

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
3 July 2008 (03.07.2008)

PCT

(10) International Publication Number
WO 2008/079667 A2

(51) International Patent Classification:
H04L 12/28 (2006.01)

(21) International Application Number:
PCT/US2007/087102

(22) International Filing Date:
12 December 2007 (12.12.2007)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
11/612,906 19 December 2006 (19.12.2006) US

(71) Applicant (for all designated States except US): **CISCO TECHNOLOGY, INC.** [US/US]; 170 West Tasman Drive, SJC/10/2/1, San Jose, California 95134-1706 (US).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **BULUSU, Shekher** [IN/US]; 39493 Gallaudet Drive, Apt. #141, Fremont, California 94538 (US). **DESAI, Ronak** [IN/US]; 41176 St. Anthony Drive, Fremont, California 94539 (US).

(74) Agents: **LAFFERTY, Wm. Brook** et al.; SCIENTIFIC-ATLANTA, INC., Intellectual Property Dept., 5030 Sugarloaf Parkway, Lawrenceville, Georgia 30044 (US).

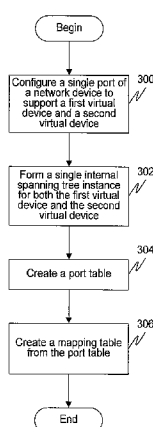
(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, MT, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— without international search report and to be republished upon receipt of that report

(54) Title: SHARED VIRTUAL DEVICE PORTS



(57) Abstract: In one embodiment, a solution is provided wherein multiple virtual devices may be configured on the same physical port of a network device. For example, a first virtual device and a second virtual device may be configured to use the same physical port. A single internal spanning tree instance may be configured for both the first virtual device and the second virtual device.

WO 2008/079667 A2

SHARED VIRTUAL DEVICE PORTS

BACKGROUND

1. Technical Field

[0001] The present disclosure relates to virtual devices in computer networks.

2. Description of the Related Art

[0002] A spanning tree is a data structure used to map the paths to the devices in a network. The spanning tree is typically used as a way to avoid loops in the network that could lead to crashes or infinite delays during routing of traffic. Multiple Spanning Tree Protocol (MSTP), as defined in IEEE 802.1q, is a standard wherein multiple spanning trees are utilized in a computer network. MSTP allows Virtual Local Area Networks (VLANs) to be mapped to a different spanning tree instances within the same network, such that traffic corresponding to a first VLAN may be routed according to a first spanning tree, while traffic corresponding to a second VLAN may be routed according to a second spanning tree.

[0003] Devices running the MSTP have one or more ports. In addition to mapping each VLAN to a spanning tree, each VLAN can also be matched to one or more of the ports.

BRIEF DESCRIPTION OF THE DRAWINGS

[0004] FIG. 1 depicts an example network that includes layer 2 Ethernet bridges conforming to IEEE Std.[™] 802.1D-2004 (IEEE 802.1D) or IEEE Std. 802.1Q-2003 (IEEE 802.1Q).

[0005] FIG. 2 illustrates an example of a port table in accordance with an embodiment of the present invention.

[0006] FIG. 3 illustrates an example method for configuring multiple virtual devices to share a single physical port of a network device in accordance with an embodiment of the present invention.

[0007] FIG. 4 illustrates an example method for handling a packet in a network device in accordance with an embodiment of the present invention.

[0008] FIG. 5 illustrates an example apparatus for configuring multiple virtual devices to share a single physical port of a network device in accordance with an embodiment of the present invention.

[0009] FIG. 6 illustrates an example apparatus for handling a packet in a network device in accordance with an embodiment of the present invention.

[0010] FIG. 7 illustrates a simplified architecture of a switch.

DESCRIPTION OF EXAMPLE EMBODIMENTS

Overview

[0011] In one embodiment, a solution is provided wherein multiple virtual devices may be configured on the same physical port of a network device. For example, a first virtual device and a second virtual device may be configured to use the same physical port. A single internal spanning tree instance may be configured for both the first virtual device and the second virtual device.

Examples

[0012] In this application, numerous specific details are set forth in order to provide a thorough understanding of the present invention. It will be obvious, however, to one skilled in the art, that the present invention may be practiced without some or all of these specific details. In other instances, well known process steps have not been described in detail in order to not obscure the present invention.

[0013] FIG. 1 depicts simple network 100 that includes layer 2 Ethernet bridges conforming to IEEE Std.[™] 802.1D-2004 (IEEE 802.1D) or IEEE Std. 802.1Q-2003 (IEEE 802.1Q) 105. In this example, there are 9 bridges A-I and three stations, 140-142, connected via 12 local area networks (LANs) 125, 130, and 150 (10 instances). The letters A-I are Bridge IDs. Lower letters (e.g., A) are “better” than higher letters (e.g., D), according to the convention of IEEE 802.1D or IEEE 802.1Q, which are hereby incorporated by reference for all purposes. Port path costs 110 are indicated for each LAN. This diagram assumes that all bridge ports attached to the same LAN have the same value configured for their port path cost, though this is not required either by IEEE 802.1D or by the present invention.

