US 20120201122A1

(19) **United States**

(12) **Patent Application Publication**

Mohan

(10) **Pub. No.: US 2012/0201122 A1**

(43) **Pub. Date: Aug. 9, 2012**

(54) **MULTI-POINT AND ROOTED MULTI-POINT PROTECTION SWITCHING**

(75) Inventor: **Dinesh Mohan**, Kanata (CA)

(73) Assignee: **Rockstar Bidco, LP**, New York, NY (US)

(21) Appl. No.: **13/446,469**

(22) Filed: **Apr. 13, 2012**

Related U.S. Application Data

(63) Continuation of application No. 12/250,266, filed on Oct. 13, 2008, now Pat. No. 8,165,031.

(60) Provisional application No. 60/979,449, filed on Oct. 12, 2007.

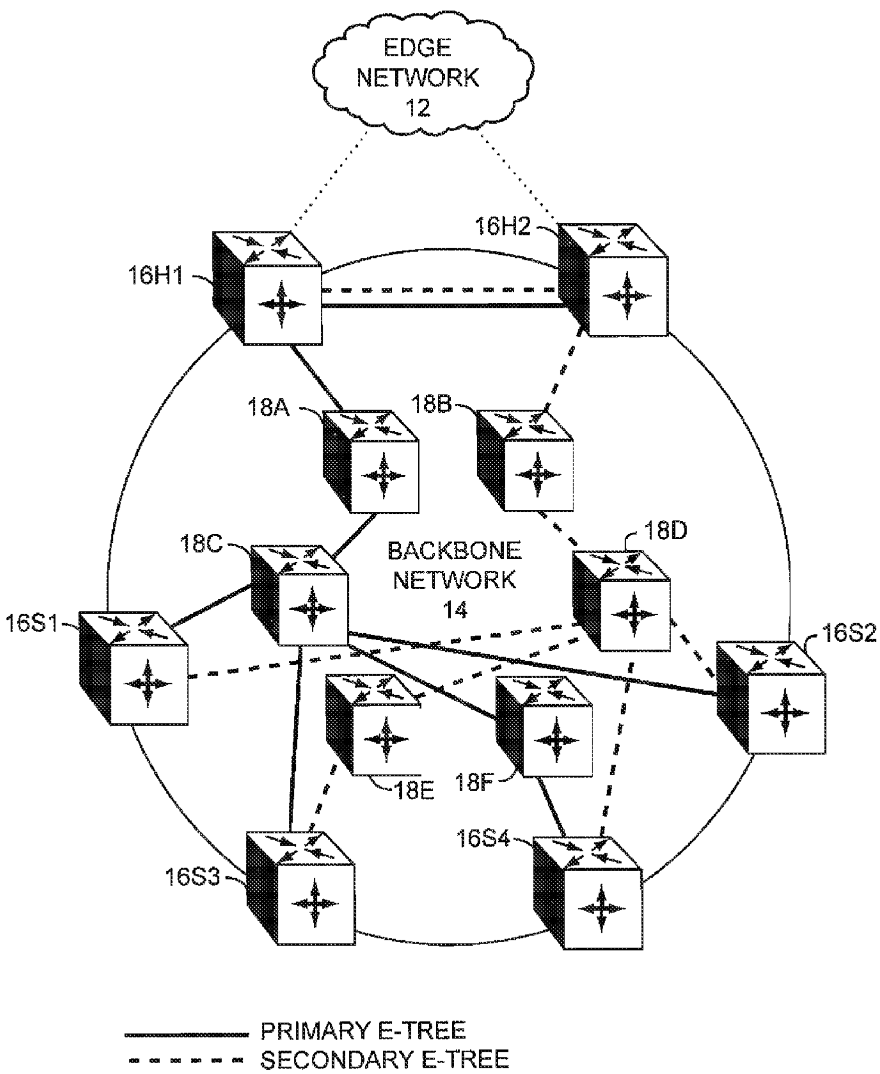
Publication Classification

(51) **Int. Cl.**
H04L 12/24 (2006.01)

(52) **U.S. Cl.** **370/217**

(57) **ABSTRACT**

The present invention relates to techniques for allowing one or more edge nodes in a backbone network to quickly and efficiently switch traffic delivery from a first virtual network to a second virtual network in response to a failure occurring in association with the first virtual network. In certain embodiments, an edge node is capable of independently detecting that a failure has occurred on the first virtual network and quickly transitioning from the first virtual network to the second virtual network for receiving or delivering traffic. Upon detecting the failure in the first virtual network, the edge node will begin delivering traffic over the second virtual network. If control messages are not already being provided over the second network, the edge node may begin providing the control messages over the second virtual network.



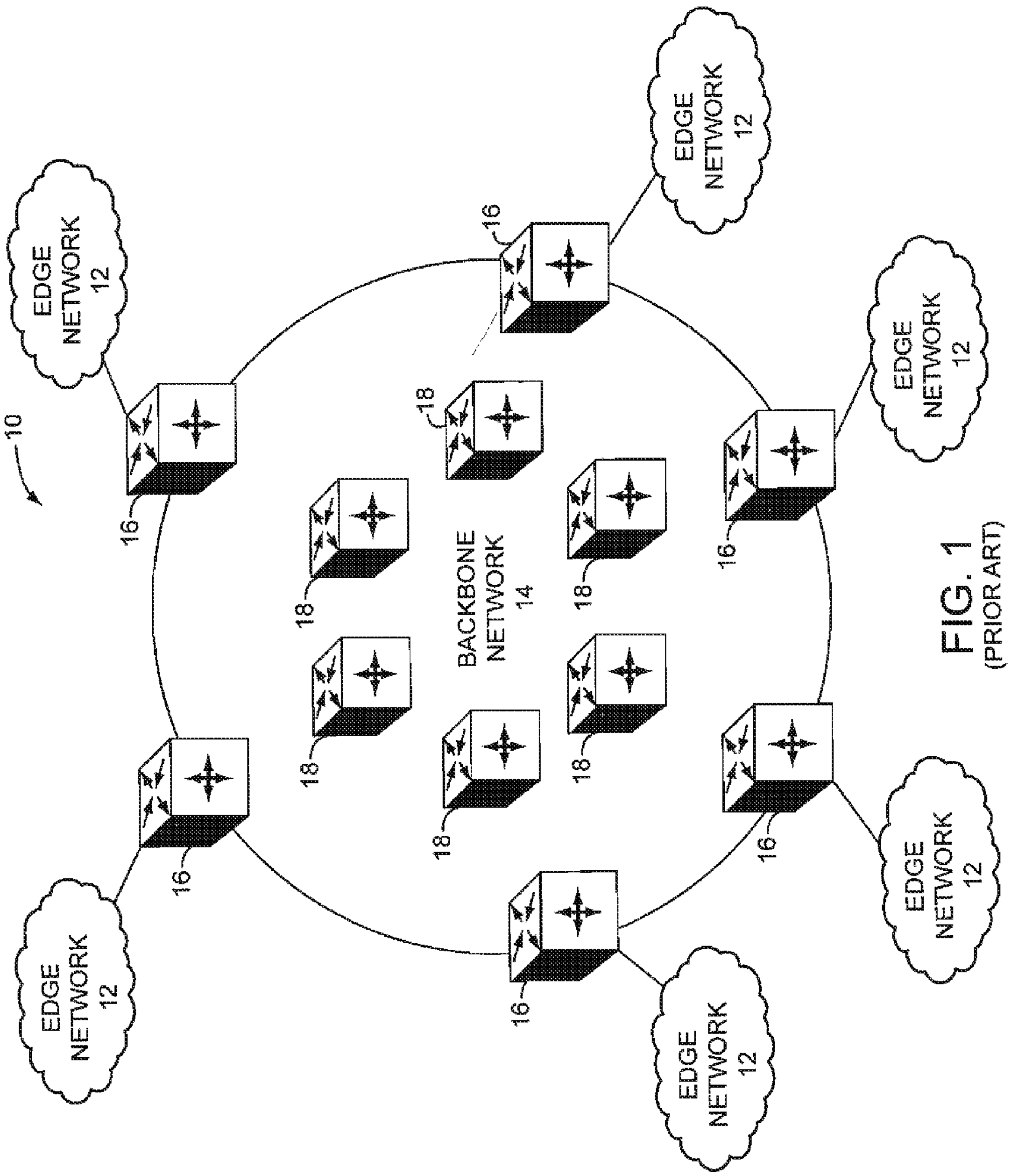
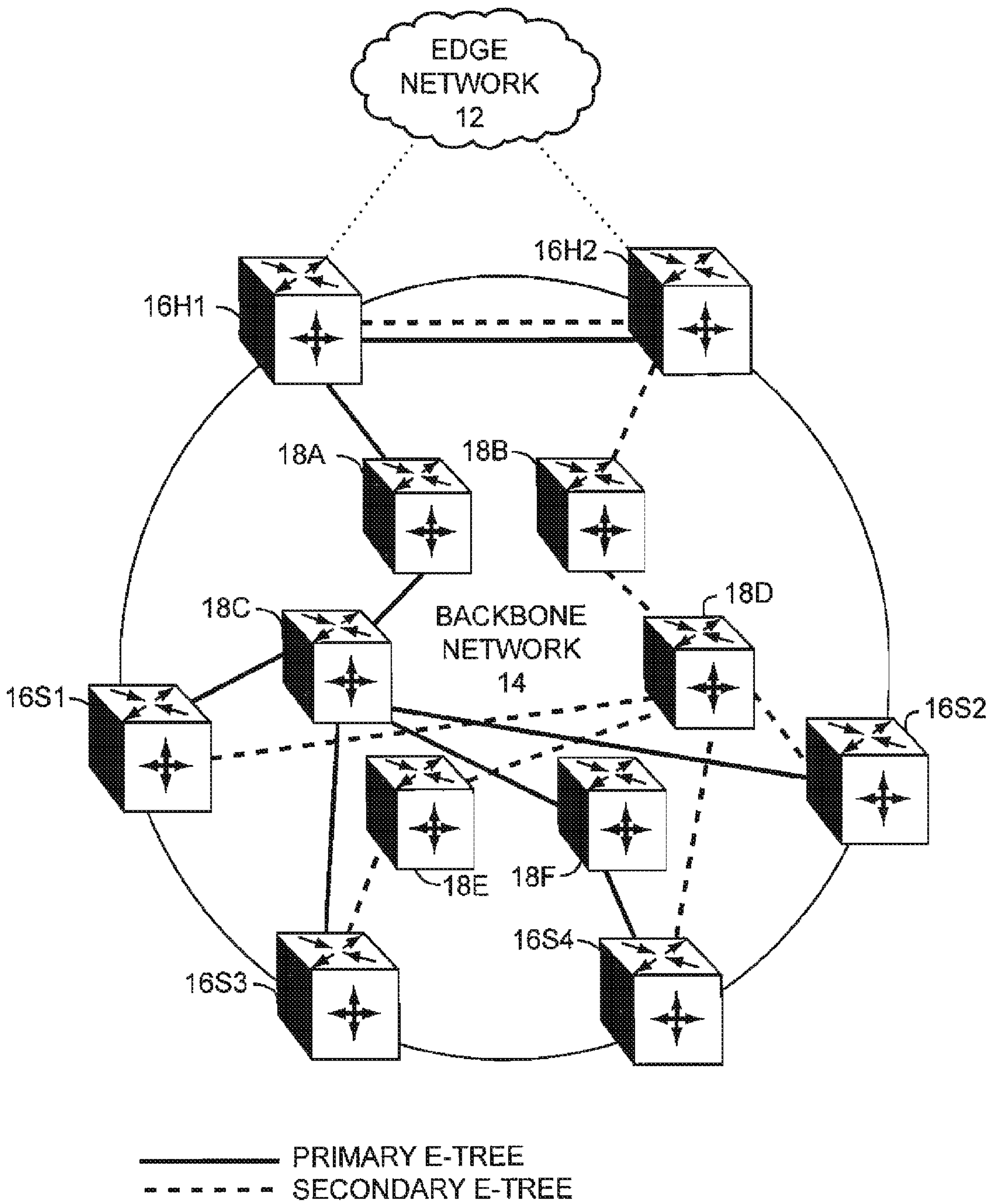


