, with at least one processor, an automated scanning operation on the data pipeline to obtain data associated with an error of the data pipeline;

determining, with at least one processor, a type of a failure of the data pipeline based on the data associated with the error of the data pipeline; and

performing, with at least one processor, an automatic restatement operation based on the type of the failure of the data pipeline, wherein the automatic restatement operation comprises an automatic operation for restarting the data pipeline from a specific checkpoint.

2. The computer implemented method of claim 1, further comprising:

providing data associated with a status of the data pipeline in a user interface (UI).

3. The computer implemented method of claim 1, further comprising:

determining data quality metrics associated with the data pipeline based on an output dataset of the data pipeline; and

displaying the data quality metrics associated with the data pipeline in a UI.

4. The computer implemented method of claim 1, further comprising:

executing one or more data flow activities using the data pipeline based on executing the data pipeline.

5. The computer implemented method of claim 4, further comprising:

determining one or more data parameters associated with the one or more data flow activities based on executing the one or more data flow activities; and

transmitting a status message that includes the one or more data parameters associated with the one or more data flow activities.

6. The computer implemented method of claim 1, wherein the metadata parameters comprise:

a log of one or more input datasets;

wherein the method further comprising:

generating an executable file based on the metadata parameters of the data pipeline;

wherein executing the data pipeline comprises:

executing the executable file for the data pipeline based on determining that the one or more input datasets of the data pipeline are available.

7. A system for determining a characteristic of a data pipeline, comprising:

at least one processor, wherein the at least one processor is programmed or configured to:

receive metadata parameters for a data pipeline, wherein the metadata parameters for the data pipeline comprise at least one of the following: data associated with one or more input datasets of a data pipeline, data associated with transformation logic of a data pipeline, data associated with a mapping of an input to an output of a data pipeline, data associated with one or more output datasets of a data pipeline, or any combination thereof;

store the metadata parameters in a data repository;

generate a logical representation of the data pipeline based on the metadata parameters;

use the metadata parameters to instantiate the data pipeline in a distributed computing system;

execute an application programming interface (API) to determine whether an input dataset of the data pipeline is available;

execute the data pipeline based on the metadata parameters of the data pipeline and based on determining that the input dataset of the data pipeline is available;

model the data pipeline using the logical representation of the data pipeline;

monitor activity of the data pipeline after the data pipeline is executed, wherein, when monitoring the activity of the data pipeline, the at least one processor is programmed or configured to:

receive checkpoints associated with activity of the data pipeline after the data pipeline is executed;

perform an automated scanning operation on the data pipeline to obtain data associated with an error of the data pipeline;

determine a type of a failure of the data pipeline based on the data associated with the error of the data pipeline; and

perform an automatic restatement operation based on the type of the failure of the data pipeline, wherein the automatic restatement operation comprises an automatic operation for restarting the data pipeline from a specific checkpoint.

8. The system of claim 7, wherein the at least one processor is further programmed or configured to:

provide data associated with a status of the data pipeline in a user interface (UI).

9. The system of claim 7, wherein the at least one processor is further programmed or configured to:

determine data quality metrics associated with the data pipeline based on an output dataset of the data pipeline; and

display the data quality metrics associated with the data pipeline in a UI.

10. The system of claim 7, wherein the processor is further programmed or configured to:

execute one or more data flow activities using the data pipeline based on executing the data pipeline.

11. The system of claim 10, wherein the at least one processor is further programmed or configured to:

determine one or more data parameters associated with the one or more data flow activities based on executing the one or more data flow activities; and

transmit a status message that includes the one or more data parameters associated with the one or more data flow activities.

12. The system of claim 7, wherein the metadata parameters comprise:

a log of one or more input datasets;

wherein the at least one processor is further programmed or configured to:

generate an executable file based on the metadata parameters of the data pipeline;

wherein, when executing the data pipeline, the at least one processor is programmed or configured to:

execute the executable file for the data pipeline based on determining that the one or more input datasets of the data pipeline are available.

13. A computer program product, the computer program product comprising at least one non-transitory computer readable medium including one or more instructions that, when executed by at least one processor, cause the at least one processor to:

receive metadata parameters for a data pipeline, wherein the metadata parameters for the data pipeline comprise at least one of the following: data associated with one or more input datasets of a data pipeline, data associated with transformation logic of a data pipeline, data associated with a mapping of an input to an output of a data pipeline, data associated with one or more output datasets of a data pipeline, or any combination thereof;

store the metadata parameters in a data repository;

generate a logical representation of the data pipeline based on the metadata parameters;

use the metadata parameters to instantiate the data pipeline in a distributed computing system;

execute an application programming interface (API) to determine whether an input dataset of the data pipeline is available;

execute the data pipeline based on the metadata parameters of the data pipeline and based on determining that the input dataset of the data pipeline is available;

model the data pipeline using the logical representation of the data pipeline;

monitor activity of the data pipeline after the data pipeline is executed, wherein, the one or more instructions that cause the at least one processor to monitor the activity of the data pipeline, cause the at least one processor to:

receive checkpoints associated with activity of the data pipeline after the data pipeline is executed;

perform an automated scanning operation on the data pipeline to obtain data associated with an error of the data pipeline;

determine a type of a failure of the data pipeline based on the data associated with the error of the data pipeline; and

perform an automatic restatement operation based on the type of the failure of the data pipeline, wherein the automatic restatement operation comprises an automatic operation for restarting the data pipeline from a specific checkpoint.

14. The computer program product of claim 13, wherein the one or more instructions further cause the at least one processor to:

provide data associated with a status of the data pipeline in a user interface (UI).

15. The computer program product of claim 13, wherein the one or more instructions further cause the at least one processor to:

determine data quality metrics associated with the data pipeline based on an output dataset of the data pipeline; and

display the data quality metrics associated with the data pipeline in a UI.

16. The computer program product of claim 13, wherein the one or more instructions further cause the at least one processor to:

execute one or more data flow activities using the data pipeline based on executing the data pipeline.

17. The computer program product of claim 16, wherein the one or more instructions further cause the at least one processor to:

determine one or more data parameters associated with the one or more data flow activities based on executing the one or more data flow activities; and

transmit a status message that includes the one or more data parameters associated with the one or more data flow activities.