WHAT IS CLAIMED IS:

1. A method, comprising:

accessing ~~receiving~~ an original file comprising a set of serialized objects;

identifying, based on the original file, an original pipeline comprising a plurality of transformations;

determining a first computing cost for a first transformation of the plurality of transformations;

generating a modified first transformation by modifying the first transformation using a ~~predefined~~defined optimization;

determining a second computing cost of the modified first transformation; and

upon determining that the second computing cost is lower than the first computing cost, replacing, in the original pipeline, the first transformation with the ~~optimized~~ modified first transformation.

2. The method of claim 1, wherein identifying the original pipeline comprises generating a data processing graph, wherein each respective vertex in the data processing graph represents a respective transformation of the plurality of transformations, and wherein each respective edge in the data processing graph specifies data flow in the original pipeline.

3. The method of claim 2, wherein generating the data processing graph comprises:

instantiating the original pipeline based on the original file;

executing the original pipeline on sample data;

identifying the plurality of transformations based on the execution; and

identifying a sequence of the plurality of transformations based on the execution.

4. The method of claim 2, wherein generating the data processing graph comprises evaluating the original file to identify the plurality of transformations and the data flow in the original pipeline, without instantiating the original pipeline.

5. The method of claim 1, wherein determining the first computing cost for the first transformation comprises:

executing the first transformation on sample data; and

evaluating the execution to determine a latency of performing the first transformation.

6. The method of claim 1, wherein the ~~predefined~~defined optimization comprises increased parallelization in the first transformation.

7. The method of claim 1, wherein the ~~predefined~~defined optimization comprises a one-row optimization for the first transformation.

8. The method of claim 1, the method further comprising:

generating an optimized file of serialized objects based on the original pipeline and the ~~optimized~~ modified first transformation.

9. One or more computer-readable storage media collectively containing computer program code that, when executed by operation of one or more computer processors, performs an operation comprising:

accessing~~receiving~~ an original file comprising a set of serialized objects;

identifying, based on the original file, an original pipeline comprising a plurality of transformations;

determining a first computing cost for a first transformation of the plurality of transformations;

generating a modified first transformation by modifying the first transformation using a ~~predefined~~defined optimization;

determining a second computing cost of the modified first transformation; and

upon determining that the second computing cost is lower than the first computing cost, replacing, in the original pipeline, the first transformation with the ~~optimized~~ modified first transformation.

10. The computer-readable storage media of claim 9, wherein identifying the original pipeline comprises generating a data processing graph, wherein each respective vertex in the data processing graph represents a respective transformation of the plurality of transformations, and wherein each respective edge in the data processing graph specifies data flow in the original pipeline.

11. The computer-readable storage media of claim 10, wherein generating the data processing graph comprises:

instantiating the original pipeline based on the original file;

executing the original pipeline on sample data;

identifying the plurality of transformations based on the execution; and

identifying a sequence of the plurality of transformations based on the execution.

12. The computer-readable storage media of claim 10, wherein generating the data processing graph comprises evaluating the original file to identify the plurality of transformations and the data flow in the original pipeline, without instantiating the original pipeline.

13. The computer-readable storage media of claim 9, wherein determining the first computing cost for the first transformation comprises:

executing the first transformation on sample data; and

evaluating the execution to determine a latency of performing the first transformation.

14. The computer-readable storage media of claim 9, wherein the ~~predefined~~defined optimization comprises (i) increased parallelization in the first transformation, or (ii) a one-row optimization for the first transformation.

15. The computer-readable storage media of claim 9, the operation further comprising:

generating an optimized file of serialized objects based on the original pipeline and the ~~optimized~~ modified first transformation.

16. A system comprising:

one or more computer processors; and

one or more memories collectively containing one or more programs which when executed by the one or more computer processors performs an operation, the operation comprising:

~~receiving~~ accessing an original file comprising a set of serialized objects;

identifying, based on the original file, an original pipeline comprising a plurality of transformations;

determining a first computing cost for a first transformation of the plurality of transformations;

generating a modified first transformation by modifying the first transformation using a ~~predefined~~defined optimization;

determining a second computing cost of the modified first transformation; and

upon determining that the second computing cost is lower than the first computing cost, replacing, in the original pipeline, the first transformation with the ~~optimized~~ modified first transformation.

17. The system of claim 16, wherein identifying the original pipeline comprises generating a data processing graph, wherein each respective vertex in the data processing graph represents a respective transformation of the plurality of transformations, and wherein each respective edge in the data processing graph specifies data flow in the original pipeline.

18. The system of claim 17, wherein generating the data processing graph comprises:

instantiating the original pipeline based on the original file;

executing the original pipeline on sample data;

identifying the plurality of transformations based on the execution; and

identifying a sequence of the plurality of transformations based on the execution.

19. The system of claim 17, wherein generating the data processing graph comprises evaluating the original file to identify the plurality of transformations and the data flow in the original pipeline, without instantiating the original pipeline.

20. The system of claim 16, wherein determining the first computing cost for the first transformation comprises:

executing the first transformation on sample data; and

evaluating the execution to determine a latency of performing the first transformation.