[0014] Bridges create a spanning tree over network 100 by exchanging protocol packets called Bridge Protocol Data Units (BPDUs). Using these packets, the protocol state machines implemented in each bridge select certain ports to be part of the active topology of the network and certain others to be blocked. The spanning tree is “spanning” in the sense that all LANs are connected. It is a “tree” in that there is exactly one path between any given pair of bridges or stations.

[0015] According to Spanning Tree Protocol (STP), one of the bridges of network 100 (in this example, bridge A) will be elected as the “Root Bridge”. The tree is constructed by each bridge selecting the port that is closest to the Root Bridge as its “Root Port,” where “closest” is defined as the path to the root bridge with the least numerical sum of root path costs for the bridge ports traversed. (Only ingress from a LAN to a bridge counts in this summation, not egress from a bridge to a LAN.) Where two or more paths have the same sum of port path costs, tiebreaker values are used. The tiebreaker values are Bridge IDs and Port IDs, as defined in IEEE 802.1D and IEEE 802.1Q. Bridge E selects from among three equal cost paths, thus breaking three potential spanning tree loops, by using the bridge IDs of the adjacent bridges to select port 117 as its root port, and to mark as “alternate ports” and block ports 115 and 120. (Although these terms can have different meanings when used by those of skill in the art, the terms “packet” and “frame” will sometimes be used interchangeably herein).

[0016] Although STP provides for the orderly flow of packets, it does not allow for all LANs in a network to be used in an optimal fashion. There is never a path from any bridge or station in the network to the root that has a lower sum of path costs than the path along that root bridge’s spanning tree, but paths between other pairs of bridges or stations may not be optimal. For example, if ports 115 and 120 are blocked, the path from station 142 to station 141 is 142-E-F-I-A-B-D-G-141, for a total cost of 11 (station 141 does not count the cost of the last hop), instead of the much shorter path 142-E-G-141, cost 3.

[0017] Multiple Spanning Tree Protocol (MSTP) offers an improvement over STP in that multiple spanning trees are created. Network 100 could be configured to have up to 64 spanning trees. For example, each bridge A through I could be configured to be the Regional Root Bridge of a separate multiple spanning tree instance (MSTI). IEEE 802.1Q bases the determination of the MSTI to which a given frame is assigned, and over which that frame is forwarded, solely upon the frame’s Virtual Local Area Network Identifier (VLAN ID or VID), which in turn, is based upon only a) which port the frame entered the network, and/or b) what higher-layer protocol the frame carries. VIDs can often be assigned to ports and/or protocols that result in the selection of suitable MSTIs. However, this is not always possible, since any given station connected to the network may converse with a number of other stations, and in turn, they with others. For example, in network 100, each station 140, 141, and 142 may converse with each other.

[0018] In one example of the present invention, it is recognized that it may be desirable to share ports between two or more virtual devices using MSTP or a similar port-centric protocol. Each virtual device may be viewed as, for example, a different layer 2 switches with a set of ports assigned to it. A shared port may carry traffic belonging to network classifications (e.g., VLANs belonging to different virtual devices that are mapped on that port). Network identifiers (e.g., VLAN identifiers) may be used to define these classifications. The shared port presents a problem, however, when port-centric protocols such as MSTP, are utilized. This is due to the fact that there is only one MSTP internal spanning tree controlling and running the multiple spanning trees on the shared port, but since the port is shared between multiple virtual devices and each virtual device has its own internal spanning tree, there is the potential for conflict or ambiguity when deciding which tree to utilize.

[0019] A typical device running MSTP will have up to 65 different spanning tree instances. The first, called the Internal Spanning Tree (IST) instance, runs on all ports of the physical switch. Then there are up to 64 different Multiple Spanning Tree (MST) instances. The IST instance's job is to receive and send Bridge Protocol Data Units (BPDUs) from and to the MST instances. The BPDUs are the data structure that may contain spanning tree information. Each of the 64 different MST instances may be mapped to a different VLAN. Traffic corresponding with that VLAN would then use the corresponding spanning tree.

[0020] When multiple virtual devices are defined on a single physical switch, each of the multiple devices has its own IST instance and its own MST instances. As described briefly above, problems may be encountered when one attempts to assign multiple virtual devices to a single port, as it is then not clear which IST instance (and possibly MST instance) to utilize.

[0021] One solution would be to simply restrict the virtual devices so that ports are never shared between virtual devices. For obvious reasons, however, this is not an ideal solution, as there is a strong need to share ports between multiple virtual devices so that port bandwidth is efficiently utilized.

[0022] In an embodiment of the present invention, virtual devices that share a port may be grouped together as a virtual device group. Then all the IST instances for each virtual device in the virtual device group may be merged into a single IST instance for the entire virtual device group. The virtual devices may maintain their current MST instances. The single IST

then may send out BPDUs to all the MST instances in the virtual device group. It should be noted that implementations are possible wherein instead of merging two IST instances, a single joint IST instance is either created from scratch or created using two or more existing IST instances as templates.

