

AMENDED CLAIMS

received by the International Bureau on 18 February 2016 (18.02.2016)

1. (Currently Amended) 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 for sending or receiving one or more 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 a plurality of subset boundaries between components in different subsets based at least in part on characteristics of linked components;

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 that formed subset to be performed according to information embedded in the prepared code for that formed subset, where the runtime system instantiates a number of instances of that formed subset, and the number is determined dynamically during runtime based at least in part on performance of the runtime system in processing a set of multiple data elements.

2. (Original) 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 boundaries, and associating each component of the graph-based program specification with a single subset identifier determined from the record of traversed subset boundaries.

3. (Original) The method of claim 2 wherein each subset identifier associated with an identified subset of the plurality of component is unique.

4. (Original) The method of claim 2 wherein the record of traversed subset boundaries is maintained as a path of identifier values.

5. (Original) The method of claim 4 wherein the path of identifier values includes a string of identifier values separated from each other by a separation character.

6. (Original) 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 boundaries.

7. (Original) The method of claim 6 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 first subset boundary upon traversing the first subset boundary; and

changing the value of the subset identifier to the first subset identifier value upon traversing a second subset boundary associated with the first subset boundary.

8. (Original) The method of any preceding claim wherein identifying one or more subset boundaries based at least in part on characteristics of linked components includes identifying a subset boundary based on a link between a port of a first type on an upstream component and a port of a second type on a downstream component.

9. (Original) The method of any preceding claim wherein identifying one or more subset boundaries based at least in part on characteristics of linked components includes identifying a subset boundary based on a determined type of a link between an upstream component and a downstream component, where the determined type of link is one of multiple different types of links between components.

10. (Original) The method of any preceding claim wherein generating the prepared code for each formed subset includes embedding information into the prepared code for at least one formed subset that indicates allowed concurrency among processing tasks corresponding to the components in that formed subset.

11. (Original) The method of any preceding claim wherein generating the prepared code for each formed subset includes embedding information into the prepared code for at least one formed subset that indicates precedence with respect to other formed subsets.

12. (Original) The method of any preceding claim wherein generating the prepared code for each formed subset includes embedding information into the prepared code for at least one formed subset that indicates transactionality of one or more processing tasks corresponding to the components in that formed subset.

13. (Original) The method of any preceding claim wherein generating the prepared code for each formed subset includes embedding information into the prepared code for at least one formed subset that indicates at least one resource to be locked during execution of the prepared code.

14. (Original) The method of any preceding claim wherein generating the prepared code for each formed subset includes embedding information into the prepared code for at least one formed subset that indicates ordering characteristics among data elements processed by one or more processing tasks corresponding to the components in that formed subset.

15. (Original) The method of any preceding claim wherein generating the prepared code for each formed subset includes embedding information into the prepared code for at least one formed subset that indicates a number of data elements to be operated upon by each instance of the formed subset executed using the prepared code.

16. (Currently Amended) Software stored in a non-transitory form on a computer-readable medium, 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 for sending or receiving one or more 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 a plurality of subset boundaries between components in different subsets based at least in part on characteristics of linked components;
- 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 that formed subset to be performed according to information embedded in the prepared code for that formed subset, where the runtime system instantiates a number of instances of that formed subset, and the number is determined dynamically during runtime based at least in part on performance of the runtime system in processing a set of multiple data elements.

17. (Currently Amended) 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 for sending or receiving one or more 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 a plurality of subset boundaries between components in different subsets based at least in part on characteristics of linked components;
- 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 that formed subset to be performed according to information embedded in the prepared code for that formed subset, where the runtime system instantiates a number of instances of that formed subset, and the number is determined dynamically during runtime based at least in part on performance of the runtime system in processing a set of multiple data elements.

18. (New) The method of any of claims 1 to 15 wherein, if there are multiple instances, each of the instances is applied to different respective subsets of data elements in the set of multiple data elements.