FIG. 1
(PRIOR ART)



EDGE
NETWORK
12
16H1
16H2
18A
18B
18C
18D
BACKBONE
NETWORK
14
16S1
16S2
18E
18F
16S3
16S4
PRIMARY E-TREE
SECONDARY E-TREE

FIG. 2

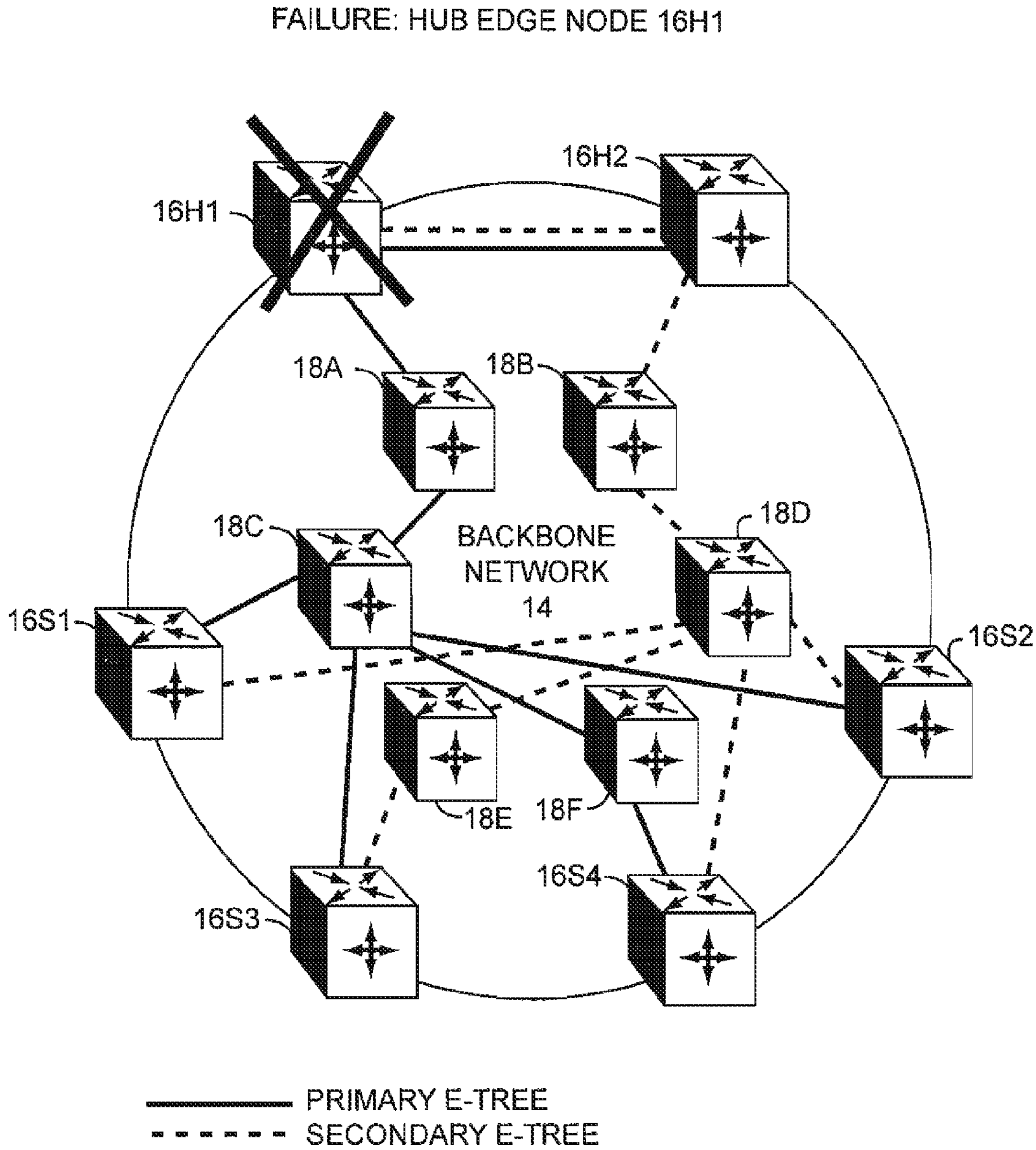

FAILURE: HUB EDGE NODE 16H1
16H1
16H2
18A
18B
18C
18D
BACKBONE NETWORK 14
16S1
16S2
18E
18F
16S3
16S4
PRIMARY E-TREE
SECONDARY E-TREE