[0023] It should be noted that MSTs are typically identified by the order in which they are assigned to a VLAN within a virtual device. For example, a particular device may have MSTs identified as MST₁-MST₆₄, and mapped to VLAN₁-VLAN₆₄.. This can create a problem, however, when a single port is shared between multiple virtual devices, as a first virtual device may have, for example, an MST₁ instance and a second virtual device may also have an MST₁ instance, despite the fact that the multiple spanning trees themselves are different (mapped to different VLANs). In such circumstances, a mapping table that maps the virtual device and original MST instance to a new MST mapped identification (MSTID) may be utilized. This allows, for example, the first MST₁ instance to be assigned MSTID of MST₁, while the second MST₁ instance may be assigned an MSTID of MST₂, removing the conflict.

[0024] A shared port can also carry traffic corresponding to multiple network classifications (e.g., VLANs). Ordinarily, a VLAN-to-MST instance mapping would indicate which MST instance to use for a particular VLAN. With a shared port, however, it is possible that the same VLAN may be identified on multiple virtual devices sharing the same port, and those same VLANs may identify different corresponding MST instances. It is therefore necessary to select one single MST instance to correspond to a VLAN that is defined by two or more virtual devices. This may include prompting the user for a selection or randomly selecting one, for example. The MST instance corresponding to each VLAN may also be stored in the mapping table.

[0025] FIG. 2 illustrates an example of a port table in accordance with an embodiment of the present invention. In this example, one column 200 of the table represents the virtual device identification, another column 202 of the table represents a VLAN identifier, another column 204 of the table represents the MST instance identifier, and another column 206 of the table represents a MST mapped identification.

[0026] Regions are typically identified by a signature comprising the name and revision identification of the region and also the instance-VLAN mapping for the port. Thus, a shared

port will modify this region signature since a new instance-VLAN mapping (as identified by the mapping table).

[0027] FIG. 3 illustrates an example method for configuring multiple virtual devices to share a single physical port of a network device in accordance with an embodiment of the present invention. The network device may be a switch. At 300, a single port of a network device may be configured to support a first virtual device and a second virtual device. At 302, a single internal spanning tree instance may be formed for both the first virtual device and the second virtual device. In one embodiment, this may involve merging an internal spanning tree corresponding to the first virtual device with an internal spanning tree corresponding to the second virtual device. In another embodiment, this may involve creating a single internal spanning tree instance for both the first and the second virtual devices from scratch (i.e., without reference to an internal spanning tree instance defined for the first virtual device or the second virtual device). At 304, a port table may be created. The port table may indicate, for the port, which network identifications (e.g., VLAN identifications) are assigned to which virtual device. Each network topology or customer may be assigned to only one virtual device. This table may be ascertained from a VLAN to instance mapping for the first virtual device and a VLAN to instance mapping for the second virtual device. For VLANs that could potentially map to two (or more) virtual devices, a choice may then be made by a user or administrator as to which virtual device to assign to the VLAN. Alternatively, some other method may be used to select which virtual device to assign to the VLAN (such as randomly). It should be noted that the port table itself is optional, as the same information may be stored in the mapping table. At 306, a mapping table may be created from the port table. The mapping table may include an identification of each virtual device, the VLANs assigned to the virtual device, corresponding multiple spanning tree instances for the VLANs, and mapped multiple spanning tree instances for the VLANs. The mapped multiple spanning tree instances may represent spanning tree instance identifications that are unique within the mapping table. The creating may include identifying a first multiple spanning tree instance for the first virtual device having a first label identical to a second label of a second multiple spanning tree instance for the second device. The creating may then also include adding, to the mapping table, multiple spanning tree mapping identifications that are unique within the mapping table for each of the first and second multiple spanning tree instances. The creating may also include combining network identifications defined for the first virtual device with network identifications defined for the second virtual device. For network

identifications defined for both the first virtual device and the second virtual device, one multiple spanning tree instance may be selected to correspond to the network identification. This may include prompting the user for a selection.

[0028] FIG. 4 illustrates an example method for handling a packet in a network device in accordance with an embodiment of the present invention. At 400, the packet may be received on a port. At 402, the packet may be classified into a VLAN. This may be accomplished by, for example, examining a tag in the packet itself. At 404, a first virtual device corresponding to the VLAN may be ascertained, wherein the first virtual device shares the port with a second virtual device. This may be accomplished by, for example, checking a mapping or port table using the VLAN identified in 402 and retrieving the corresponding virtual device from that table. At 406, a mapping table may be checked to determine a multiple spanning tree instance to use for routing the packet. The mapping table may contain unique identifications for each of a plurality of multiple spanning tree instances corresponding to the port. The spanning tree instance chosen may be the one corresponding to the VLAN and virtual device corresponding to the packet identified earlier.

