



(19) **United States**

(12) **Patent Application Publication**

Ward et al.

(10) **Pub. No.: US 2003/0145294 A1**

(43) **Pub. Date: Jul. 31, 2003**

(54) **VERIFYING INTERCONNECT FABRIC DESIGNS**

Publication Classification

(76) Inventors: **Julie Ann Ward**, Menlo Park, CA (US); **Troy Alexander Shahoumian**, Sunnyvale, CA (US); **John Wilkes**, Palo Alto, CA (US)

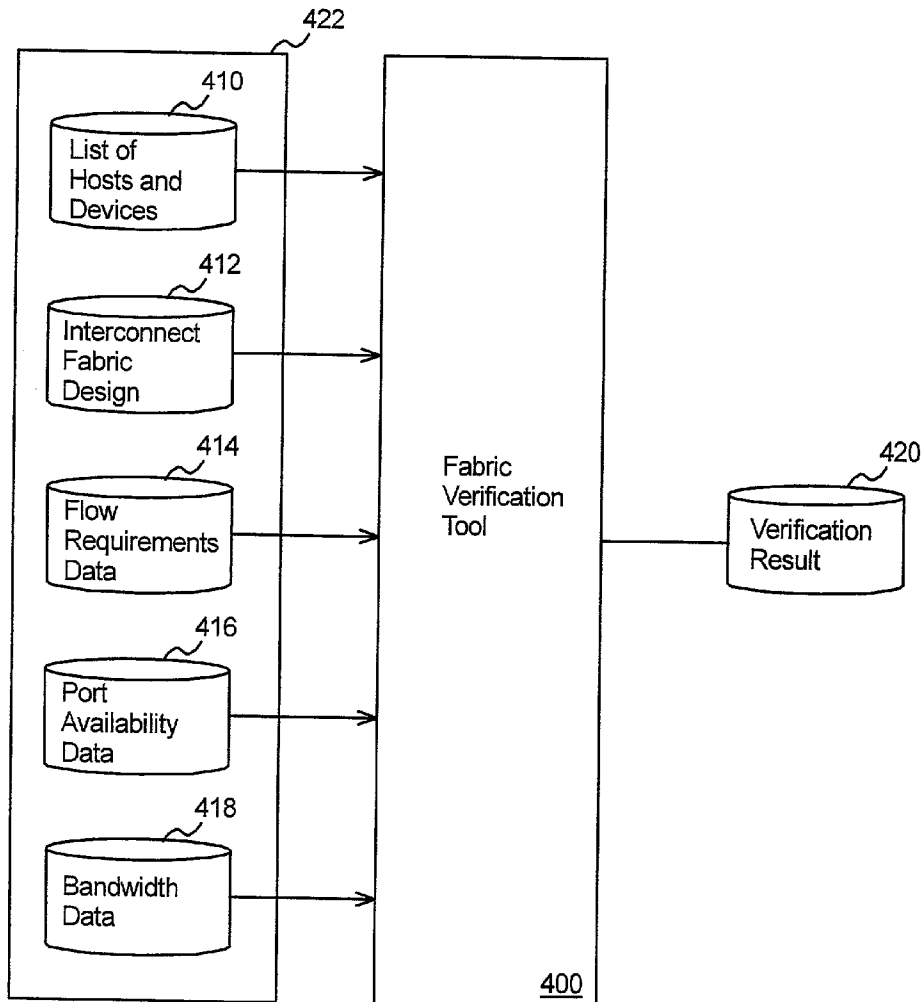
(51) **Int. Cl.⁷** **G06F 17/50**
(52) **U.S. Cl.** **716/5; 703/2**

Correspondence Address:
HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P.O. Box 272400
Fort Collins, CO 80527-2400 (US)

ABSTRACT

A technique for verifying an interconnect fabric design for interconnecting a plurality of network nodes. A design for the interconnect fabric specifies an arrangement of elements of the fabric and flow requirements among the network nodes. The invention programmatically verifies that the flow requirements are satisfied by the design and that the design does not violate constraints on the elements, such as available bandwidth or number of ports. This may also include determining whether the network can continue to satisfy the flow requirements in the event of one or more failures of elements of the interconnect fabric.