FIG. 3

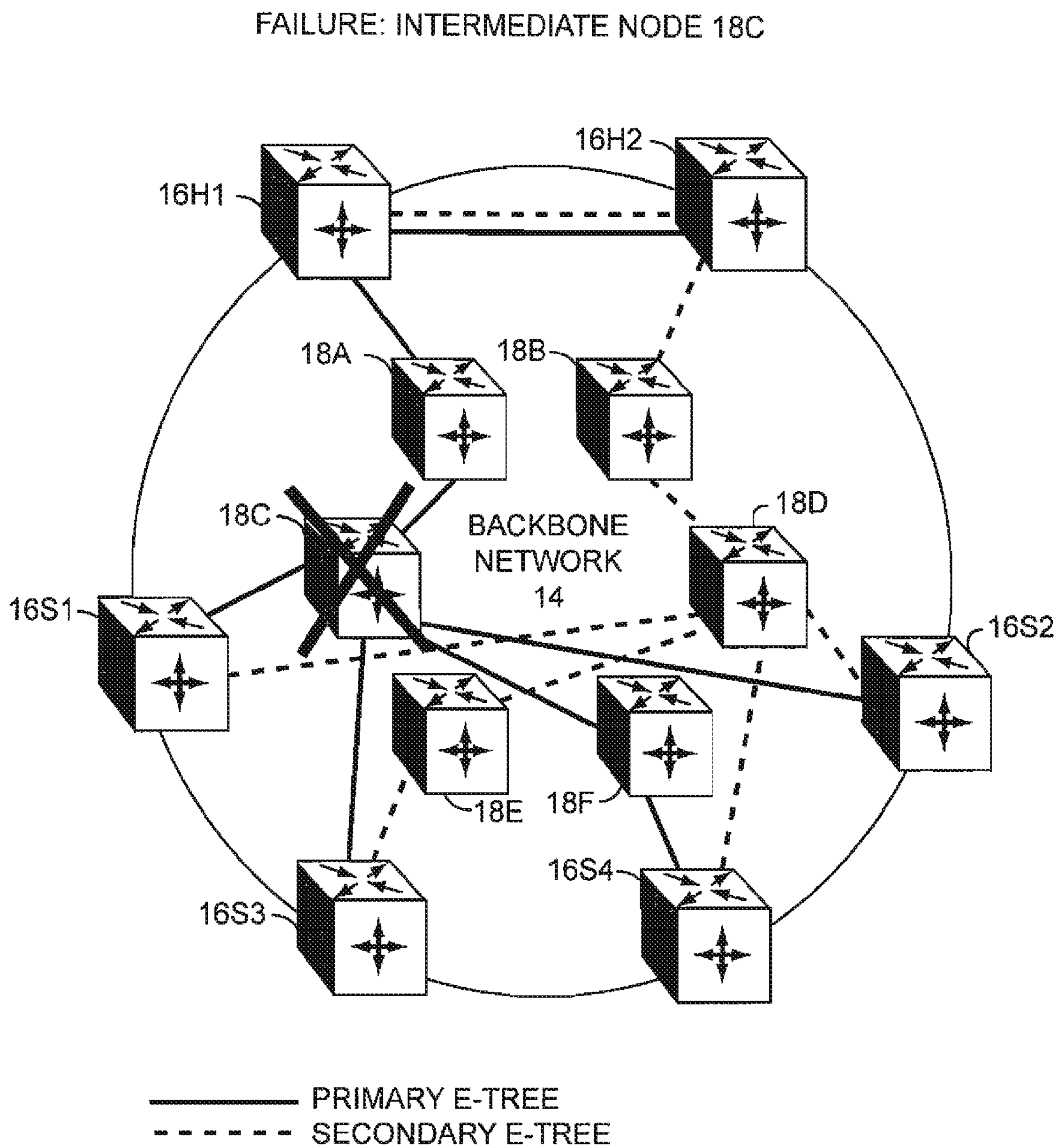

FAILURE: INTERMEDIATE NODE 18C
16H1
16H2
18A
18B
18C
18D
BACKBONE
NETWORK
14
16S1
16S2
18E
18F
16S3
16S4
PRIMARY E-TREE
SECONDARY E-TREE

FIG. 4

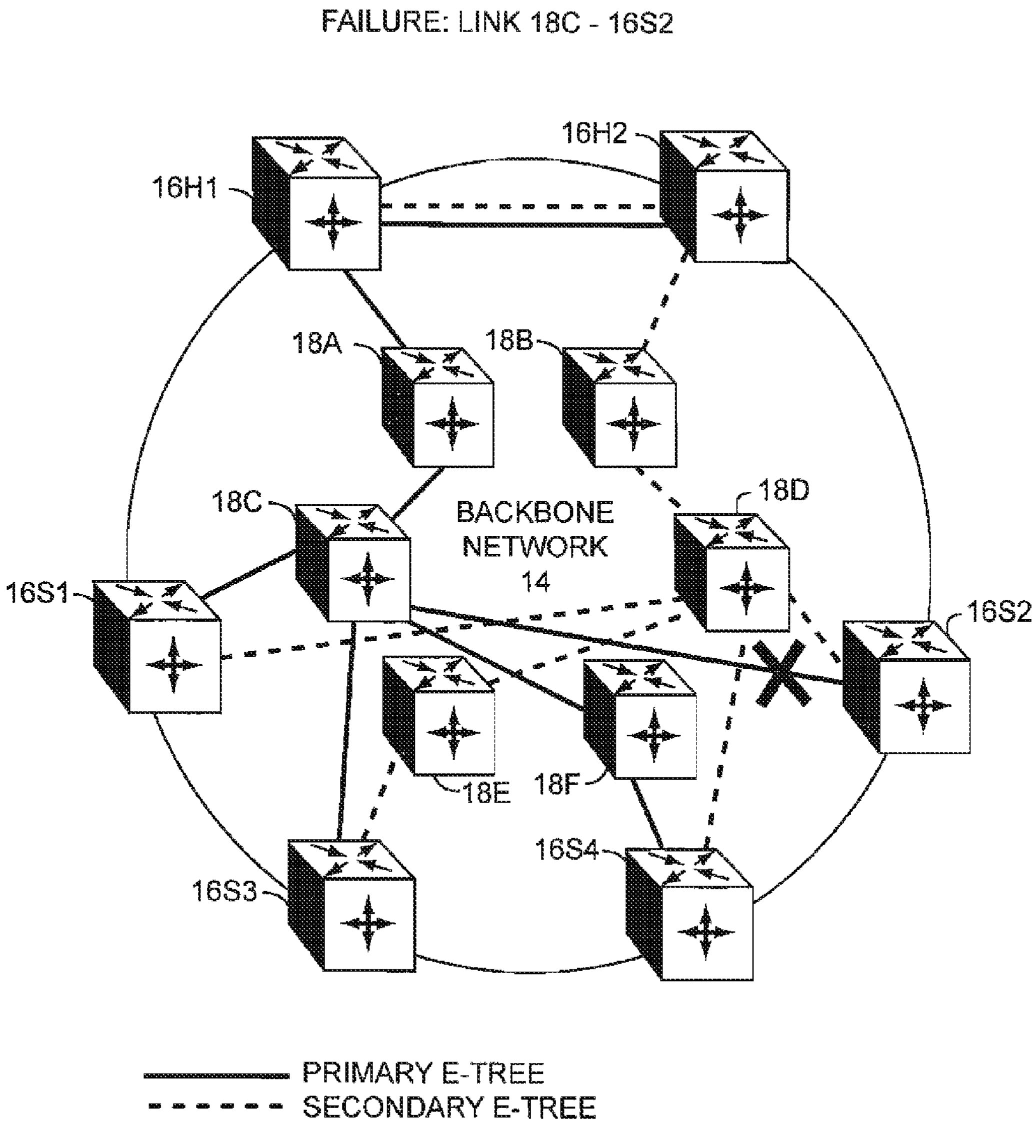

FAILURE: LINK 18C - 16S2
16H1
16H2
18A
18B
18C
18D
BACKBONE
NETWORK
14
16S1
16S2
18E
18F
16S3
16S4
PRIMARY E-TREE
SECONDARY E-TREE