[0029] FIG. 5 illustrates an example apparatus for configuring multiple virtual devices to share a single physical port of a network device in accordance with an embodiment of the present invention. The network device may be a switch. A multiple virtual device single port configurator 500 may configure a single port of a network device to support a first virtual device and a second virtual device. A single internal spanning tree instance former 502 coupled to the multiple virtual device single port configurator 500 may form a single internal spanning tree instance for both the first virtual device and the second virtual device. In one embodiment, this may involve merging an internal spanning tree corresponding to the first virtual device with an internal spanning tree corresponding to the second virtual device. In another embodiment, this may involve creating a single internal spanning tree instance for both the first and the second virtual devices from scratch (i.e., without reference to an internal spanning tree instance defined for the first virtual device or the second virtual device). A port table creator 504 coupled to the single internal spanning tree instance former 502 may create a port table. The port table may indicate, for the port, which network identifications (e.g., VLAN identifications) are assigned to which virtual device. Each network topology or customer may be assigned to only one virtual device. This table may be ascertained from a VLAN to instance mapping for the first virtual device and a VLAN to

instance mapping for the second virtual device. For VLANs that could potentially map to two (or more) virtual devices, a choice may then be made by a user or administrator as to which virtual device to assign to the VLAN. Alternatively, some other method may be used to select which virtual device to assign to the VLAN (such as randomly). It should be noted that the port table itself is optional, as the same information may be stored in the mapping table. A mapping table creator 506 coupled to the port table creator 504 may create a mapping table from the port table. The mapping table may include an identification of each virtual device, the VLANs assigned to the virtual device, corresponding multiple spanning tree instances for the VLANs, and mapped multiple spanning tree instances for the VLANs. The mapped multiple spanning tree instances may represent spanning tree instance identifications that are unique within the mapping table. The creating may include identifying a first multiple spanning tree instance for the first virtual device having a first label identical to a second label of a second multiple spanning tree instance for the second device. The creating may then also include adding, to the mapping table, multiple spanning tree mapping identifications that are unique within the mapping table for each of the first and second multiple spanning tree instances. The creating may also include combining network identifications defined for the first virtual device with network identifications defined for the second virtual device. For network identifications defined for both the first virtual device and the second virtual device, one multiple spanning tree instance may be selected to correspond to the network identification. This may include prompting the user for a selection.

[0030] FIG. 6 illustrates an example apparatus for handling a packet in a network device in accordance with an embodiment of the present invention. A packet receiver 600 may receive the packet on a port. A packet VLAN classifier 602 coupled to the packet receiver 600 may classify the packet into a VLAN. This may be accomplished by, for example, examining a tag in the packet itself. A VLAN virtual device ascertainment 604 coupled to the packet VLAN classifier 602 may ascertain a first virtual device corresponding to the VLAN, wherein the first virtual device shares the port with a second virtual device. This may be accomplished by, for example, checking a mapping or port table using the VLAN identified earlier and retrieving the corresponding virtual device from that table. A mapping table multiple spanning tree instance determiner 606 coupled to the VLAN virtual device ascertainment 604 may check a mapping table to determine a multiple spanning tree instance to use to route the packet. The mapping table may contain unique identifications for each of a plurality of

multiple spanning tree instances corresponding to the port. The spanning tree chosen may be the one corresponding to the VLAN and virtual device corresponding to the packet identified earlier.

[0031] FIG. 7 illustrates a simplified architecture of a switch 700. Switch 700 includes *N* line cards, each of which characterized by an ingress side (or input) 705 and an egress side (or output) 725. Line card ingress sides 765 are connected via switching fabric 750, which includes a crossbar in this example, to line card egress sides 725. In this embodiment, one or more line cards performs one or more of the processes described above.

[0032] It should be noted that even though throughout this document reference is made to a first virtual device and a second virtual device, the claimed invention may include embodiments where more than two virtual devices share the same physical port.

[0033] It should also be noted that, while MSTP is discussed, the claimed invention may include embodiments wherein other protocols are utilized. Additionally, the term “spanning tree” should be read broadly to include any data structure used to map the layout of network devices. “Internal spanning tree” should be read broadly to include any spanning tree used to update other spanning trees. “Multiple spanning tree” should be read broadly to include any spanning tree used for the routing of packets.

[0034] Additionally, whenever tables are mentioned in this document, the term should be interpreted broadly to encompass any data structure in any format.

[0035] Although illustrative embodiments and applications of this invention are shown and described herein, many variations and modifications are possible which remain within the concept, scope, and spirit of the invention, and these variations would become clear to those of ordinary skill in the art after perusal of this application. Accordingly, the embodiments described are to be considered as illustrative and not restrictive, and the invention is not to be limited to the details given herein, but may be modified within the scope and equivalents of the appended claims.

