

What is claimed is:

1. A method for processing a graph-based program specification, the method including:
receiving the graph-based program specification, the graph-based program specification including:
a plurality of components, each corresponding to a processing task and
including one or more ports, the ports including scalar data ports for sending or receiving
a single data element and collection data ports for sending or receiving a collection of
multiple data elements; and
one or more links, each link of the one or more links connecting an output port of an upstream
component of the plurality of components to an input port of a downstream component of
the plurality of components; and
processing the graph-based program specification to generate prepared code representing
subsets of the plurality of components of the graph-based program specification, the
processing including:
identifying one or more subset boundaries, including identifying one or more links
connecting a collection data port of a component to a scalar data port of a
component;
forming the subsets based on the identified subset boundaries; and
generating prepared code for each formed subset that when used for execution by a
runtime system causes processing tasks corresponding to the components in
each formed subset to be performed;
wherein the one or more subset boundaries include one or more subset entry points, the
one or more subset boundaries include one or more subset exit points, and
forming the subsets based on the identified subset boundaries includes forming
the subsets based on identified subset entry points and identified subset exit
points.
2. The method of claim 1 wherein identifying one or more subset boundaries includes identifying one
or more subset entry points, including identifying at least one link connecting a collection data output port
of an upstream component to a scalar data input port of a downstream component.
3. The method of claim 1 wherein identifying one or more subset boundaries includes identifying one
or more subset exit points, including identifying at least one link connecting a scalar data output port of an
upstream component to a collection data input port of a downstream component.

4. The method of claim 1 wherein each subset formed includes one or more components, no more than one subset entry point, and one or more subset exit points.

5. The method of claim 1 wherein forming the subsets includes traversing the components of the graph-based program specification while maintaining a record of traversed subset entry points and subset exit points, and associating each component of the graph-based program specification with a single subset identifier determined from the record of traversed subset entry points and subset exit points.

6. The method of claim 5 wherein each subset identifier associated with an identified subset of the plurality of component is unique.

7. The method of claim 5 wherein the record of traversed subset entry points and subset exit points is maintained as a path of identifier values.

8. The method of claim 7 wherein the path of identifier values includes a string of identifier values separated from each other by a separation character.

9. The method of claim 1 wherein forming the subsets includes:
associating a first component of the graph-based program specification with a subset identifier;
propagating the subset identifier to components downstream from the first component; and
modifying the subset identifier during propagation of the subset identifier based on the identified subset entry points and the identified subset exit points.

10. The method of claim 9 wherein modifying the subset identifier during propagation of the subset identifier includes:

changing a value of the subset identifier from a first subset identifier value to a second subset identifier value associated with a subset entry point upon traversing the subset entry point; and
changing the value of the subset identifier to the first subset identifier value upon traversing a subset exit point associated with the subset entry point.

11. A non-transitory medium storing software, for processing a graph-based program specification, the software including instructions for causing a computing system to:

receive the graph-based program specification, the graph-based program specification including:
a plurality of components, each corresponding to a processing task and including one or more

ports, the ports including scalar data ports for sending or receiving a single data element and collection data ports for sending or receiving a collection of multiple data elements; and

one or more links, each link of the one or more links connecting an output port of an upstream component of the plurality of components to an input port of a downstream component of the plurality of components; and

process the graph-based program specification to generate prepared code representing subsets of the plurality of components of the graph-based program specification, the processing including:

- identifying one or more subset boundaries, including identifying one or more links connecting a collection data port of a component to a scalar data port of a component;
- forming the subsets based on the identified subset boundaries; and
- generating prepared code for each formed subset that when used for execution by a runtime system causes processing tasks corresponding to the components in each formed subset to be performed;

wherein the one or more subset boundaries include one or more subset entry points, the one or more subset boundaries include one or more subset exit points, and forming the subsets based on the identified subset boundaries includes forming the subsets based on identified subset entry points and identified subset exit points.

12. The non-transitory medium of claim 11 wherein identifying one or more subset boundaries includes identifying one or more subset entry points, including identifying at least one link connecting a collection data output port of an upstream component to a scalar data input port of a downstream component.

13. The non-transitory medium of claim 11 wherein identifying one or more subset boundaries includes identifying one or more subset exit points, including identifying at least one link connecting a scalar data output port of an upstream component to a collection data input port of a downstream component.

14. The non-transitory medium of claim 11 wherein identifying one or more subset boundaries includes identifying one or more subset exit points, including identifying at least one link connecting a scalar data output port of an upstream component to a collection data input port of a downstream component.

15. The non-transitory medium of claim 11 wherein forming the subsets includes traversing the components of the graph-based program specification while maintaining a record of traversed subset entry points and subset exit points, and associating each component of the graph-based program

specification with a single subset identifier determined from the record of traversed subset entry points and subset exit points.