FIG. 5

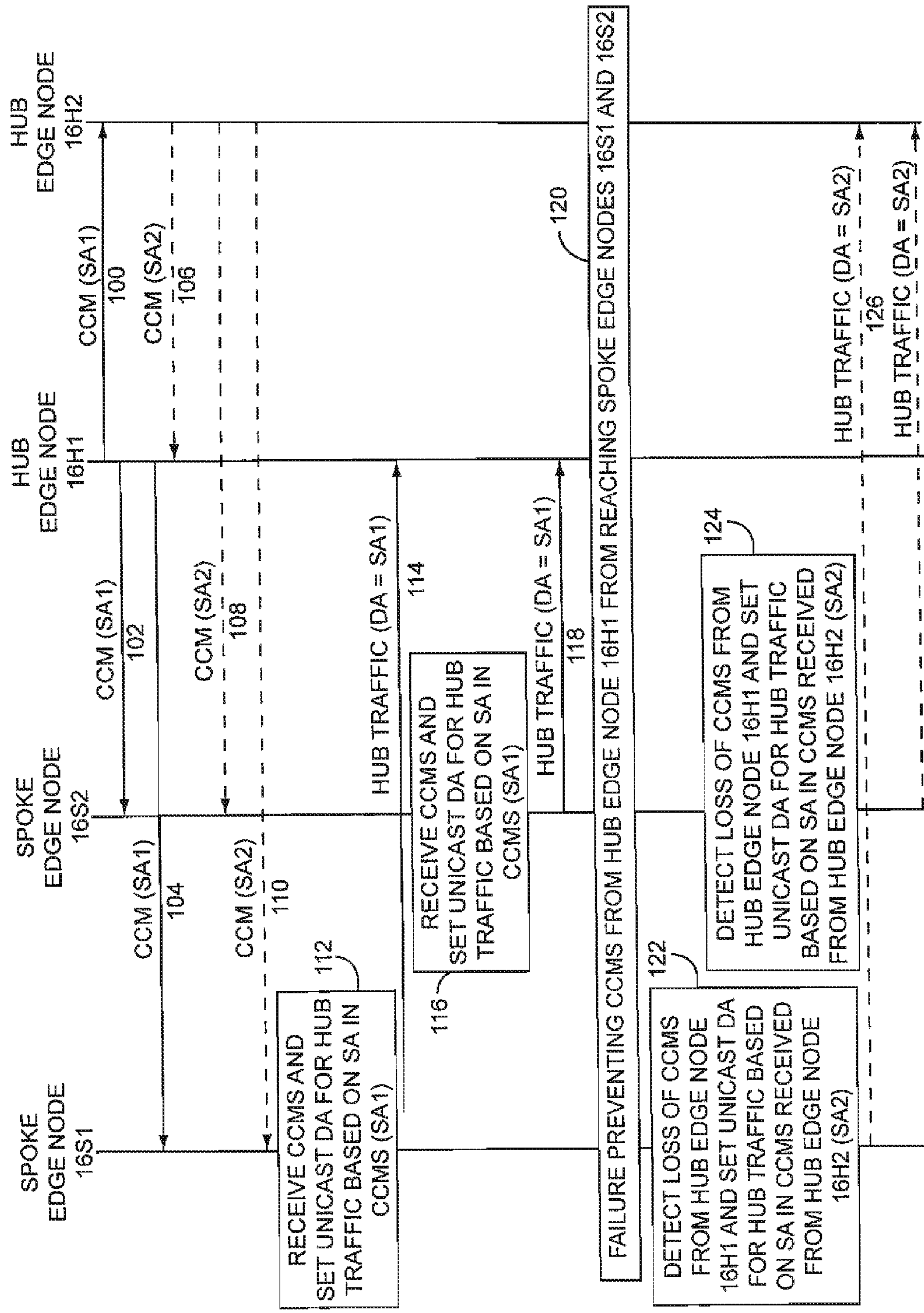

SPOKE EDGE NODE 16S1
SPOKE EDGE NODE 16S2
HUB EDGE NODE 16H1
HUB EDGE NODE 16H2
CCM (SA1) 100
CCM (SA2) 106
CCM (SA1) 102
CCM (SA2) 108
CCM (SA1) 104
CCM (SA2) 110
RECEIVE CCMS AND SET UNICAST DA FOR HUB TRAFFIC BASED ON SA IN CCMS (SA1)
112
HUB TRAFFIC (DA = SA1)
114
RECEIVE CCMS AND SET UNICAST DA FOR HUB TRAFFIC BASED ON SA IN CCMS (SA1)
116
HUB TRAFFIC (DA = SA1)
118
FAILURE PREVENTING CCMS FROM HUB EDGE NODE 16H1 FROM REACHING SPOKE EDGE NODES 16S1 AND 16S2
120
DETECT LOSS OF CCMS FROM HUB EDGE NODE 16H1 AND SET UNICAST DA FOR HUB TRAFFIC BASED ON SA IN CCMS RECEIVED FROM HUB EDGE NODE 16H2 (SA2)
122
DETECT LOSS OF CCMS FROM HUB EDGE NODE 16H1 AND SET UNICAST DA FOR HUB TRAFFIC BASED ON SA IN CCMS RECEIVED FROM HUB EDGE NODE 16H2 (SA2)
124
HUB TRAFFIC (DA = SA2)
126
HUB TRAFFIC (DA = SA2)
128

FIG. 6

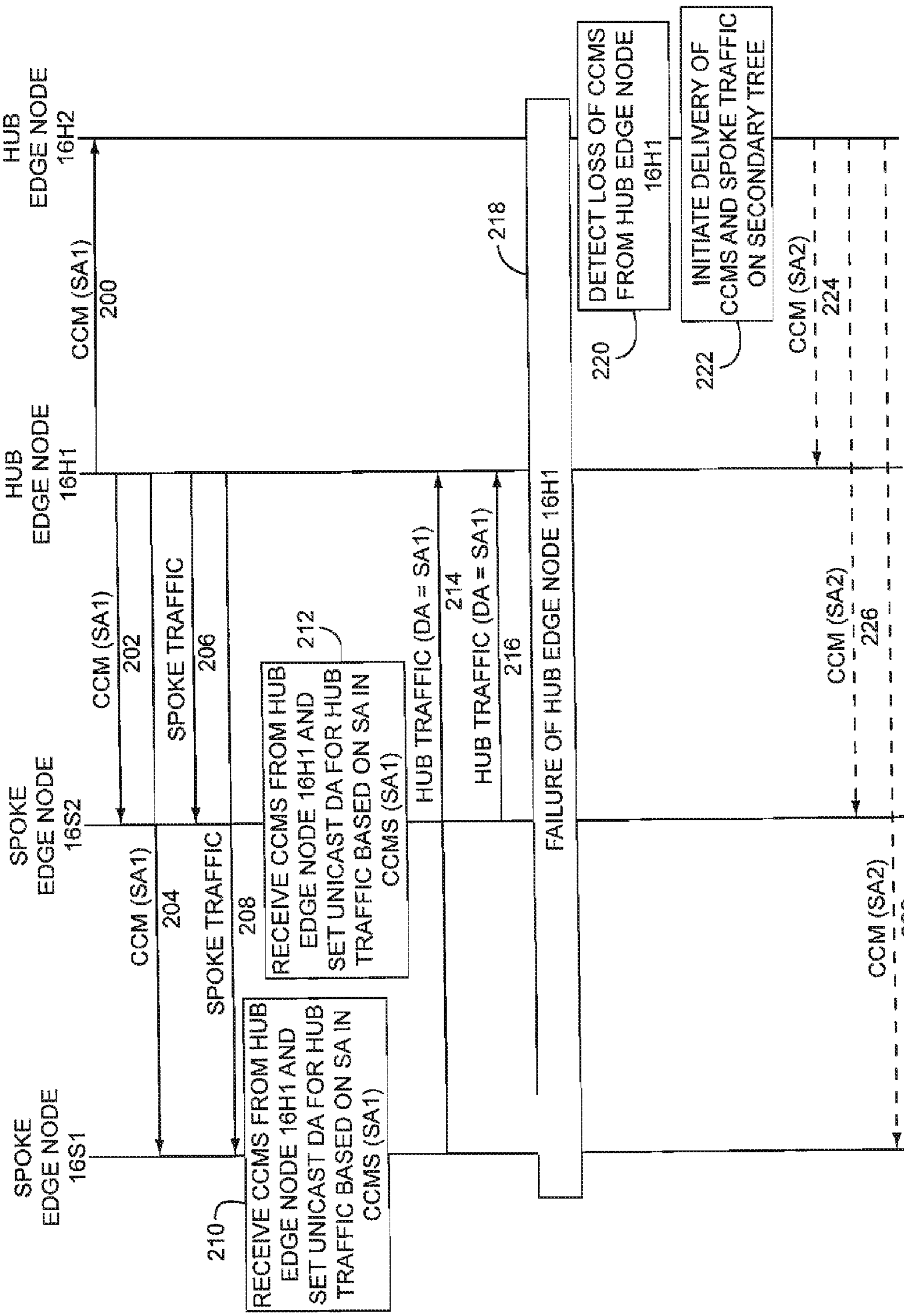

SPOKE EDGE NODE 16S1
SPOKE EDGE NODE 16S2
HUB EDGE NODE 16H1
HUB EDGE NODE 16H2
CCM (SA1)
200
CCM (SA1)
202
CCM (SA1)
204
SPOKE TRAFFIC
206
SPOKE TRAFFIC
208
RECEIVE CCMS FROM HUB EDGE NODE 16H1 AND SET UNICAST DA FOR HUB TRAFFIC BASED ON SA IN CCMS (SA1)
210
RECEIVE CCMS FROM HUB EDGE NODE 16H1 AND SET UNICAST DA FOR HUB TRAFFIC BASED ON SA IN CCMS (SA1)
212
HUB TRAFFIC (DA = SA1)
214
HUB TRAFFIC (DA = SA1)
216
FAILURE OF HUB EDGE NODE 16H1
218
DETECT LOSS OF CCMS FROM HUB EDGE NODE 16H1
220
INITIATE DELIVERY OF CCMS AND SPOKE TRAFFIC ON SECONDARY TREE
222
CCM (SA2)
224
CCM (SA2)
226
CCM (SA2)
228

