What is claimed is:

1. A computer-implemented method for analyzing a computer program ecosystem including multiple computer programs, the method including:

performing a static analysis of the computer program ecosystem, including identifying static dependencies among elements of the computer program ecosystem based on values of parameters in one or more parameter sets associated with the computer program ecosystem, the elements of the computer program ecosystem including the multiple computer programs of the computer program ecosystem and data resources associated with the computer programs;

performing a runtime analysis of the computer program ecosystem, including identifying one or more of the elements of the computer program ecosystem that were utilized during execution of the computer program ecosystem to process data records;

performing a schedule analysis of the computer program ecosystem, including identifying a computer program of the computer program ecosystem that has a schedule dependency from another computer program of the computer program ecosystem;

identifying a subset of the elements of the computer program ecosystem as an ecosystem unit based on the results of the static analysis, the runtime analysis, and the schedule analysis; and

migrating the ecosystem unit from a first computer system to a second computer system, testing the ecosystem unit, or both.

2. The method of claim 1, in which performing the runtime analysis includes obtaining runtime log information generated by execution of the computer program ecosystem; and analyzing the runtime log information to identify the one or more elements of the computer program ecosystem that were utilized during execution of the computer program ecosystem.

3. The method of claim 2, in which analyzing the runtime log information includes analyzing the values of parameters in the runtime log information to identify one or more computer programs that were executed during execution of the computer program ecosystem.

4. The method of claim 3, in which identifying the subset of elements as the ecosystem unit includes identifying the computer programs that were executed during execution of the computer program ecosystem as parts of the ecosystem unit.

5. The method of claim 3 or 4, in which identifying one or more computer programs of the computer program ecosystem that were executed during execution of the computer program ecosystem includes identifying runtime dependencies among the executed computer programs.

6. The method of any of claims 3 to 5, in which identifying the runtime dependencies among the executed computer programs includes identifying the runtime dependencies based on the values of the parameters in the runtime log information.

7. The method of any of claims 3 to 6, in which analyzing the values of the parameters in the runtime log information includes generalizing one or more of the values.

8. The method of any of claims 3 to 7, in which analyzing the runtime log information includes analyzing the values of parameters in the runtime log information to identify one or more data resources that were accessed during execution of the computer program ecosystem.

9. The method of claim 8, including analyzing the values of the parameters in the runtime log information to determine the name of each data resource that was accessed during execution of the computer program ecosystem.

10. The method of claim 8 or 9, in which identifying the subset of elements as the ecosystem unit includes identifying the data resources that were accessed during execution of the computer program ecosystem as parts of the ecosystem unit.

11. The method of any of the preceding claims, in which identifying a subset of the elements of the computer program ecosystem as an ecosystem unit includes including, in the ecosystem unit, the one or more elements that were utilized during execution of the computer program ecosystem.

12. The method of any of the preceding claims, in which performing a schedule analysis includes obtaining schedule information from a scheduler, in which the scheduler is distinct from the computer program ecosystem.

13. The method of any of the preceding claims, in which performing a schedule analysis includes identifying a first computer program of the computer program ecosystem that is scheduled to be executed after execution of a second computer program of the computer program ecosystem.

14. The method of claim 13, in which identifying the subset of elements as the ecosystem unit includes identifying the first computer program and the second computer program as part of the ecosystem unit.

15. The method of any of the preceding claims, including executing the computer program ecosystem to process data.

16. The method of claim 15, in which executing the computer program ecosystem includes instantiating one or more of the computer programs of the computer program ecosystem according to values of parameters in one or more of the parameter sets associated with the computer program ecosystem.

17. The method of claim 16, in which executing the computer program ecosystem includes executing the one or more instantiated computer programs to process the data.

18. The method of claim 17, in which the runtime log information includes information indicative of the one or more executed instantiated computer programs.

19. The method of any of claims 15 to 18, in which executing the computer program ecosystem includes accessing one or more of the data resources associated with the computer program ecosystem.

20. The method of claim 19, in which the runtime log information includes information indicative of the one or more accessed data resources.

21. The method of claim 19 or 20, in which accessing a data resource associated with the computer program ecosystem includes reading data from one or more of a file, a database, and an object.

22. The method of any of claims 19 to 21, in which accessing a data resource associated with the computer program ecosystem includes writing data to one or more of a file, a database, and an object.

23. The method of any of claims 19 to 22, in which executing the computer program ecosystem includes generating the runtime log information.

24. The method of claim 23, in which generating the runtime log information includes writing the runtime log information to a log file.

25. The method of any of the preceding claims, in which the one or more computer programs include dataflow graphs.

26. A computer system for analyzing a computer program ecosystem including multiple computer programs, the computer system including:

one or more processors coupled to a memory, the one or more processors configured to perform the steps of the method of any one of claims 1 to 25.

27. A computer readable medium storing instructions for causing a computer system to analyze a computer program ecosystem including multiple computer programs, the instructions causing the computer system to perform the steps of the method of any one of claims 1 to 25.

28. A computer system for analyzing a computer program ecosystem including multiple computer programs, the computer system including:

means for performing a static analysis of the computer program ecosystem, including identifying static dependencies among elements of the computer program ecosystem based on values of parameters in one or more parameter sets associated with the computer program ecosystem, the elements of the computer program ecosystem including the multiple computer programs of the computer program ecosystem and data resources associated with the computer programs;

means for performing a runtime analysis of the computer program ecosystem, including identifying one or more of the elements of the computer program ecosystem that were utilized during execution of the computer program ecosystem to process data records;

means for performing a schedule analysis of the computer program ecosystem, including identifying a computer program of the computer program ecosystem that has a schedule dependency from another computer program of the computer program ecosystem;

means for identifying a subset of the elements of the computer program ecosystem as an ecosystem unit based on the results of the static analysis, the runtime analysis, and the schedule analysis; and

means for migrating the ecosystem unit from a first computer system to a second computer system, testing the ecosystem unit, or both.