(21) Appl. No.: **10/058,258**
(22) Filed: **Jan. 25, 2002**



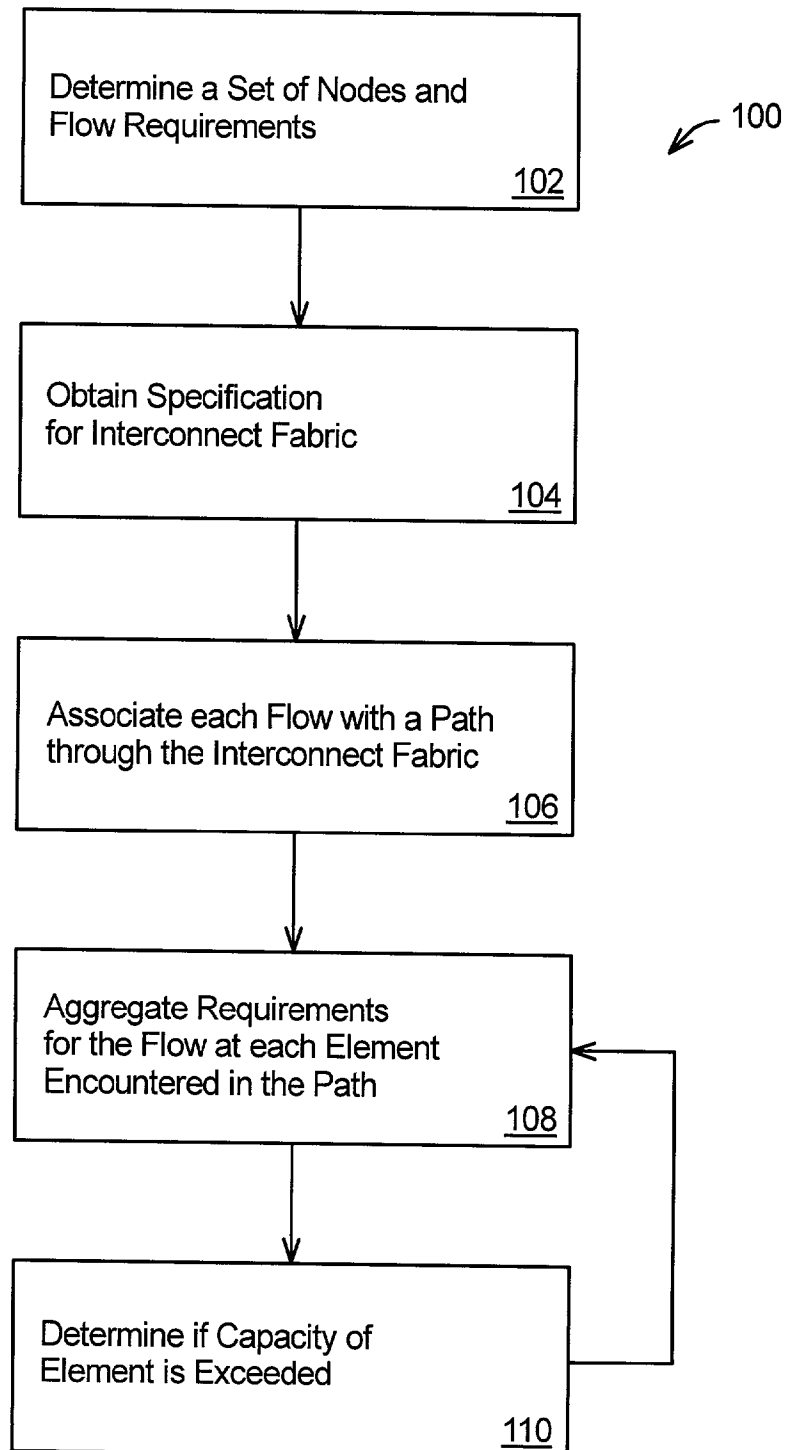


Fig. 1

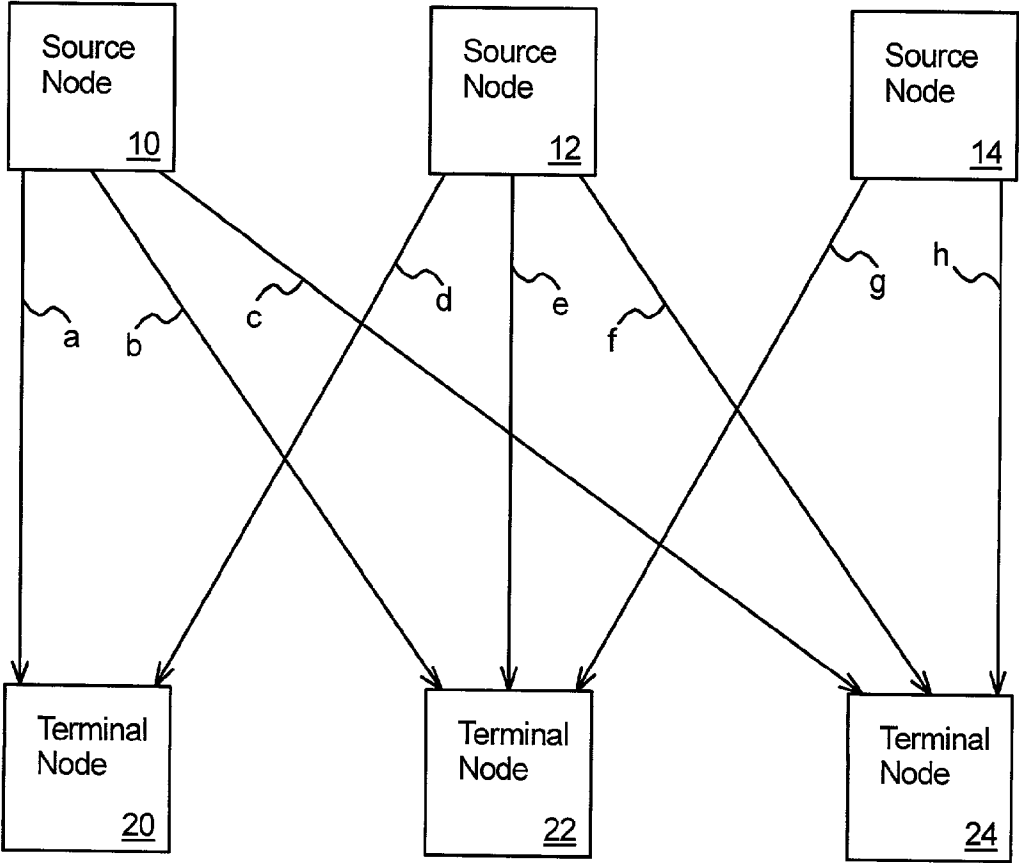


Fig. 2

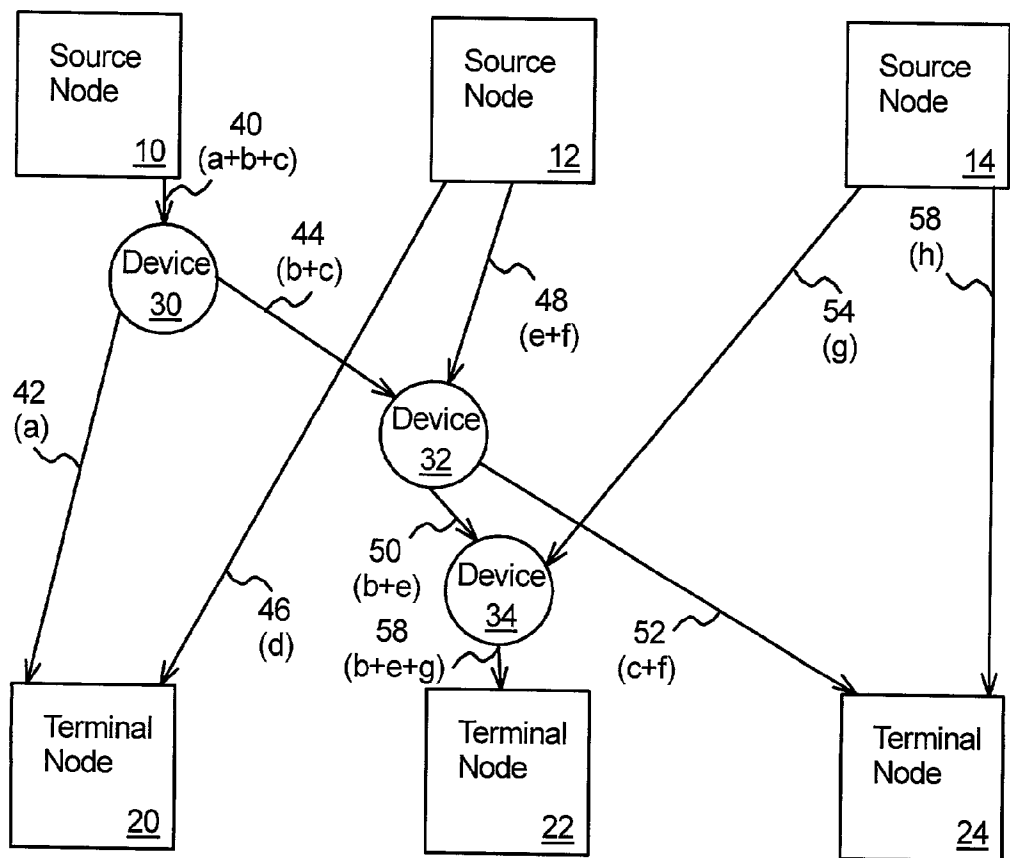


Fig. 3

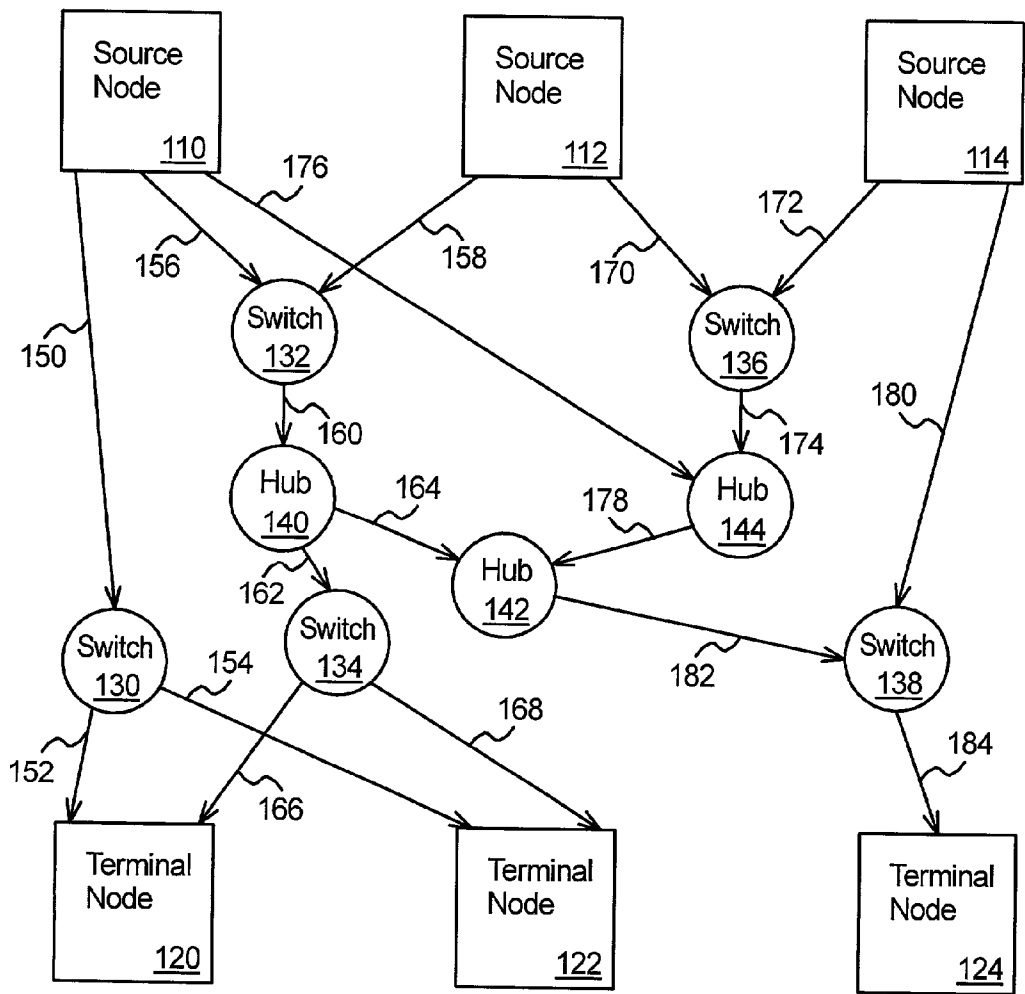


Fig. 4

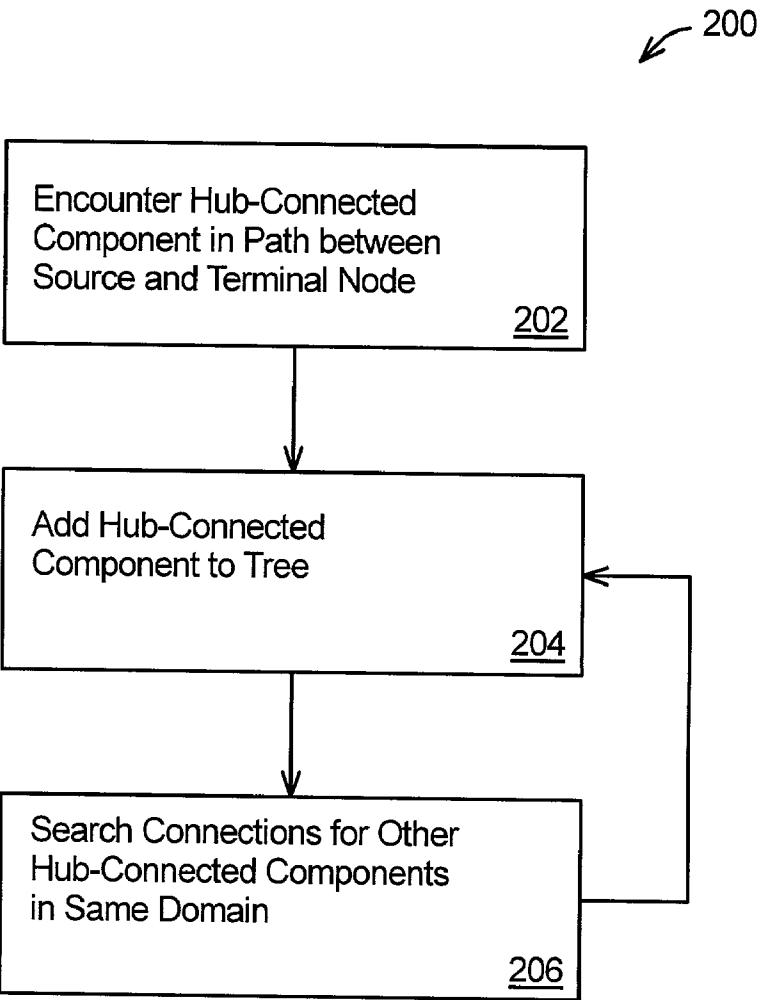


Fig. 5

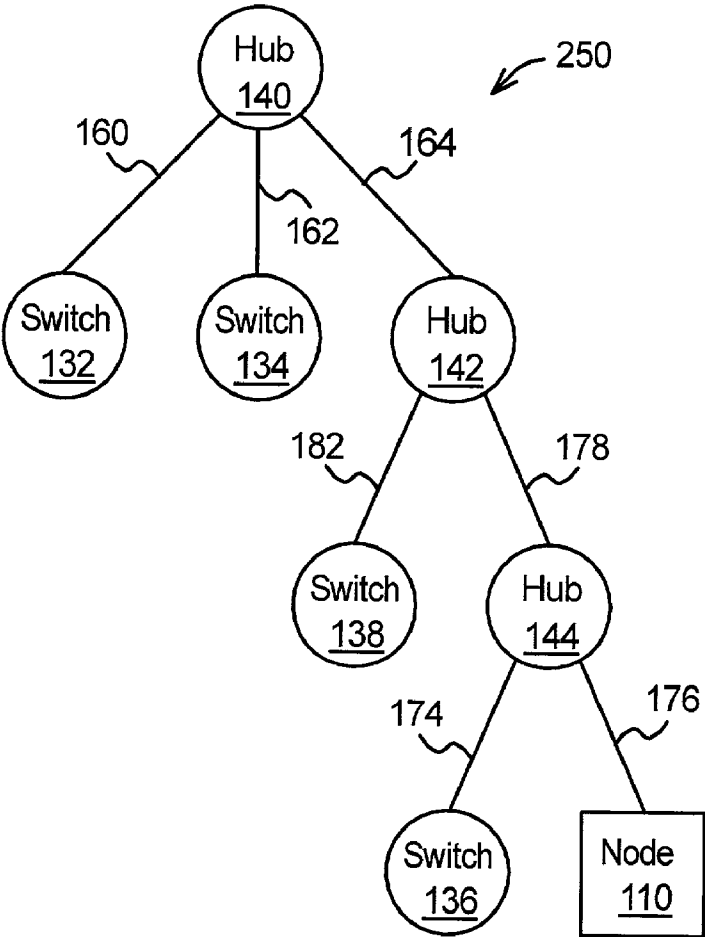


Fig. 6

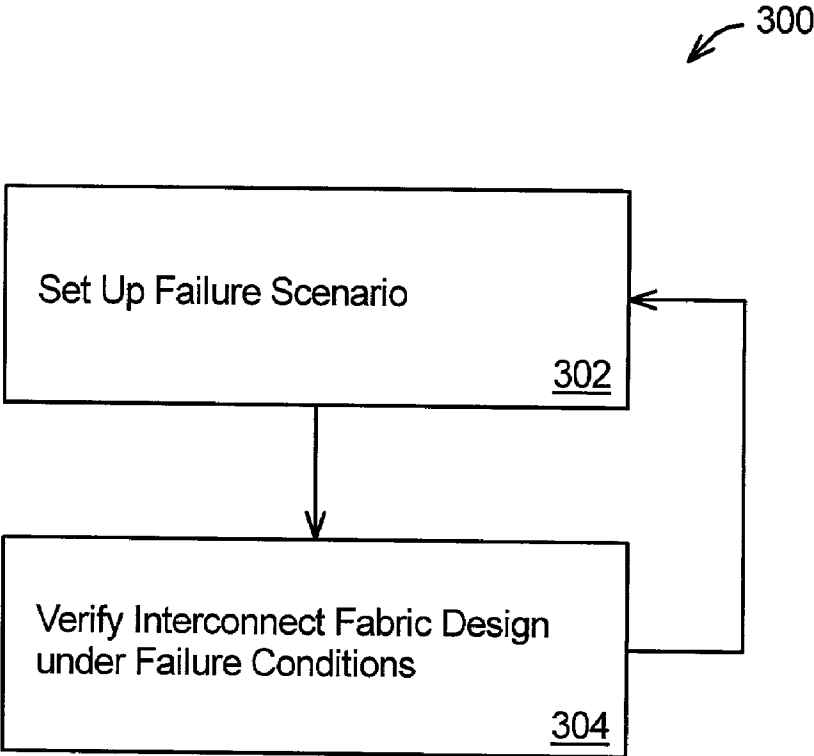


Fig. 7

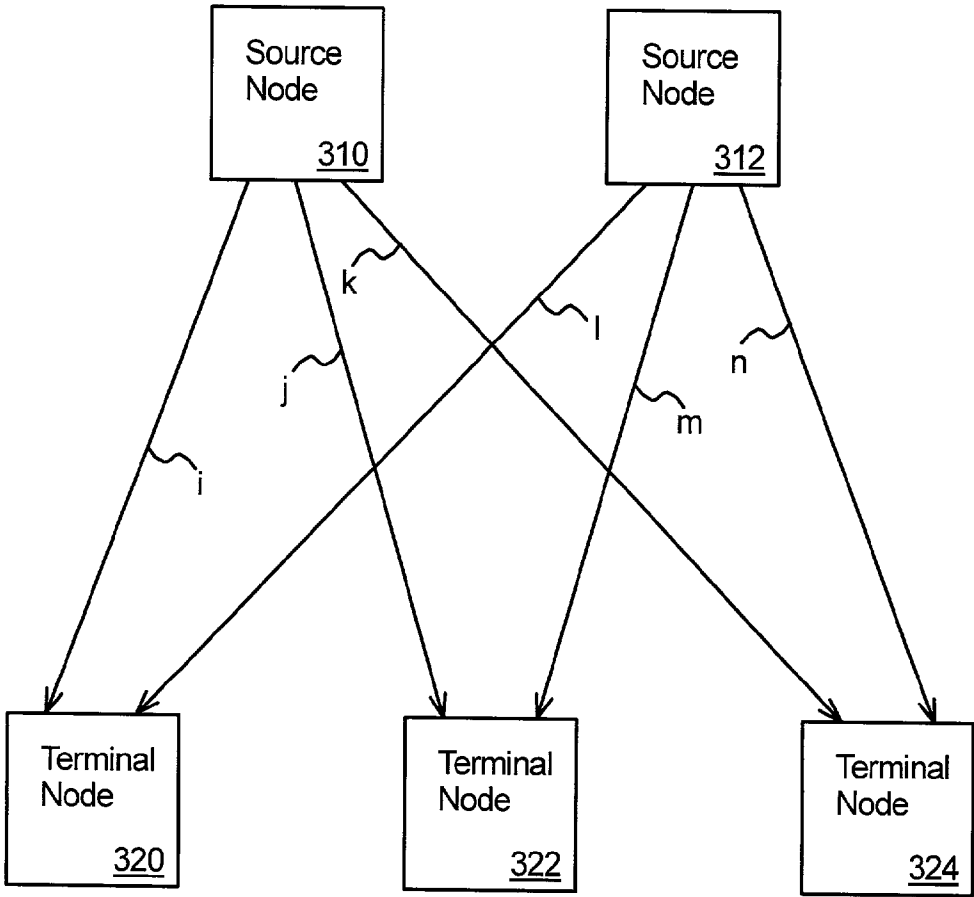


Fig. 8

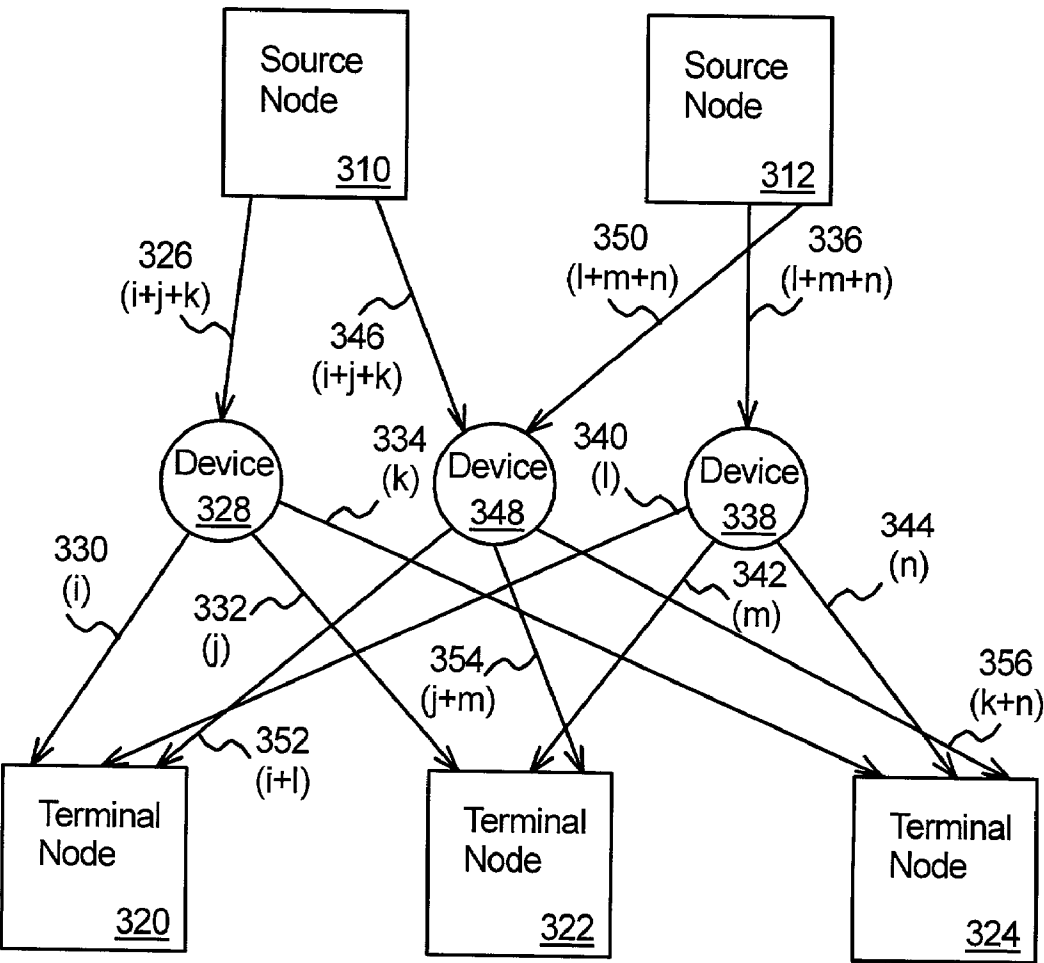


Fig. 9

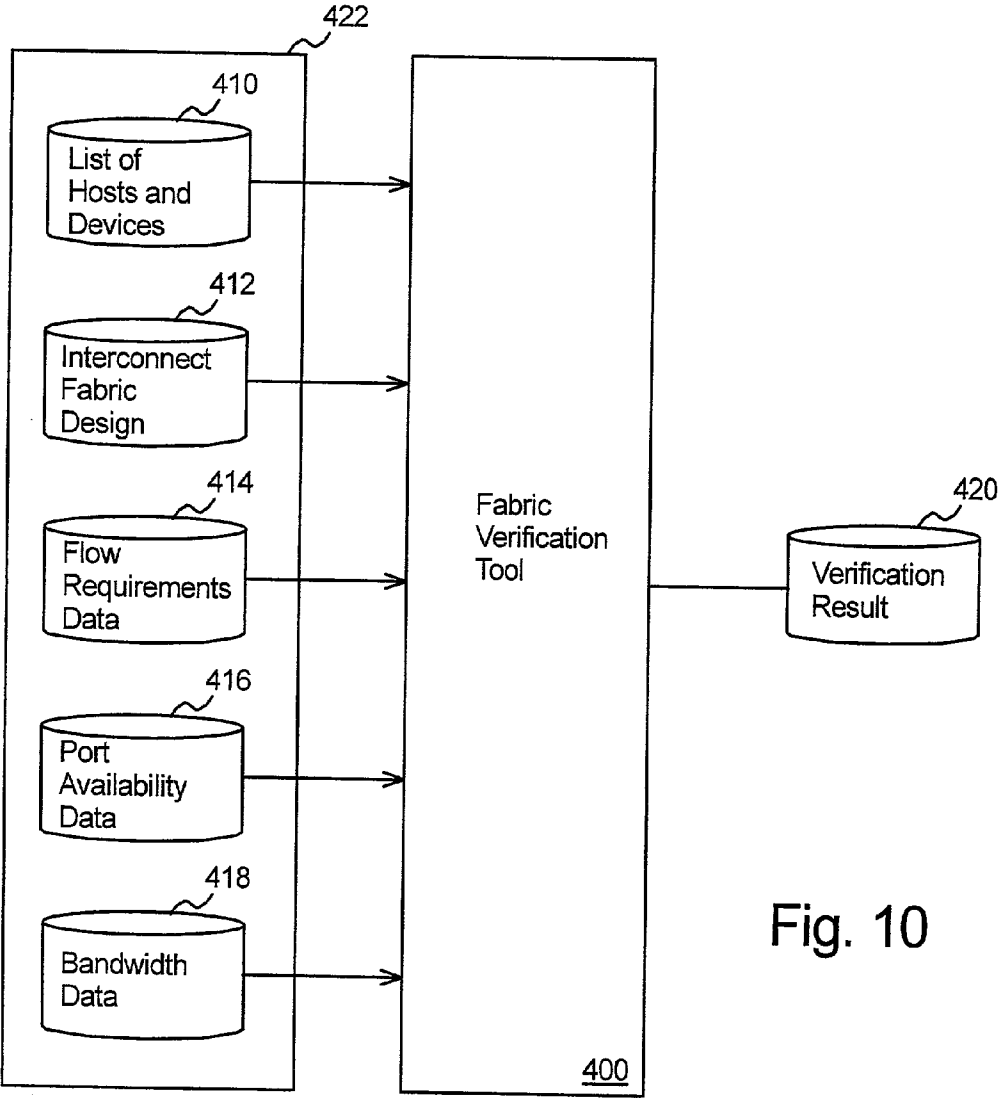


Fig. 10

VERIFYING INTERCONNECT FABRIC DESIGNS

FIELD OF THE INVENTION

[0001] The present invention pertains to the field of networks. More particularly, this invention relates to verification of designs for networks.

BACKGROUND OF THE INVENTION

[0002] An interconnect fabric provides for communication among a set of nodes in a network. Communications originate within the network at a source node and terminate at a terminal node. Thus, a wide variety of networks may be viewed as a set of source nodes that communicate with a set of terminal nodes via an interconnect fabric. For example, a storage area network may be arranged as a set of computers as source nodes which are connected to a set of storage devices as terminal nodes via an interconnect fabric that includes communication links and devices such as hubs, routers, switches, etc. Devices such as hubs, routers, switches, etc., are hereinafter referred to as interconnect devices. Depending on the circumstances, a node may assume the role of source node with respect to some communications and of terminal node for other communications.

[0003] The communication requirements of an interconnect fabric may be characterized in terms of a set of flow requirements. A typical set of flow requirements specifies the required communication bandwidth from each source node to each terminal node. The design of an interconnect fabric usually involves selecting the appropriate arrangement of physical communication links and interconnect devices and related components that will meet the flow requirements.