FIG. 7A

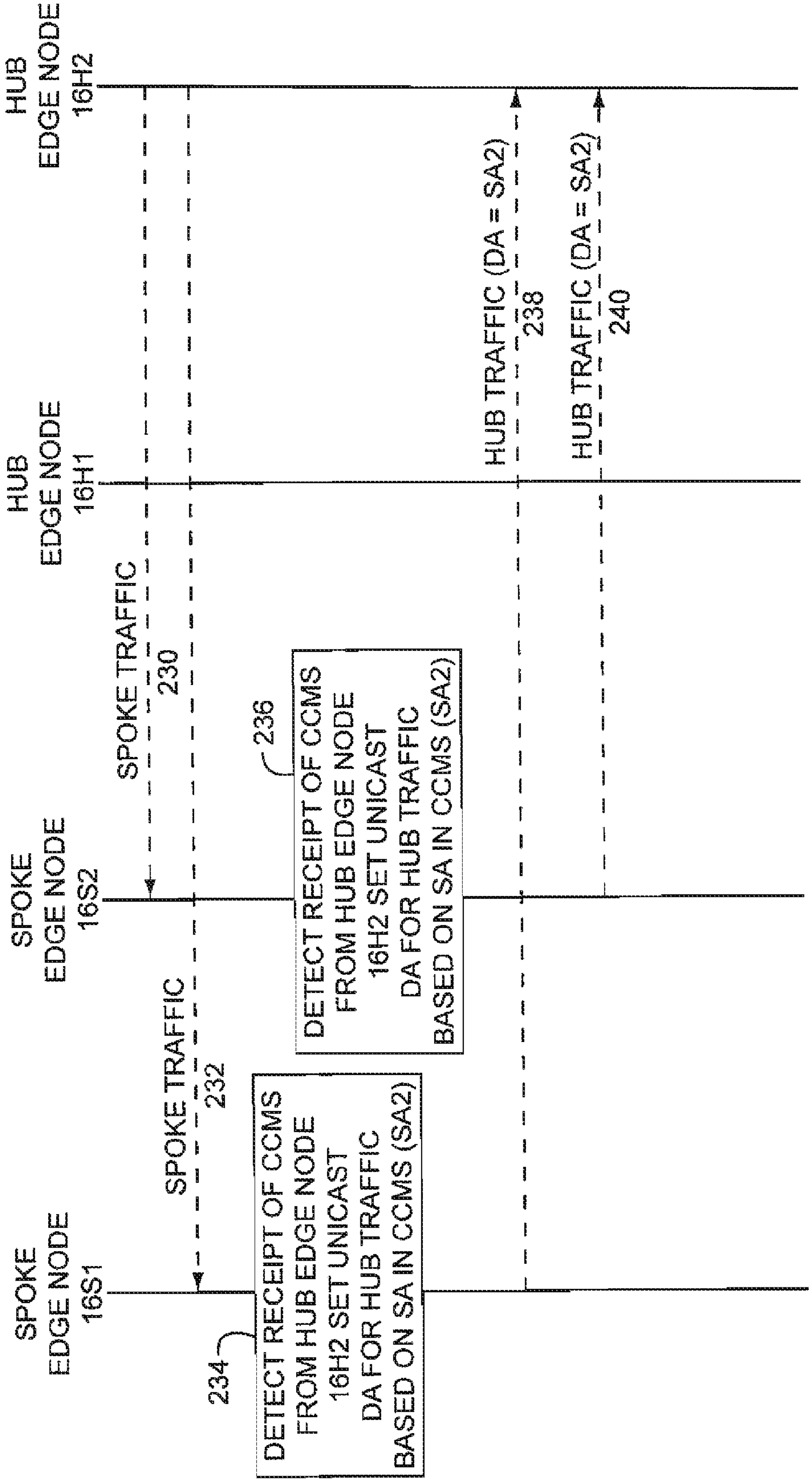
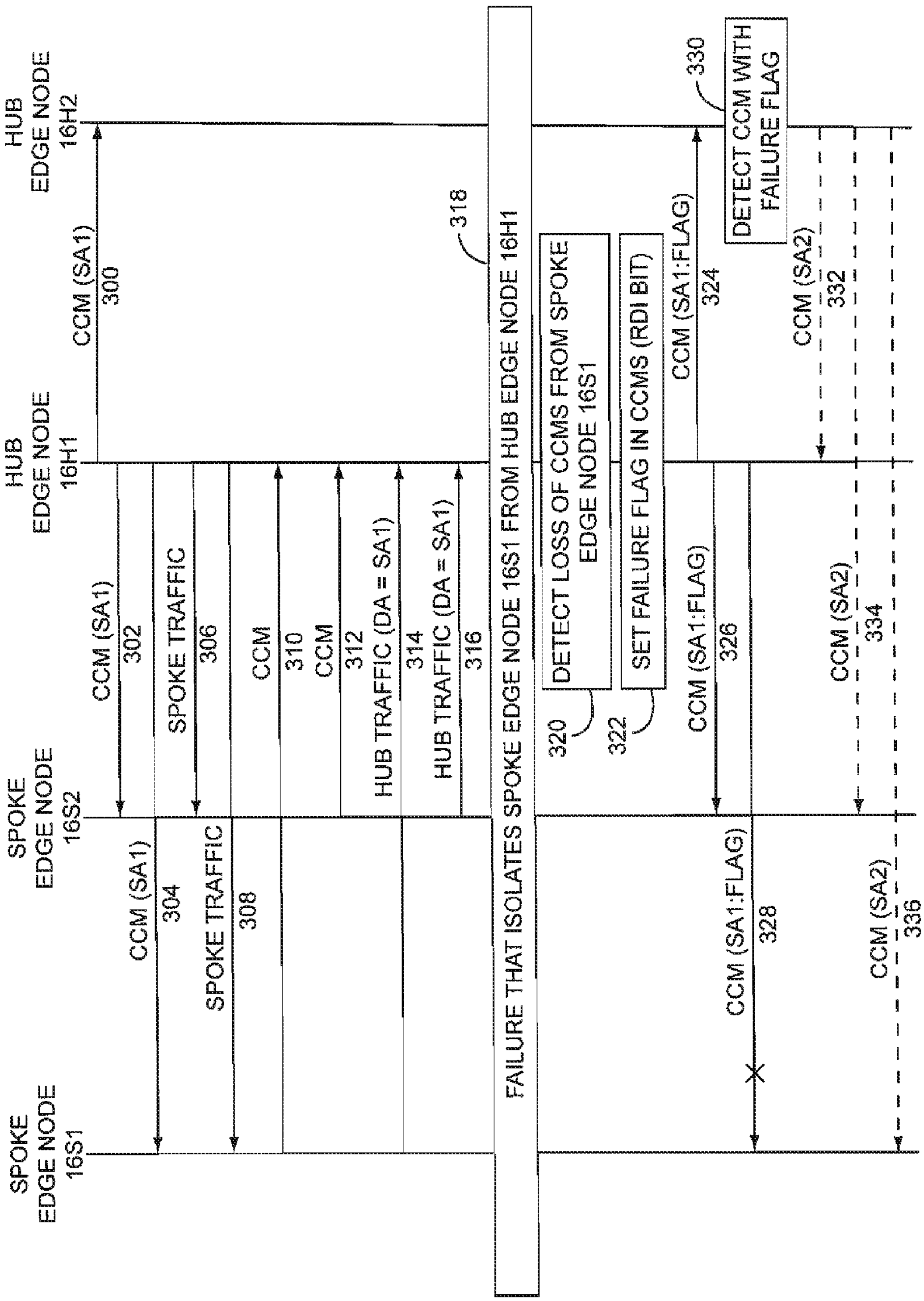

SPOKE EDGE NODE 16S1
SPOKE EDGE NODE 16S2
HUB EDGE NODE 16H1
HUB EDGE NODE 16H2
SPOKE TRAFFIC 230
SPOKE TRAFFIC 232
234
DETECT RECEIPT OF CCMS FROM HUB EDGE NODE 16H2 SET UNICAST DA FOR HUB TRAFFIC BASED ON SA IN CCMS (SA2)
236
DETECT RECEIPT OF CCMS FROM HUB EDGE NODE 16H2 SET UNICAST DA FOR HUB TRAFFIC BASED ON SA IN CCMS (SA2)
HUB TRAFFIC (DA = SA2) 238
HUB TRAFFIC (DA = SA2) 240

FIG. 7B



SPOKE EDGE NODE 16S1
SPOKE EDGE NODE 16S2
HUB EDGE NODE 16H1
HUB EDGE NODE 16H2
CCM (SA1) 300
CCM (SA1) 302
CCM (SA1) 304
SPOKE TRAFFIC 306
SPOKE TRAFFIC 308
CCM 310
CCM 312
HUB TRAFFIC (DA = SA1) 314
HUB TRAFFIC (DA = SA1) 316
FAILURE THAT ISOLATES SPOKE EDGE NODE 16S1 FROM HUB EDGE NODE 16H1
318
DETECT LOSS OF CCMS FROM SPOKE EDGE NODE 16S1
320
SET FAILURE FLAG IN CCMS (RDI BIT)
322
CCM (SA1:FLAG) 324
CCM (SA1:FLAG) 326
CCM (SA1:FLAG) 328
DETECT CCM WITH FAILURE FLAG
330
CCM (SA2) 332
CCM (SA2) 334
CCM (SA2) 336