WE CLAIM:

1. A method comprising:
configuring a single port of a network device to support a first virtual device and a second virtual device; and
forming a single internal spanning tree instance for both the first virtual device and the second virtual device.
2. The method of claim 1, further comprising:
creating a mapping table, wherein the mapping table contains information about at least one network identification and at least one corresponding multiple spanning tree mapping identification.
3. The method of claim 2, wherein the network identifications are Virtual Local Area Network (VLAN) identifications.
4. The method of claim 2, wherein the creating includes:
identifying a first multiple spanning tree instance for the first virtual device having a first label identical to a second label of a second multiple spanning tree instance for the second device; and
adding, to the mapping table, multiple spanning tree mapping identifications that are unique within the mapping table for each of the first and second multiple spanning tree instances.
5. The method of claim 2, wherein the creating includes combining network identifications defined for the first virtual device with network identifications defined for the second virtual device.
6. The method of claim 5, wherein the creating further includes, for a network identification defined for both the first virtual device and the second virtual device, selecting only one multiple spanning tree instance to correspond to the network identification.
7. The method of claim 6, wherein the selecting includes prompting a user for a selection.

8. The method of claim 1, wherein the network device is a switch.
9. The method of claim 1, wherein the configuring a single internal spanning tree instance includes merging an internal spanning tree instance defined for the first virtual device with an internal spanning tree instance defined for the second virtual device.
10. The method of claim 1, wherein the forming a single spanning tree instance includes creating a single spanning tree for both the first and second virtual devices without referencing an internal spanning tree instance defined for the first virtual device or the second virtual device.
11. The method of claim 1, further comprising creating a port table, wherein the port table indicates, for the physical port, which network identifications are assigned to which virtual device.
12. A method comprising:
 - receiving a packet on a port;
 - identifying a VLAN for the packet;
 - ascertaining a first virtual device corresponding to the VLAN, wherein the first virtual device shares the port with a second virtual device; and
 - determining a multiple spanning tree instance to use for routing the packet by checking a data structure containing unique identifications for each of a plurality of multiple spanning tree instances corresponding to the port.
13. The method of claim 12, wherein the classifying includes examining a VLAN tag in the packet.
14. The method of claim 12, wherein the ascertaining includes checking a port table corresponding to the, the port table including one or more VLAN identifications and corresponding virtual devices.
15. The method of claim 12, wherein the ascertaining includes checking a mapping table.

16. The method of claim 15, wherein the mapping table includes contains information about one or more network identifications and corresponding multiple spanning tree mapping identifications.
17. A switch comprising:
one or more line cards, wherein at least one of the one or more line cards is configured to:
 configure a single port of a network device to support a first virtual device and a second virtual device; and
 form a single internal spanning tree instance for both the first virtual device and the second virtual device.
18. A switch comprising:
one or more line cards, wherein at least one of the one or more line cards is configured to:
 receive a packet on a port;
 identify a VLAN for the packet;
 ascertain a first virtual device corresponding to the VLAN, wherein the first virtual device shares the port with a second virtual device; and
 determine a multiple spanning tree instance to use for routing the packet by checking a data structure containing unique identifications for each of a plurality of multiple spanning tree instances corresponding to the port.
19. An apparatus comprising:
 means for configuring a single port of a network device to support a first virtual device and a second virtual device; and
 means for forming a single spanning tree instance for both the first virtual device and the second virtual device.
20. An apparatus comprising:
 means for receiving a packet on a port;
 means for identifying a VLAN for the packet;
 means for ascertaining a first virtual device corresponding to the VLAN, wherein the first virtual device shares the port with a second virtual device; and

means for determining a multiple spanning tree instance to use for routing the packet by checking a data structure containing unique identifications for each of a plurality of multiple spanning tree instances corresponding to the port.