[0004] Once a design of an interconnect fabric has been obtained, it may be desired to verify that the design actually meets the communication requirements. Prior methods for verifying an interconnect fabric design may be based on manual techniques. Unfortunately, such techniques are usually error prone and time-consuming. Other techniques include simulation of the network design. Simulations, however, can also be time-consuming to set up and to run since they generally require simulation of the network design and of a synthetic load on the network.

[0005] Therefore, what is needed is an improved technique for verifying the design of a network. It is to these ends that the present invention is directed.

SUMMARY OF THE INVENTION

[0006] A technique is disclosed for verifying an interconnect fabric design for interconnecting a plurality of network nodes. A design for the interconnect fabric specifies an arrangement of elements of the fabric and flow requirements among the network nodes. The invention programmatically verifies the design. This may include determining whether the flow requirements are satisfied by the design and whether the design violates constraints on the elements, such as bandwidth capacity and number of available ports. This may also include determining whether the network can continue to satisfy the flow requirements in the event of one or more failures of elements of the interconnect fabric.

[0007] In one aspect of the invention, a computer implemented method is provided for verifying a design for an

interconnect fabric. The design includes an arrangement of interconnect elements for interconnecting a plurality of network nodes. The design also has requirements for a plurality of flows among the network nodes. Each of the plurality of flows is associated with a path for the flow through the interconnect fabric. For each interconnect element in each path, requirements (e.g., bandwidth or a number of ports) associated with each of the corresponding flows is aggregated. A determination is made as to whether the aggregated requirements exceeds a capacity of the interconnect element.

[0008] In another aspect of the present invention, a system is provided for verifying a design for an interconnect fabric. A set of design information includes requirements for a plurality of flows and a design specification. Each of the plurality of flows is associated with a path for the flow through the interconnect fabric. For each interconnect element in each path, a fabric design verification tool aggregates requirements (e.g., bandwidth or a number of ports) associated with each of the corresponding flows and determines whether the aggregated requirements exceeds a capacity of the interconnect element.

[0009] The interconnect elements may include interconnect devices and links. The interconnect devices may be selected from the group consisting of switches and hubs. When the interconnect devices includes a hub, the extent of a domain of hub connected components may be identified. Identifying the extent of the domain of hub connected components may include a depth first search of the interconnect fabric for the hub connected components and may include constructing a tree data structure wherein a hub occupies a position in the tree and a other interconnect elements connected to the hub occupy positions in the tree one level down from the hub.

[0010] Requirements of ports may also be aggregated for each of the plurality of flows and a determination made as to whether a number of available ports of one or more of the interconnect elements is exceeded by the aggregated requirements of ports. Whether a flow corresponds to a valid path through the interconnect fabric may be determined. A valid path starting at a source node for the flow, terminating at an end node for the flow and passing through a contiguous subset of the interconnect elements.

[0011] A flow may be assigned to a primary path in the design and then the flow may be assigned to a backup path in the design. The backup path is intended to support the flow in the event of a failure in the primary path. Thus, the interconnect fabric may be evaluated to determine whether it supports the requirements of the flows under various different failure scenarios.

[0012] Other features and advantages of the present invention will be apparent from the detailed description that follows.

BRIEF DESCRIPTION OF THE DRAWINGS

[0013] The present invention is described with respect to particular exemplary embodiments thereof and reference is accordingly made to the drawings in which:

[0014] FIG. 1 shows a method for verifying a design of an interconnect fabric according to an aspect of the present invention;

[0015] FIG. 2 shows an arrangement of flows in an exemplary interconnect fabric design;

[0016] FIG. 3 shows a design specification for an exemplary interconnect fabric design;

[0017] FIG. 4 illustrates an exemplary design for an interconnect fabric including switches and hubs;

[0018] FIG. 5 illustrates a method for identifying hub connected components in a design for an interconnect fabric according to an aspect of the present invention;

[0019] FIG. 6 illustrates an exemplary tree data structure that may be formed by the method of FIG. 5;

[0020] FIG. 7 illustrates a method for verifying a design of an interconnect fabric under various different failure condition scenarios according to an aspect of the present invention;

[0021] FIG. 8 shows an arrangement of flows for an exemplary interconnect fabric design;

[0022] FIG. 9 shows an exemplary design for an interconnect fabric including primary and backup paths for each of the flows of FIG. 8; and

[0023] FIG. 10 shows a system having a fabric design verification tool that may be used to verify a design for an interconnect fabric in accordance with an aspect of the present invention.

DETAILED DESCRIPTION OF A PREFERRED EMBODIMENT

[0024] FIG. 1 shows a method 100 for verifying a design of an interconnect fabric according to an aspect of the present invention. Requirements for the design may be referred to as flow requirements. The flow requirements may include, for example, source and terminal nodes for communication flows and required communication bandwidth for the flows. The interconnect design fabric specifies an arrangement of elements of the fabric, such as links and interconnect devices, which is intended to satisfy the flow requirements. The invention programmatically verifies the design. This may include determining whether the flow requirements are satisfied by the design and whether the design violates constraints on the elements, such as bandwidth capacity and number of available ports.

[0025] At step 102, a set of network nodes, such as source and terminal nodes, that are interconnected by the interconnect fabric design are determined. In addition, flow requirements for the fabric are determined. Table 1 shows an example set of flow requirements for an interconnect fabric design.

	Terminal Node 20	Terminal Node 22	Terminal Node 24
Source Node 10	a	b	c
Source Node 12	d	e	f
Source Node 14	—	g	h

[0026] The flow requirements in this example specify three source nodes (source nodes 10-14 in the figures below) and three terminal nodes (terminal nodes 20-24 in the figures below). For the interconnect fabric design to meet the flow requirements, it must contain communication paths between all pairs of the source nodes 10-14 and terminal nodes and 20-24 having positive flow requirements and must have sufficient bandwidth to support all of the flow requirements simultaneously.

[0027] In one embodiment, the source nodes 10-14 are host computers and terminal nodes 20-24 are storage devices and the bandwidth values for flows a-h are numbers expressed in units of megabits per second. Thus, the interconnect fabric design may be for storage area network.

[0028] In other embodiments, there may be multiple flow requirements between a given source and terminal node pair. In such embodiments, the cells of Table 1 may contain a list of two or more entries.

[0029] FIG. 2 shows an arrangement of flows in the interconnect fabric design obtained at step 102 for this example. Accordingly, a flow a forms a connection between the source node 10 and the terminal node 20, a flow b forms a connection between the source node 10 and the terminal node 22, and a flow c forms a connection between the source node 10 and the terminal node 24. Similarly, flows d, e, and f, respectively, form connections from the source node 12 to the terminal nodes 20-24 and flows g and h, respectively, form connections from the source node 14 to the terminal nodes 22-24.

[0030] Because the set of nodes and the flow requirements are the basic constraints on the design for the interconnect fabric, they may have been used as a starting point for the design which is to be verified by the present invention. Accordingly, they will generally be readily available. For example, U.S. application Ser. No. 09/707,227, filed Nov. 16, 2000, the contents of which are hereby incorporated by reference, discloses a technique for designing interconnect fabrics using a set of nodes and flow requirements as a starting point. It will be apparent, however, the present technique may be used to verify interconnect fabric designs obtained by other techniques, such as manual or other methods. Further, the set of nodes and flow requirements may be obtained in other ways. For example, the set of nodes may be obtained from the design itself. Also, the present invention may be used to verify whether an interconnect fabric initially designed to support one set of flow requirements will support a different set of flow requirements. For example, it may be desired to determine whether an existing design will meet the requirements of a new application.

[0031] In a step 104, a specification of the interconnect fabric design which is to be verified by the present invention is obtained. Typically, the design specifies a set of interconnect devices and communication links. The devices may include for example, hubs, routers, switches, and so forth. The links form physical connections among the nodes and the interconnect devices. These may include, for example, fiber optic links, fibre channel links, wire-based links, and links such as SCSI, as well as wireless links.

[0032] FIG. 3 shows a design specification for the example flow requirements. The design of FIG. 3 may be developed by the technique of U.S. application Ser. No.

09/09/707,227, mentioned above, or by another technique. As shown in **FIG. 3**, devices **30**, **32**, and **34** and a set of links **40-58** interconnect the nodes **10-14** and **20-24**. More particularly, flows **a**, **b** and **c** from the source node **10** are merged and connected to the device **30** by a link **40**. The flow **a** is connected between the device **30** and the terminal node **20** by a link **42**. The flows **b** and **c** from the device **30** are merged and connected to the device **32** by a link **44**. The flow **d** from the source node **12** is connected to the terminal node **20** by a link **46**. The flows **e** and **f** from the source node **12** are merged and connected to the device **32** by a link **48**. The flows **b** and **e** from the device **32** are merged and connected to the device **34** by a link **50**. The flows **c** and **f** from the device **32** are merged and connected to the terminal node **24** by a link **52**. The flow **g** from the source node **14** is connected to the device **34** by a link **54**. The flows **b**, **e** and **g** from the device **34** are merged and connected to the terminal node **22** by a link **56**. The flow **h** from the source node **14** is connected to the terminal node **22** by a link **58**. Rather than being represented graphically, as in **FIG. 3**, the design specification may be represented other ways. For example, the design specification may be in the form of a list including elements and connections between the elements.