FIG. 8A

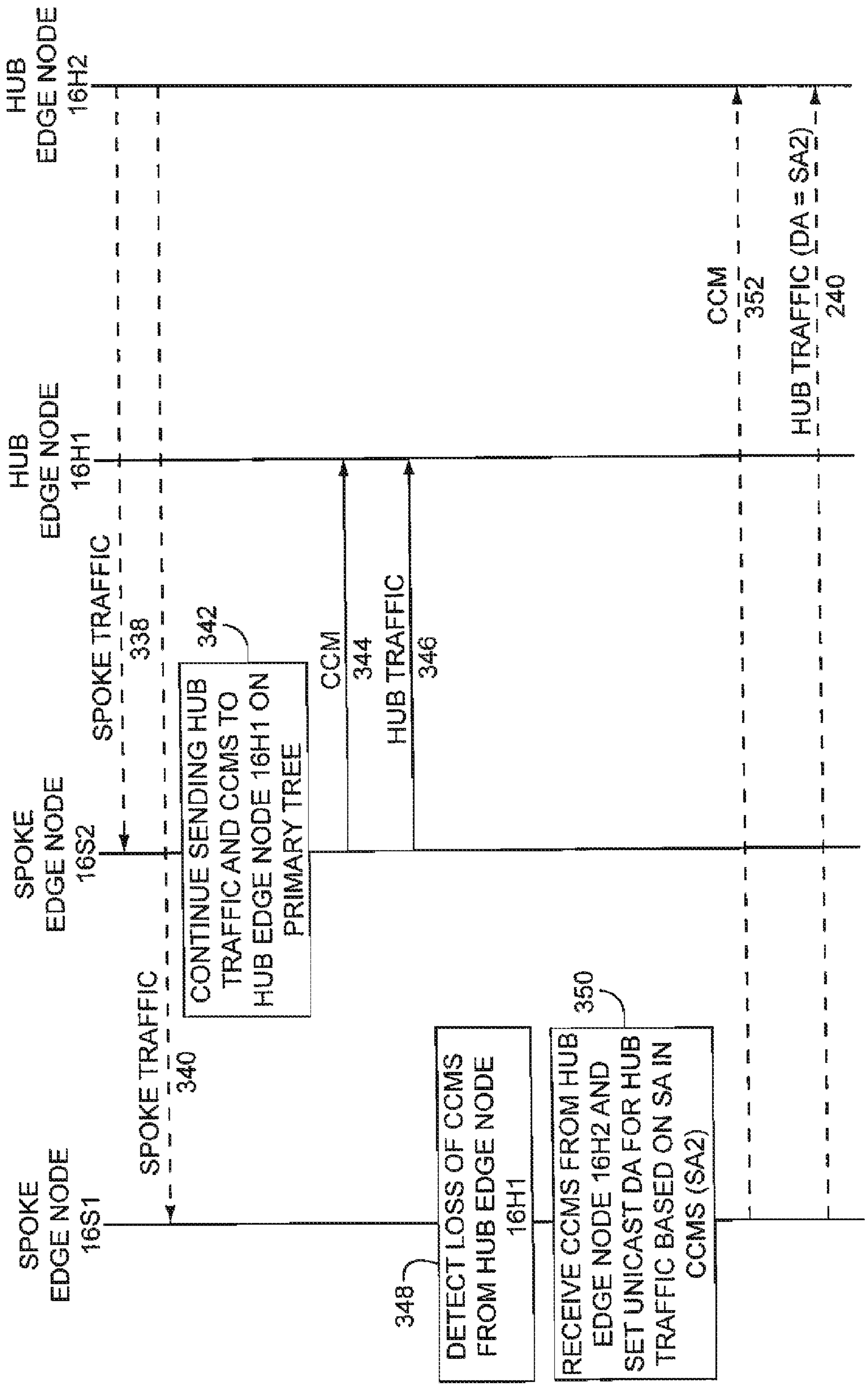

SPOKE EDGE NODE 16S1
SPOKE EDGE NODE 16S2
HUB EDGE NODE 16H1
HUB EDGE NODE 16H2
SPOKE TRAFFIC 338
SPOKE TRAFFIC 340
342
CONTINUE SENDING HUB TRAFFIC AND CCMS TO HUB EDGE NODE 16H1 ON PRIMARY TREE
CCM 344
HUB TRAFFIC 346
348
DETECT LOSS OF CCMS FROM HUB EDGE NODE 16H1
350
RECEIVE CCMS FROM HUB EDGE NODE 16H2 AND SET UNICAST DA FOR HUB TRAFFIC BASED ON SA IN CCMS (SA2)
CCM 352
HUB TRAFFIC (DA = SA2) 240

FIG. 8B

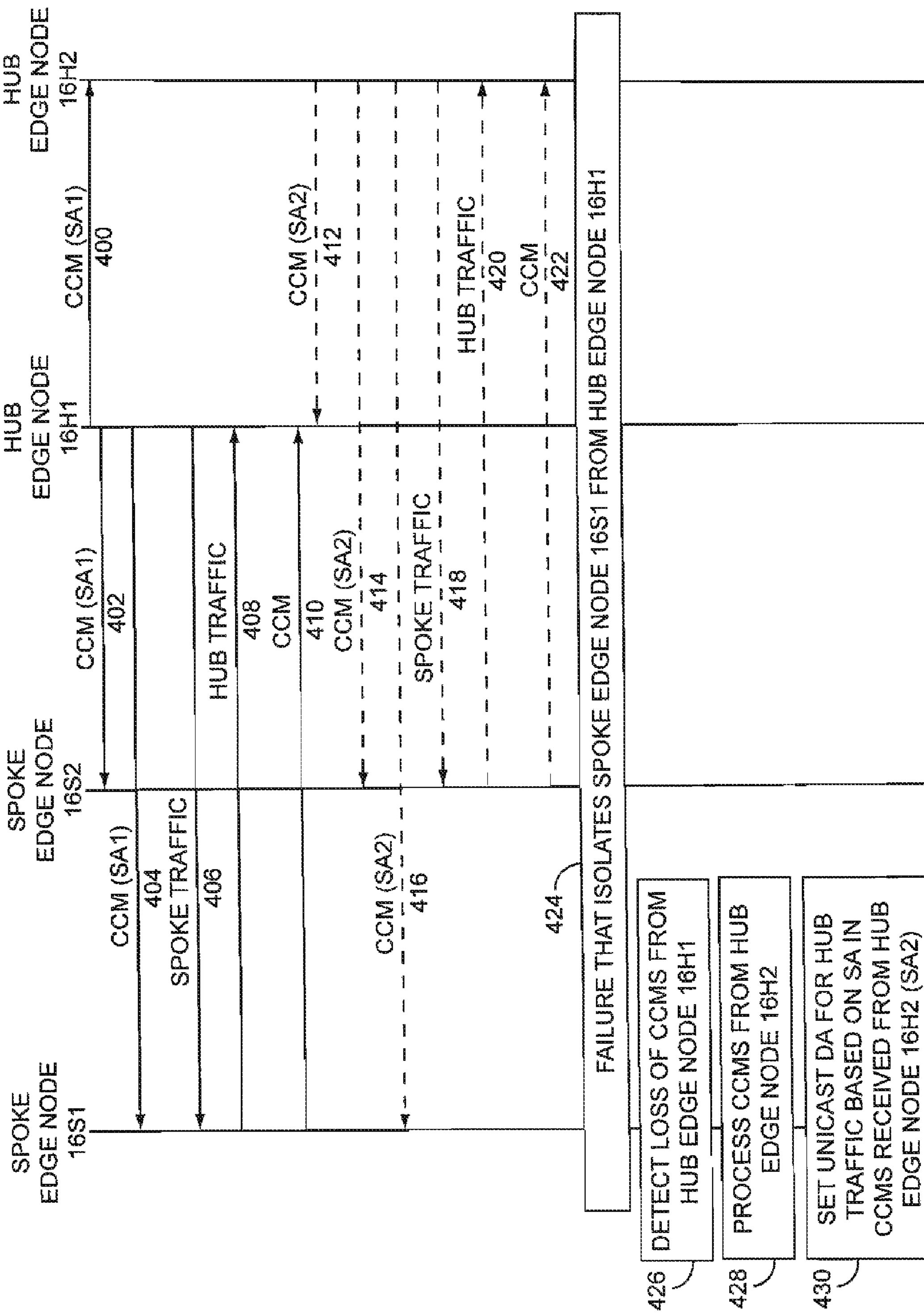

SPOKE EDGE NODE 16S1
SPOKE EDGE NODE 16S2
HUB EDGE NODE 16H1
HUB EDGE NODE 16H2
CCM (SA1) 400
CCM (SA1) 402
CCM (SA1) 404
SPOKE TRAFFIC 406
HUB TRAFFIC 408
CCM 410
CCM (SA2) 412
CCM (SA2) 414
CCM (SA2) 416
SPOKE TRAFFIC 418
HUB TRAFFIC 420
CCM 422
424
FAILURE THAT ISOLATES SPOKE EDGE NODE 16S1 FROM HUB EDGE NODE 16H1
426
DETECT LOSS OF CCMS FROM HUB EDGE NODE 16H1
428
PROCESS CCMS FROM HUB EDGE NODE 16H2
430
SET UNICAST DA FOR HUB TRAFFIC BASED ON SA IN CCMS RECEIVED FROM HUB EDGE NODE 16H2 (SA2)