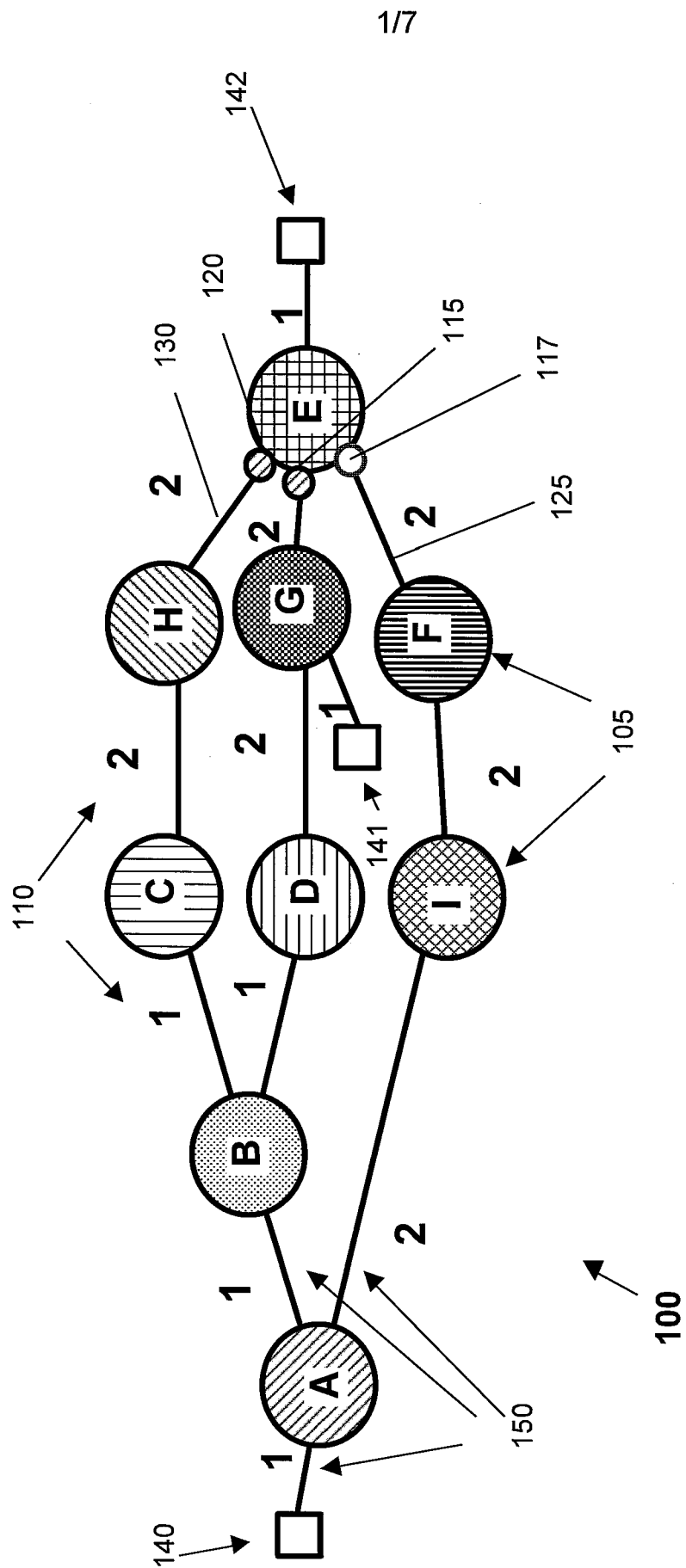
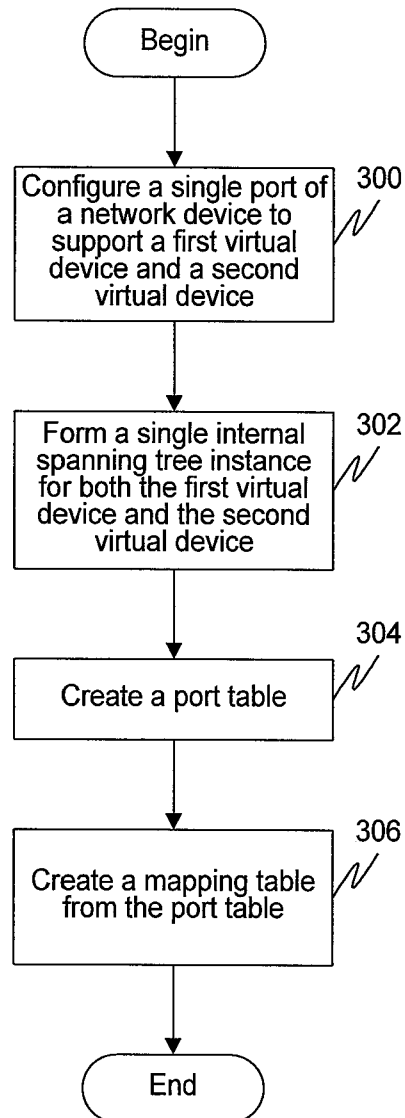


FIG. 1

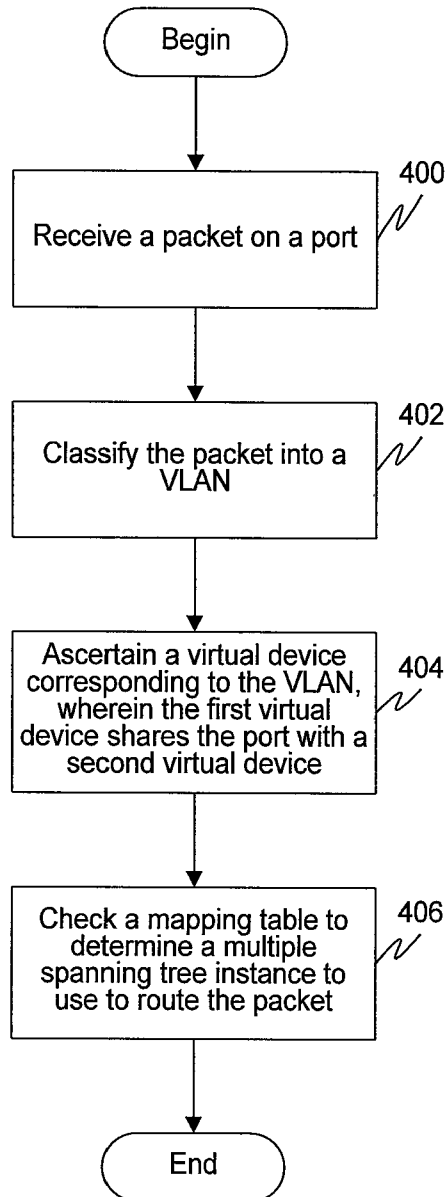
200	202	204	206
<u>VD</u>	<u>VLAN</u>	<u>MSTID</u>	<u>MSTMID</u>
VD1	VLAN1	MST1	MSTID1
VD1	VLAN2	MST2	MSTID2
VD1	VLAN3	MST3	MSTID3
VD1	VLAN4	MST4	MSTID4
VD2	VLAN3	MST3	MSTID3
VD2	VLAN4	MST4	MSTID4
VD2	VLAN5	MST1	MSTID5
VD2	VLAN6	MST2	MSTID6