[0033] In a step **106**, each flow included in the flow requirements obtained in the step **102** is associated with a path through the interconnect fabric. These associations of flows to paths may be specified by the design specification. Alternately, these associations may be developed in step **106** by comparing each flow to the design for the interconnect fabric and identifying a path through the fabric whose end points match those of the flow.

[0034] In some cases, there may be more than one possible path for the flow. In which case, the flow may be assigned to one such path and an attempt made to verify the design based on that assignment (steps **108-110**, discussed below). If the design cannot be verified, the flow may be assigned to another possible path. Flows may be assigned to new paths until the design can be verified or all the possible paths for all flows have been tried unsuccessfully.

[0035] To be a valid path for a flow, the path should start at the source node for the flow, terminate at the end node for the flow and pass through a contiguous subset of the links and devices identified in the step **104**. If a valid path cannot be identified for a flow in step **106**, this indicates that the design will not meet the flow requirements. If the design is rejected in step **106** because it does not include a valid path for each flow, it may then be modified to add one or more valid paths as needed or a new design may be selected.

[0036] In the example, each of the flows **a-h** is associated with a corresponding path through the interconnect fabric. Thus, flow **a** is associated with a path from the source node **10**, through link **40**, device **30** and link **42**, terminating at terminal node **20**. Flow **b** is associated with a path from the source node **10**, through link **40**, device **30**, link **44**, device **32**, link **50**, device **34**, and link **56**, terminating at terminal node **22**. Flow **c** is associated with a path from the source node **10**, through link **40**, device **30**, link **44**, device **32** and link **52**, terminating at terminal node **24**. Flow **d** is associated with a path from the source node **12**, through link **46** and terminating at terminal node **20**. Flow **e** is associated with a path from the source node **12**, through link **48**, device **32**, link **50**, device **34** and link **56**, terminating at terminal node

22. Flow **f** is associated with a path from the source node **12**, through link **48**, device **32** and link **52**, terminating at terminal node **24**. Flow **g** is associated with a path from the source node **14**, through link **54**, device **34** and link **34**, terminating at terminal node **22**. Flow **h** is associated with a path from the source node **14**, through link **58** and terminating at terminal node **58**.

[0037] In steps **108** and **110**, the paths identified in the step **106** are evaluated to determine whether the flow requirements for the associated flows are met by the design. More particularly, in step **108**, a path may be selected for evaluation. Elements of the selected path are then identified. These elements may include, for example, each port, interconnect device and link encountered in the path. For each such element, the requirements for the flow that corresponds to the path through that element are aggregated along with requirements for other flows through that same element. These requirements may include, for example, the bandwidth and the number of ports required for the flows. For each selected path, its flow requirements are aggregated with those of other paths that were evaluated prior to the selected path. Then, in state **110**, a determination is made as to whether the capacity of each element is exceeded by the aggregated requirements. This process is repeated for each flow and for each element of each flow.

[0038] In the example of **FIG. 3**, assume that each of interconnect devices **30-34** is a switch having a maximum bandwidth capacity of 100 Mb/s. Assume also that each of the interconnect devices **30-34** has four available ports and each port of the devices **30-34** has a maximum bandwidth capacity of 100 Mb/s. In addition, assume that each port of each of the source nodes **10-14** and each port of each of the terminal nodes **20-24** and each of the links **40-58** also has maximum bandwidth capacity of 100 Mb/s. Assume also that each of flows **a-d** and flows **f-h** require a bandwidth of 33 Mb/s and that flow **e** requires 0.5 Mb/s.

[0039] In a first pass through the step **108**, the path for flow **a** may be selected. The bandwidth requirement for the flow **a** may then be associated with each of a port at the source node **10**, the link **40**, the device **30**, the link **42** and a port at the terminal node **20**. For example, this information may be saved in computer memory. In addition, the requirement of one port at the node **10** (shared by flows **a**, **b** and **c**), two ports (an entry port and an exit port) at the device **30** and one port at the node **20** may be recorded. Then, in the step **110**, a determination may be made as to whether any of the bandwidth capacities of these elements is exceeded by the flow **a** and whether the number of available ports for each of these elements is exceeded by the flow **a**.

[0040] In a next pass through the step **108**, the path for the flow **b** may be selected. Because the flow **b** uses the same port at the source node as the flow **a**, the bandwidth requirements for both flows are aggregated. The sum of these flow requirements may then be saved in the step **108** for comparison with capacity of the port at node **20** in the step **110**. Similarly, the flow **b** also uses the link **40**, and the same entry port at the device **30** that is used by the flow **a**. Thus, the bandwidth requirements of flow **b** for each of these elements can be aggregated with those of flow **a**. However, the flow **b** uses a different exit port at the device **30**. Thus, the requirement of a third port at the device **30** may be recorded. Then, in the step **110**, requirements of the flow **b**,

aggregated with those of flow a, may be compared to the capacities of the corresponding elements of the network to determine whether any are exceeded.

[0041] While not used by the flow a, the link 44, the device 32, the link 50, the device 34, the link 58 and a port of the terminal node 22 are used by the flow b. Thus, in step 110, the requirements for the flow b at each of these elements may be compared the capacities of the corresponding element to determine whether any are exceeded.

[0042] In this example, none of the capacities are exceeded by the flows a and b. For example, the device 30 has maximum input bandwidth capacity of 100 Mb/s, however, the total used by flows a and b is 66 Mb/s, which is less than the maximum. As another example, the device 30 has four ports, however, the flows a and b only require three ports at the device 30.

[0043] In another pass through the step 108, the path for the flow c may be selected and its requirements aggregated with those of flows a and b. Thus, the requirements for the flow c may be aggregated with those of the other flows for each of the source node 10, the link 40, the device 30, the link 44, the device 32, the link 52 and the terminal node 24. Then, in step 110, the aggregated requirements for the flows a, b and c may be compared to the capacities of the corresponding elements of the network to determine whether any are exceeded.

[0044] The steps 108 and 110 may be repeated for each of the flows. In this manner, the additional requirements of each flow may be aggregated with the flows considered in previous passes through the step 108. In a final pass through the step 110, the aggregated requirements for all of the flows to be supported by the design may be compared the capacities of the corresponding elements of the network to determine whether any are exceeded.

[0045] In the example, none of the capacities of elements of the network are exceeded by the requirements of the flows a-h. For example, the aggregated bandwidth requirement for the device 32 is 99.5 Mb/s. This includes 33 Mb/s for the flow b, 33 Mb/s for the flow c, 0.5 Mb/s for the flow e and 33 Mb/s for the flow f, resulting in a sum of 99.5 Mb/s. In addition, these flows require four ports at the device 32, two for entering flows and two for exiting flows. The maximum bandwidth capacity for the device 32 is 100 Mb/s and it has four ports. Accordingly, neither the bandwidth capacity, nor port number capacity of the device 32 is exceeded. Thus, the method 100 may terminate with a positive result after the final pass through the step 110.

[0046] In another example, assume that the flow e requires 10 Mb/s of bandwidth, rather than the 0.5 Mb/s previously assumed. In this case, the aggregated bandwidth requirements for the device 32 includes 33 Mb/s for the flow b, 33 Mb/s for the flow c, 10 Mb/s for the flow e and 33 Mb/s for the flow f, resulting in a sum of 109 Mb/s. This exceeds the maximum bandwidth available for the device 32, which is 100 Mb/s. Accordingly, a determination in the state 110 may be that the bandwidth capacity of the device 32 is exceeded. In response, the design for the interconnect fabric may be modified in order to increase its bandwidth capacity or the flow requirements relaxed in order to reduce the bandwidth requirements.

[0047] In the example above, the interconnect devices 30-34 are switches. Accordingly, communications for a flow

that passes through one of these devices are passed from an entry port of the device to a specified exit port of the device. The bandwidth requirements for the flow may be aggregated in the step 108 along with other flows at the same input and exit ports to determine whether the maximum bandwidth capacity of either the input or exit port is exceeded. In addition, bandwidth requirements for the flow may be aggregated with all flows that enter the interconnect device to determine whether the maximum bandwidth capacity of the device is exceeded.