FIG. 9A

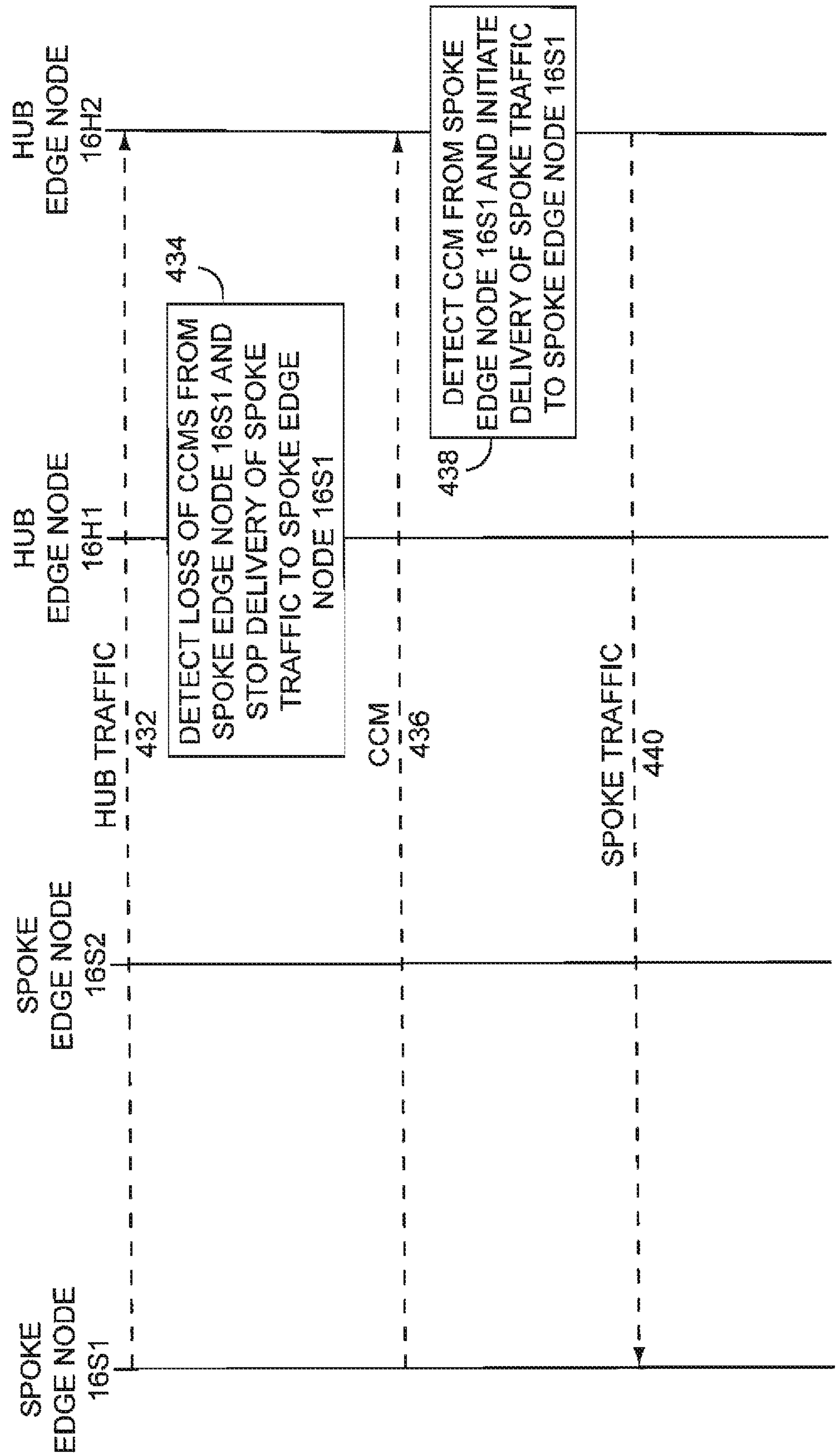

SPOKE EDGE NODE 16S1
SPOKE EDGE NODE 16S2
HUB EDGE NODE 16H1
HUB EDGE NODE 16H2
HUB TRAFFIC 432
DETECT LOSS OF CCMS FROM SPOKE EDGE NODE 16S1 AND STOP DELIVERY OF SPOKE TRAFFIC TO SPOKE EDGE NODE 16S1
434
CCM 436
DETECT CCM FROM SPOKE EDGE NODE 16S1 AND INITIATE DELIVERY OF SPOKE TRAFFIC TO SPOKE EDGE NODE 16S1
438
SPOKE TRAFFIC 440