FIG. 2

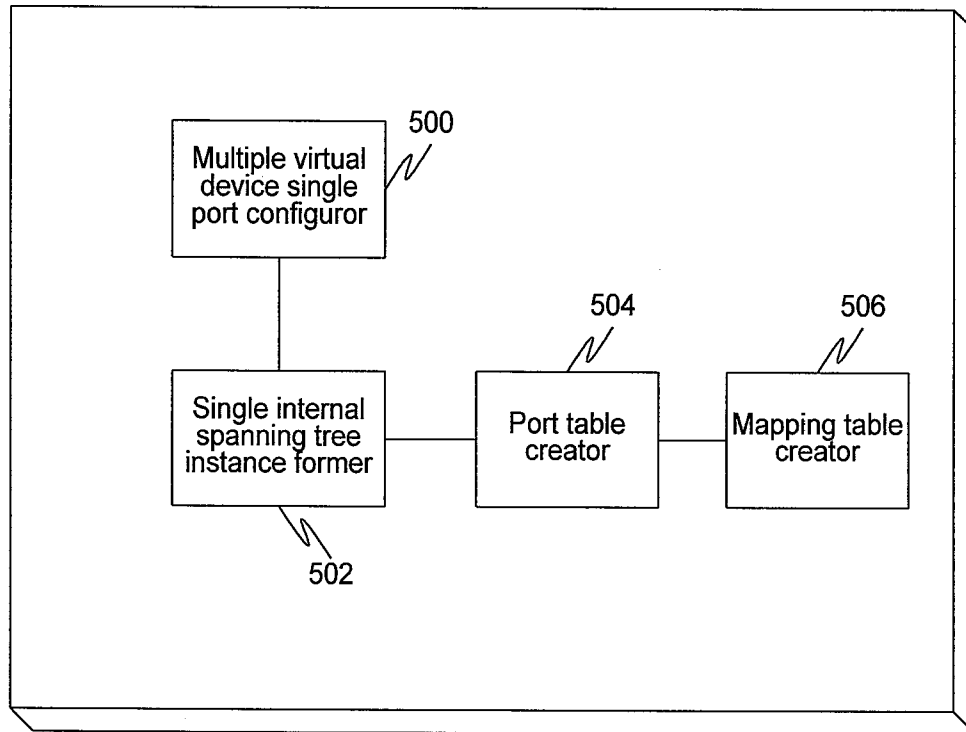
3/7

**FIG. 3**

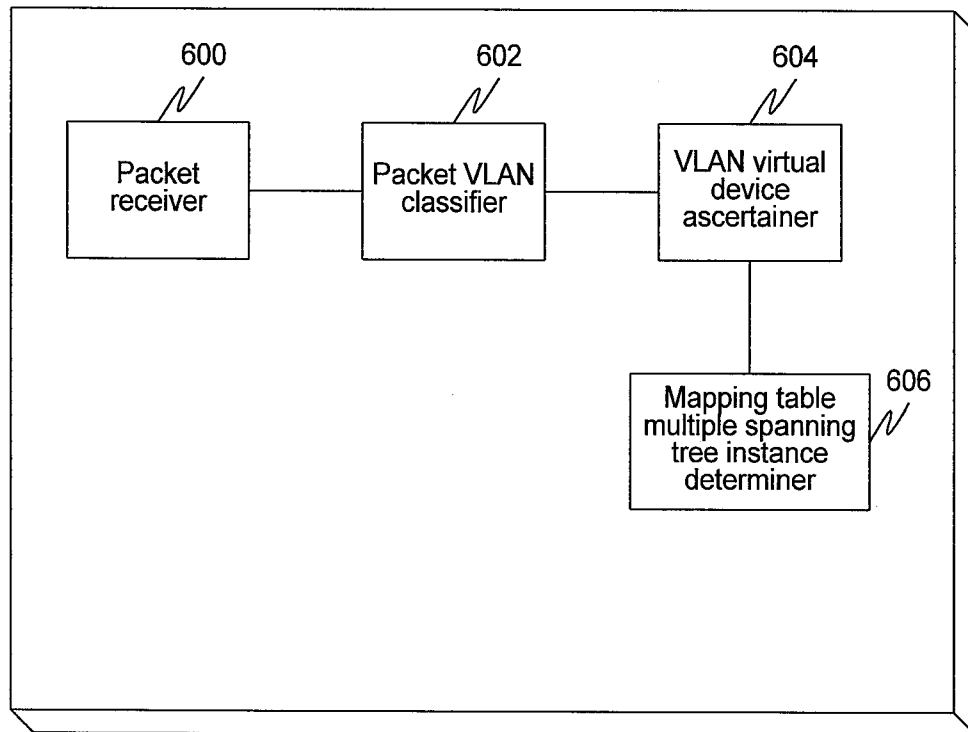