[0048] For other devices, such as hubs or repeaters, communications for a flow that enters a port of the device may be repeated at all other ports of the device, not just a specified exit port as in the case of switches. As a result, bandwidth consumed at one port to receive communications is also consumed at each other port in order to retransmit the communications. Accordingly, in step 108, the bandwidth requirement for a flow entering such a device is aggregated along with the bandwidth requirements for all the other flows entering the device to determine whether the bandwidth capacity of any port is exceeded. This means that the lowest bandwidth capacity among the ports of the device determines the maximum bandwidth of the device itself.

[0049] When two or more such devices are connected together, a communication received at any port of a connected device may be repeated at each other port of that device and at each port of the other connected devices. Thus, bandwidth consumed at any port of a connected group of hub or repeater devices may also be consumed at each other port of the connected group of devices in order to repeat the communication.

[0050] Elements other than a hub or a repeater, such as a switch, a source node or a terminal node, may be connected to a port of a hub or a repeater. In which case, communications between those elements and the hub or repeater may be repeated at each other port of the hub or repeater and at each port of any other connected hub or repeater. Thus, bandwidth consumed at any port connected to hub or repeater may also be consumed at each other port of the connected group of devices in order to repeat the communication. As used herein, the term "hub connected" refers to any hub or other network element connected to a hub for which communications repeated by the hub consume bandwidth. This includes links directly connected to a port of a hub or repeater and interconnect devices directly connected to such links. Hub connected components can be said to be within the same "domain" where communications are repeated to each such hub connected component. When a path identified in step 106 utilizes any hub connected component, all of the devices in the same domain may also be impacted by the flow. Thus, in the step 108, bandwidth requirements for a flow at a hub connected component may be aggregated with the bandwidth requirements for flows at other hub connected components in the same domain in order to determine in step 110 whether the bandwidth capacity of any such component is exceeded.

[0051] FIG. 4 illustrates an exemplary design for an interconnect fabric including switches 130-138 and hubs 140-144. More particularly, a source node 110 is connected to the switch 130 by a link 150. The switch 130 is connected to a terminal node 120 by a link 152 and to a terminal node 122 by a link 154. The switch 132 is connected to the source

node **130** by a link **156** and to a source node **112** by a link **158**. The hub **140** is connected to the switch **132** by a link **160**, to the switch **134** by a link **162** and to the hub **142** by a link **164**. The switch **134** is also connected to the terminal node **120** by a link **166** and to the terminal node **122** by a link **168**.

[0052] The switch **136** is connected to the source node **112** by a link **170**, to the source node **114** by a link **172** and to the hub **144** by a link **174**. The hub **144** is also connected to the source node **110** by a link **176** and to the hub **142** by a link **178**. The switch **138** is connected to the source node **114** by a link **180**, to the hub **142** by a link **182** and to the terminal node **124** by a link **184**.

[0053] Assume that a path for a flow passes from the source node **110** to the terminal node **120** via the link **156**, the switch **132**, the link **160**, the hub **146**, the link **162**, the switch **134** and the link **166**. Returning to the step **106**, assume that a path through the interconnect fabric is being associated with the flow. Because, the switch **132** is connected to the hub **140** by the link **160**, the path includes at least one hub connected component (other hub connected components in the path include the switch **134** and the link **162**). Thus, in order to aggregate the bandwidth requirements for the appropriate other flows in step **108**, all of the other hub connected components in the same domain as the hub **140** may be identified. Once such hub connected components are identified, their associated flows can be identified so that their bandwidth requirements may be aggregated appropriately when the step **108** is performed for each such flow.

[0054] FIG. 5 illustrates a method **200** for identifying hub connected components in a design for an interconnect fabric in accordance with an aspect of the present invention. In step **202**, a hub connected component is encountered in a path between a source node and a terminal node. This may occur, for example, during the step **106** of the method **100** (FIG. 3). In the example of FIG. 4, the first hub connected component encountered may be switch **132**.

[0055] In a step **204**, the hub connected component is added to a tree data structure. FIG. 6 illustrates an exemplary tree data structure **250** that may be formed by the method **200** of FIG. 5. The data structure **250** may be stored, for example, in computer memory. As shown in FIG. 6, the switch **132** may be initially added to the data structure **250**.

[0056] In a step **206**, the connections of the hub connected component added to the tree **250** are searched for other hub connected components in the same domain. In the example, the switch **132** is connected to the hub **140**. Accordingly, the hub **140** may be identified in the step **204** as another hub connected component in the same domain. From the step **204**, the step **202** is repeated during which the newly identified component may be added to the tree data structure **250**. Thus, as shown in FIG. 6, the hub **140** may be added to the tree **250**. Hub connected links may also be identified in the tree **250** as shown in FIG. 6 by the link **160**.

[0057] The tree **250** may be formed with hub connected components other than hubs being one level lower than a directly-connected hub. Accordingly, in the example, the hub **140** is inserted into the tree **250** one level higher than the switch **132**. Once a hub is added to the tree, the following passes through the steps **204** and **206** search the interconnect

fabric design to determine whether there are any additional components connected to the hub. If any such devices are found, they are added to the tree, branching from that hub and one level down. If any of those devices are hubs, the interconnect fabric design is searched again for each such hub to determine whether there are any additional components connected to each of those hubs. Then, any such devices found are added to the tree branching from the connected hub. This process may be referred to as a depth-first search which continues until all the hub connected components in the same domain are identified and added to the tree.

[0058] In the example, after the hub **140** is added to the tree **250**, the switch **134** may be added one level down from the hub **140** in a next pass through the steps **204** and **206**. This is shown in FIG. 6. Then, the hub **142** may be added one level down from the hub **140**. Since there is no other components connected to the hub **140**, the search then moves down to the hub **142**. In the next passes through the steps **204** and **206**, the switch **136** and hub **144** are added to the tree **250** since they are connected to the hub **140**. A search for components connected to the hub **144** results in locating the switch **136** and the node **110** since they are connected to the hub **144**. Because all of the elements in the level below the hub **144** are not hubs, this means that search of the domain has been exhausted.

[0059] Thus, according to one aspect of the invention, when a hub connected component is encountered, the interconnect fabric design may be searched to determine the extent of the domain. This may be accomplished by performing a depth-first search, as explained above or by another technique, such as a union-find algorithm. The results identify all of the hub connected components in the domain for which the bandwidth requirements may be aggregated in the step **108** of FIG. 1 in order to determine whether the bandwidth capacity of any such component is exceeded.

[0060] In another aspect of the invention, the method **100** of verifying a design for an interconnect fabric may be used to evaluate the ability of the design to withstand various different failure conditions. FIG. 7 illustrates a method **100** for verifying a design of an interconnect fabric under various different failure condition scenarios according to an aspect of the present invention. In a step **302**, a failure scenario may be set up in a design for an interconnect fabric. Then, in a step **304**, the design may be evaluated to determine whether it satisfies the flow requirements under the failure scenario set up in step **302**. This process may be repeated for any number of different scenarios.

[0061] In an example, assume that a design for an interconnect fabric includes primary and backup paths for each flow, where the backup path is intended to support the flow in the event of a failure in the primary path. Thus, in a first failure condition scenario set up in step **302**, the flows may be assigned to the primary path for each flow. Then, in step **304**, the method **100** may be used to evaluate whether the flow requirements are met by the interconnect fabric under conditions of the first scenario. In a second failure condition scenario set up in a next pass through the step **302**, a network element in the primary path may be assumed to have failed. In which case, one or more of the flows may be assigned to backup paths. Then, in step **304**, the method **100** may be

used to evaluate whether the flow requirements are met by the interconnect fabric under conditions of the second scenario.

[0062] FIG. 8 shows an arrangement of flows for an interconnect fabric for this example. A flow i forms a connection between a source node 40 and a terminal node 50, a flow j forms a connection between the source node 40 and a terminal node 52, a flow k forms a connection between the source node 40 and a terminal node 54. Flows l, m, and n, respectively, form connections from a source node 42 to the terminal nodes 50-54.

[0063] FIG. 9 shows an exemplary design for an interconnect fabric including primary and backup paths for each of the flows of FIG. 8. More particularly, primary paths for the flows i, j and k pass through a link 326 that connects the source node 310 to a device 328. From the device 328, the primary path for i is through a link 330 that connects the device 328 to the terminal node 320, the primary path for j is through a link 332 that connects the device 328 to the terminal node 322 and the primary path for k passes through a link 334 that connects the device 328 to the terminal node 324. Primary paths for the flows l, m and n pass through a link 336 that connects the source node 312 to a device 338. From the device 338, the primary path for l is through a link 340 that connects the device 338 to the terminal node 320, the primary path for m is through a link 342 that connects the device 338 to the terminal node 322 and the primary path for n passes through a link 344 that connects the device 338 to the terminal node 324.