16. The non-transitory medium of claim 15 wherein each subset identifier associated with an identified subset of the plurality of component is unique.

17. The non-transitory medium of claim 15 wherein the record of traversed subset entry points and subset exit points is maintained as a path of identifier values.

18. The non-transitory medium of claim 17 wherein the path of identifier values includes a string of identifier values separated from each other by a separation character.

19. The non-transitory medium of claim 11 wherein forming the subsets includes:
associating a first component of the graph-based program specification with a subset identifier;
propagating the subset identifier to components downstream from the first component; and
modifying the subset identifier during propagation of the subset identifier based on the identified subset entry points and the identified subset exit points.

20. The non-transitory medium of claim 19 wherein modifying the subset identifier during propagation of the subset identifier includes:

changing a value of the subset identifier from a first subset identifier value to a second subset identifier value associated with a subset entry point upon traversing the subset entry point; and
changing the value of the subset identifier to the first subset identifier value upon traversing a subset exit point associated with the subset entry point.

21. A computing system for processing a graph-based program specification, the computing system including:

at least one input device or port configured to receive the graph-based program specification, the graph-based program specification including:

a plurality of components, each corresponding to a processing task and including one or more ports, the ports including scalar data ports for sending or receiving a single data element and collection data ports for sending or receiving a collection of multiple data elements;
and

one or more links, each link of the one or more links connecting an output port of an upstream component of the plurality of components to an input port of a downstream component of the plurality of components; and

at least one processor configured to process the graph-based program specification to generate

prepared code representing subsets of the plurality of components of the graph-based program specification, the processing including:

identifying one or more subset boundaries, including identifying one or more links connecting a collection data port of a component to a scalar data port of a component; forming the subsets based on the identified subset boundaries; and

generating prepared code for each formed subset that when used for execution by a runtime system causes processing tasks corresponding to the components in each formed subset to be performed;

wherein the one or more subset boundaries include one or more subset entry points, the one or more subset boundaries include one or more subset exit points, and forming the subsets based on the identified subset boundaries includes forming the subsets based on identified subset entry points and identified subset exit points.

22. The computing system of claim 21 wherein identifying one or more subset boundaries includes identifying one or more subset entry points, including identifying at least one link connecting a collection data output port of an upstream component to a scalar data input port of a downstream component.

23. The computing system of claim 21 wherein identifying one or more subset boundaries includes identifying one or more subset exit points, including identifying at least one link connecting a scalar data output port of an upstream component to a collection data input port of a downstream component.

24. The computing system of claim 21 wherein each subset formed includes one or more components, no more than one subset entry point, and one or more subset exit points.

25. The computing system of claim 21 wherein forming the subsets includes traversing the components of the graph-based program specification while maintaining a record of traversed subset entry points and subset exit points, and associating each component of the graph-based program specification with a single subset identifier determined from the record of traversed subset entry points and subset exit points.

26. The computing system of claim 25 wherein each subset identifier associated with an identified subset of the plurality of component is unique.

27. The computing system of claim 25 wherein the record of traversed subset entry points and subset exit points is maintained as a path of identifier values.

28. The computing system of claim 27 wherein the path of identifier values includes a string of identifier values separated from each other by a separation character.

29. The computing system of claim 21 wherein forming the subsets includes:
associating a first component of the graph-based program specification with a subset identifier;
propagating the subset identifier to components downstream from the first component; and
modifying the subset identifier during propagation of the subset identifier based on the identified subset entry points and the identified subset exit points.

30. The computing system of claim 29 wherein modifying the subset identifier during propagation of the subset identifier includes:
changing a value of the subset identifier from a first subset identifier value to a second subset identifier value associated with a subset entry point upon traversing the subset entry point; and
changing the value of the subset identifier to the first subset identifier value upon traversing a subset exit point associated with the subset entry point.

31. A computing system for processing a graph-based program specification, the computing system including:
means for receiving the graph-based program specification, the graph-based program specification including:
a plurality of components, each corresponding to a processing task and including one or more ports, the ports including scalar data ports for sending or receiving a single data element and collection data ports for sending or receiving a collection of multiple data elements;
and
one or more links, each link of the one or more links connecting an output port of an upstream component of the plurality of components to an input port of a downstream component of the plurality of components; and
means for processing the graph-based program specification to generate prepared code representing subsets of the plurality of components of the graph-based program specification, the processing including:
identifying one or more subset boundaries, including identifying one or more links connecting a collection data port of a component to a scalar data port of a component;
forming the subsets based on the identified subset boundaries; and
generating prepared code for each formed subset that when used for execution by a

runtime system causes processing tasks corresponding to the components in each formed subset to be performed;
wherein the one or more subset boundaries include one or more subset entry points, the one or more subset boundaries include one or more subset exit points, and forming the subsets based on the identified subset boundaries includes forming the subsets based on identified subset entry points and identified subset exit points.