4/7

**FIG. 4**

5/7

**FIG. 5**

6/7

**FIG. 6**

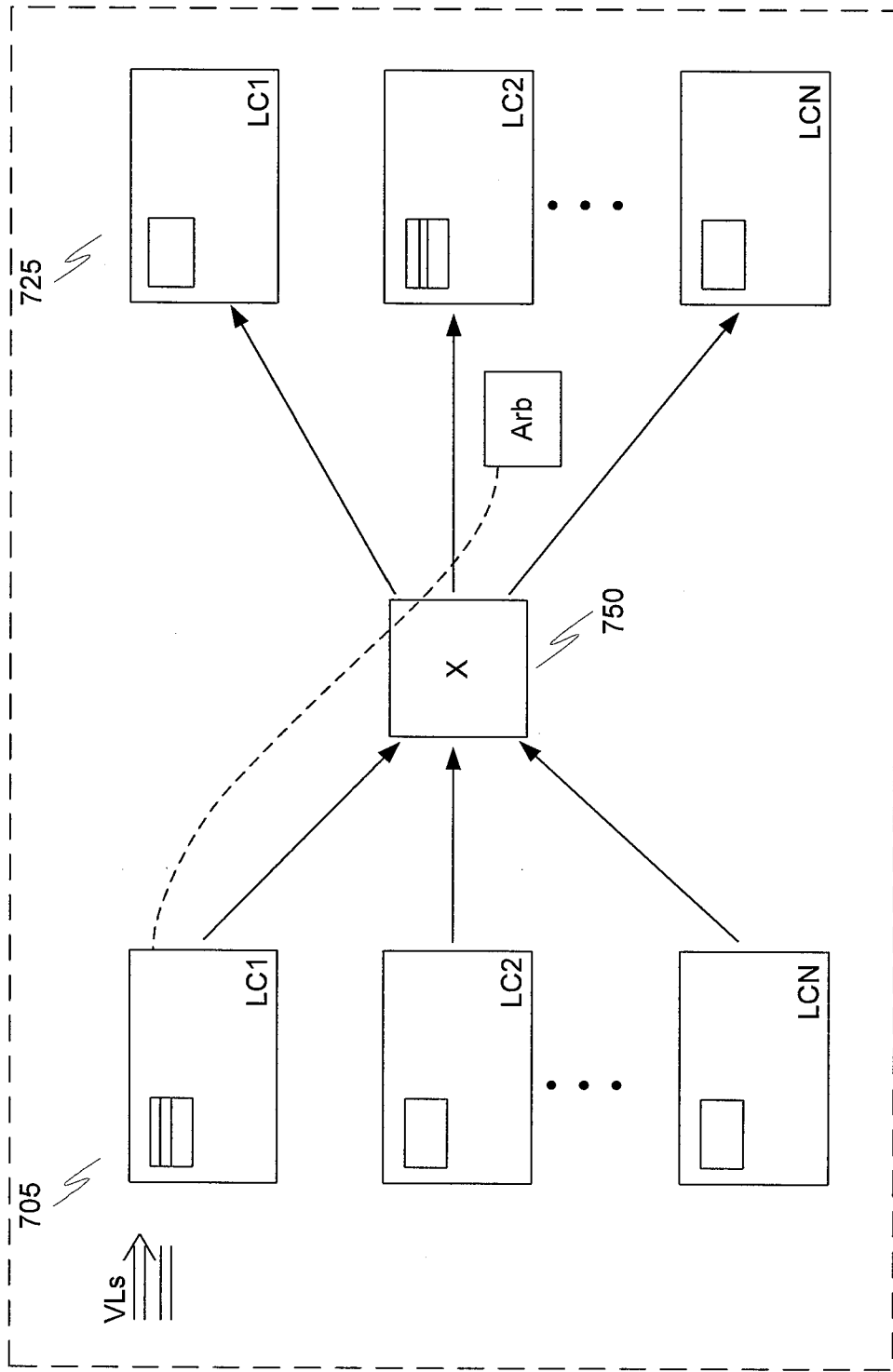


FIG. 7