[0064] Backup paths for the flows i, j and k pass through a link 346 that connects the source node 310 to a device 348. Backup paths for the flows l, m and n pass through a link 350 to the device 348. From the device 348, the backup paths for i and l are through a link 352 that connects the device 348 to the terminal node 320, the backup paths for j and m are through a link 354 that connects the device 348 to the terminal node 322 and the backup paths for k and n are through a link 356 that connects the device 348 to the terminal node 324.

[0065] Because all of the backup paths of FIG. 9 utilize the device 348, the device may not have sufficient bandwidth capacity to support all of the flows simultaneously. However, the device 348 preferably has sufficient bandwidth capacity to handle the flows i, j and k simultaneously (as may be needed if the device 328 should fail) or the flows l, m and n simultaneously (as may be needed if the device 338 should fail). Accordingly, the interconnect fabric of FIG. 9, including the primary and backup paths, may be developed by a method disclosed in co-pending U.S. application Ser. No. _____, filed, Jan. 17, 2002, entitled, "Reliability for Interconnect Fabrics," the contents of which are hereby incorporated by reference or by another method.

[0066] In the example of FIG. 9, in a first pass through the step 302 of the method 300 of FIG. 7, the flows i-n may all be assigned to their primary paths. In this scenario it may be assumed that neither of the devices 328 and 338 has failed. Thus, in a first pass through the step 304, the method 100 of FIG. 1 may be used to determine whether primary paths of the design of FIG. 9 satisfy the flow requirements for all of the flows. In a second pass through the step 302, the flows i, j and k may be assigned to their backup paths through the device 348. Thus, in this scenario, it may be assumed that the

device 328 has failed. Then in a second pass through step 304, the method 100 may be used to determine whether this failure scenario is supported by the design of FIG. 9. In a third pass through the step 302, the flows l, m and n may be assigned to their backup paths through the device 348. Thus, in this scenario, it may be assumed that the device 328 has failed. Then in a third pass through step 304, the method 100 may be used to determine whether this failure scenario is supported by the design of FIG. 9.

[0067] In other embodiments, the interconnect fabric may include redundant paths sufficient to support all of the flows simultaneously. In which case, such an interconnect fabric may be developed by a method disclosed in co-pending U.S. application Ser. No. _____, filed Dec. 19, 2001, entitled, "Reliability for Interconnect Fabrics" or by another method. Thus, in a scenario set up in step 302, one or more flows may be assigned to both a primary path for the flow and a redundant path for the flow. In step 304, the method 100 may be used to determine whether the design is capable of supporting such flows simultaneously.

[0068] Accordingly, the method 300 of FIG. 7 may be used to determine whether a design for an interconnect fabric supports the flow requirements under various different failure scenarios.

[0069] FIG. 10 shows a system having a fabric design verification tool 400 that may employ the method 100 (and the methods 200 and 300) to verify a design for an interconnect fabric. The fabric design verification tool 400 may be implemented in computer software and/or hardware to perform its functions. Design information 422 in one embodiment includes a list of hosts (source nodes) and devices (terminal nodes) 410, an interconnect design specification 412, a set of flow requirements data 414, a set of port availability data 416 and a set of bandwidth data 418. The design information 422 may be implemented as an information store, such as a file or set of files or a database, etc.

[0070] The list of hosts and devices 410 may specify the hosts and devices which are to be interconnected by an interconnect fabric design 412. This list 410 may be obtained in step 102 of FIG. 1.

[0071] The interconnect fabric design specification 412 may specify the interconnect fabric design to be verified. The design specification 412 may be obtained in the step 104 of FIG. 1.

[0072] The flow requirements data 414 may specify the desired flow requirements for the interconnect fabric design 412. The desired flow requirements may include bandwidth requirements for each pairing of the source and terminal nodes and may be obtained in the step 102 of FIG. 1.

[0073] The port availability data 416 may specify the number of communication ports available on each source node and each terminal node and each available interconnect device.

[0074] The bandwidth data 418 may specify the bandwidth of each host and device port and each type of fabric node and link. The bandwidth data may also specify maximum bandwidth for entire interconnect devices.

[0075] Verification result 420 generated by the fabric design verification tool 400 may include an indication as to whether or not the design 412 satisfies the flow requirements 414.

[0076] The foregoing detailed description of the present invention is provided for the purposes of illustration and is not intended to be exhaustive or to limit the invention to the precise embodiment disclosed. Accordingly, the scope of the present invention is defined by the appended claims.

What is claimed is:

1. A computer implemented method for verifying a design for an interconnect fabric, the design including an arrangement of interconnect elements for interconnecting a plurality of network nodes and the design having requirements for a plurality of flows among the network nodes, and for each of the plurality of flows, the method comprising associating the flow with a path for the flow through the interconnect fabric, and for each interconnect element in each path, aggregating requirements associated with each of the corresponding flows and determining whether the aggregated requirements exceeds a capacity of the interconnect element.

2. The method according to claim 1, wherein the interconnect elements include interconnect devices and links.

3. The method according to claim 2, wherein the interconnect devices are selected from the group consisting of switches and hubs.

4. The method according to claim 3, wherein when the interconnect devices includes a hub, the method further comprises identifying an extent of a domain of hub connected components.

5. The method according to claim 4, wherein said identifying the extent of the domain of hub connected components comprises performing a depth first search of the interconnect fabric for the hub connected components.

6. The method according to claim 5, wherein said identifying an extent of a domain of hub connected components comprises constructing a tree data structure wherein a hub occupies a position in the tree and a other interconnect elements connected to the hub occupy positions in the tree one level down from the hub.

7. The method according to claim 1, wherein the aggregated requirements include bandwidth requirements.

8. The method according to claim 7, further comprising aggregating requirements of ports for each of the plurality of flows and determining whether a number of available ports of one or more of the interconnect elements is exceeded by the aggregated requirements of ports.

9. The method according to claim 1, wherein the aggregated requirements include a number of ports.

10. The method according to claim 1, said method further comprising determining whether a flow corresponds to a valid path through the interconnect fabric, a valid path starting at a source node for the flow, terminating at an end node for the flow and passing through a contiguous subset of the interconnect elements.

11. The method according to claim 10, further comprising rejecting the design if it does not include a valid path for each flow.

12. The method according to claim 1, wherein said associating comprises assigning a flow to a primary path in the design and further comprising assigning the flow to a backup path in the design to determine whether the design has capacity for the flow in the primary path and the backup path simultaneously.

13. The method according to claim 1, wherein said associating comprises assigning a flow to a backup path for the flow in the design to determine whether the design has

capacity for the flow in the secondary path in event of a failure in a primary path for the flow.

14. A system for verifying a design for an interconnect fabric comprising:

a set of design information including requirements for a plurality of flows and a design specification wherein each of the plurality of flows is associated with a path for the flow through the interconnect fabric; and

a fabric design verification tool that, for each interconnect element in each path, aggregates requirements associated with each of the corresponding flows and determines whether the aggregated requirements exceeds a capacity of the interconnect element.

15. The system according to claim 14, wherein the interconnect elements include interconnect devices and links.

16. The system according to claim 15, wherein the interconnect devices are selected from the group consisting of switches and hubs.

17. The system according to claim 16, wherein when the interconnect devices includes a hub, the design verification tool identifies an extent of a domain of hub connected components.

18. The system according to claim 17, wherein the design verification tool identifies the extent of the domain of hub connected components by performing a depth first search of the interconnect fabric for the hub connected components.

19. The system according to claim 18, wherein the design verification tool identifies an extent of a domain of hub connected components by constructing a tree data structure wherein a hub occupies a position in the tree and a other interconnect elements connected to the hub occupy positions in the tree one level down from the hub.

20. The method according to claim 14, wherein the aggregated requirements include bandwidth requirements.

21. The system according to claim 20, wherein the design verification tool aggregates requirements of ports for each of the plurality of flows and determines whether a number of available ports of one or more of the interconnect elements is exceeded by the aggregated requirements of ports.

22. The method according to claim 14, wherein the aggregated requirements include a number of ports.

23. The system according to claim 14, wherein the design verification tool determines whether a flow corresponds to a valid path through the interconnect fabric, a valid path starting at a source node for the flow, terminating at an end node for the flow and passing through a contiguous subset of the interconnect elements.

24. The system according to claim 23, wherein the design verification tool rejects the design if it does not include a valid path for each flow.

25. The system according to claim 14, wherein the design verification tool assigns a flow to a primary path in the design and also assigns the flow to a backup path in the design to determine whether the design has capacity for the flow in the primary path and the backup path simultaneously.

26. The system according to claim 14, wherein the design verification tool assigns a flow to a backup path for the flow in the design to determine whether the design has capacity for the flow in the backup path in event of a failure in a primary path for the flow.

* * * * *