FIG. 9B

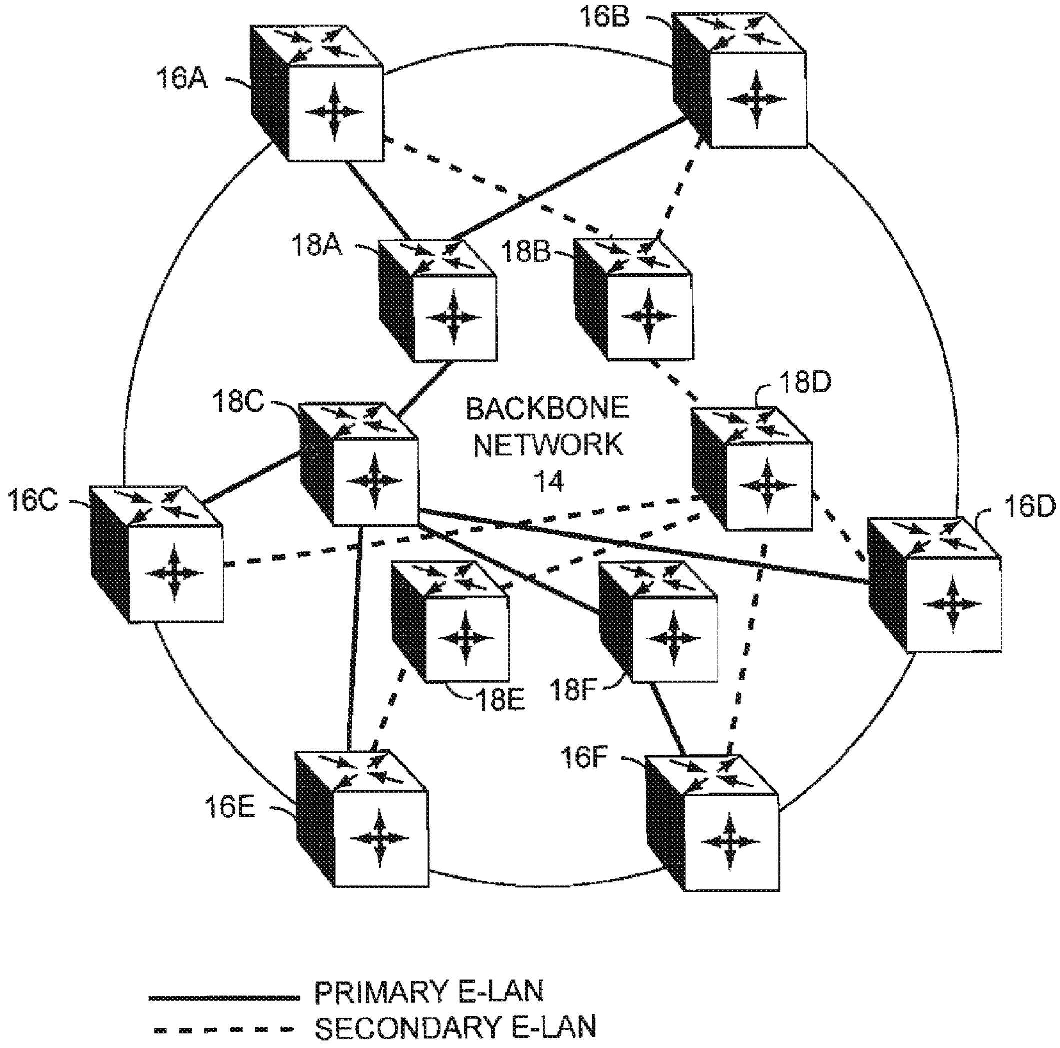

16A
16B
18A
18B
18C
18D
BACKBONE
NETWORK
14
16C
16D
18E
18F
16E
16F
PRIMARY E-LAN
SECONDARY E-LAN

FIG. 10

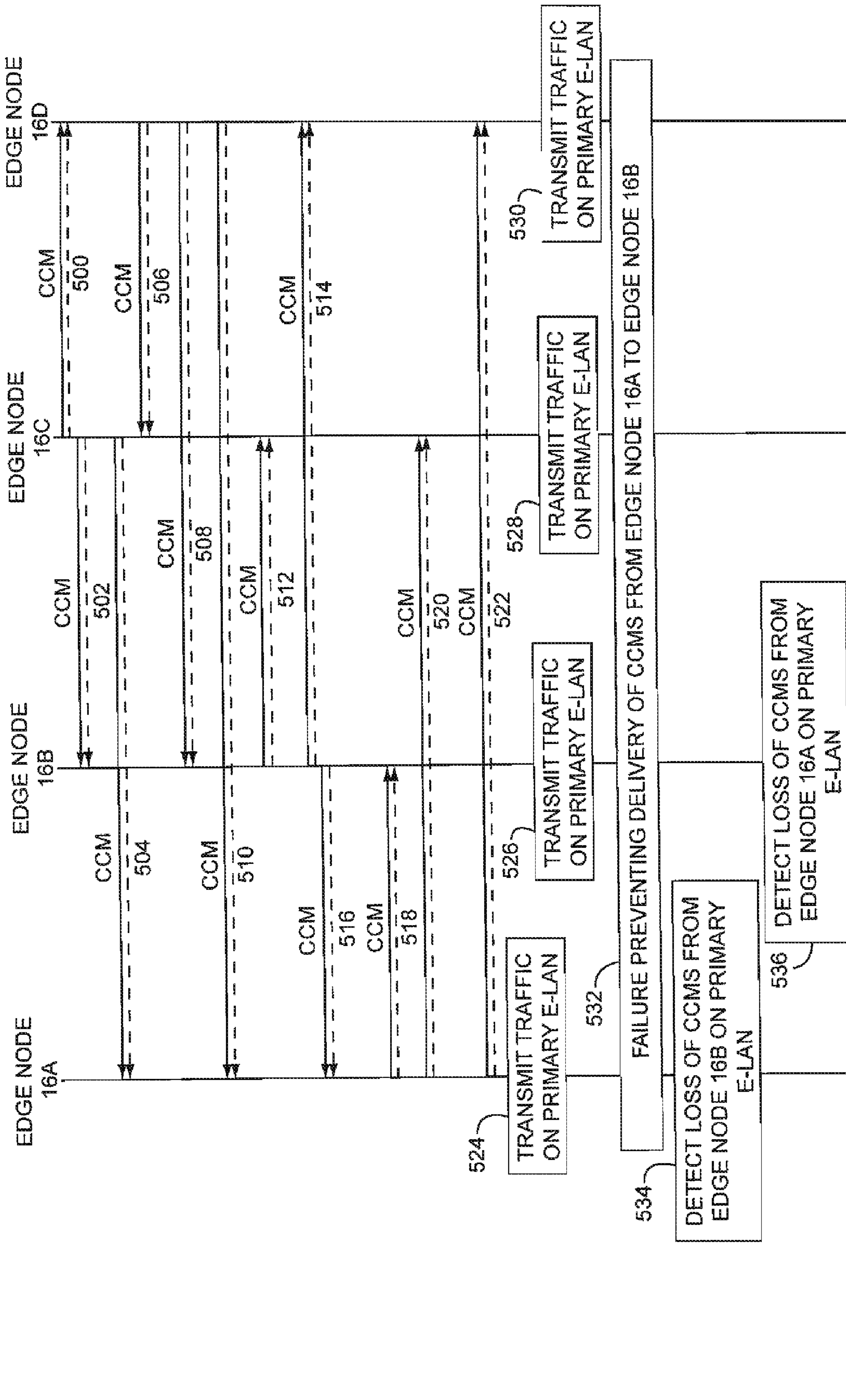

EDGE NODE 16A
EDGE NODE 16B
EDGE NODE 16C
EDGE NODE 16D
CCM
500
CCM
502
CCM
504
CCM
506
CCM
508
CCM
510
CCM
512
CCM
514
CCM
516
CCM
518
CCM
520
CCM
522
524
TRANSMIT TRAFFIC ON PRIMARY E-LAN
526
TRANSMIT TRAFFIC ON PRIMARY E-LAN
528
TRANSMIT TRAFFIC ON PRIMARY E-LAN
530
TRANSMIT TRAFFIC ON PRIMARY E-LAN
532
FAILURE PREVENTING DELIVERY OF CCMS FROM EDGE NODE 16A TO EDGE NODE 16B
534
DETECT LOSS OF CCMS FROM EDGE NODE 16B ON PRIMARY E-LAN
536
DETECT LOSS OF CCMS FROM EDGE NODE 16A ON PRIMARY E-LAN

FIG. 11A

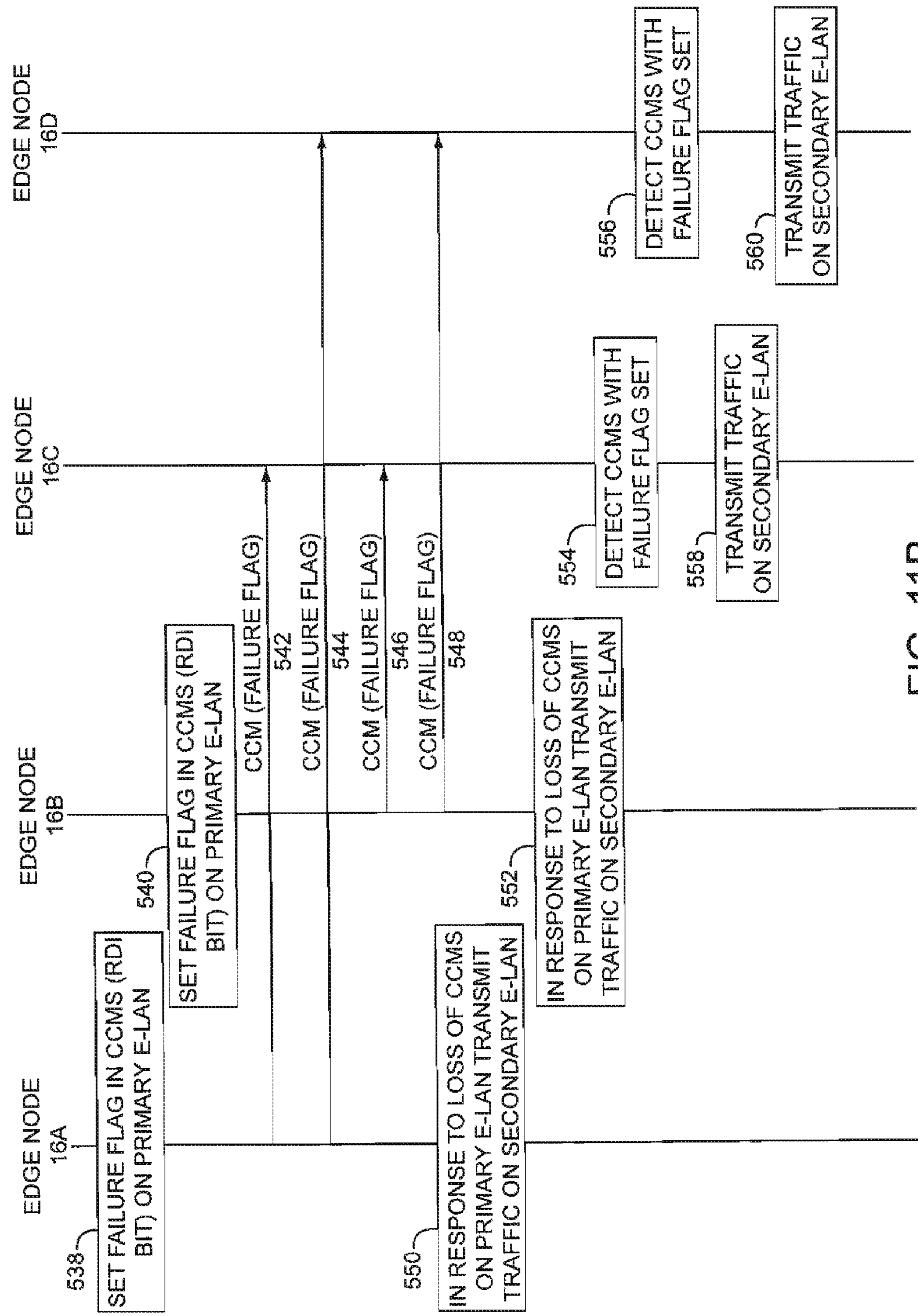

EDGE NODE 16A
EDGE NODE 16B
EDGE NODE 16C
EDGE NODE 16D
538 SET FAILURE FLAG IN CCMS (RDI BIT) ON PRIMARY E-LAN
540 SET FAILURE FLAG IN CCMS (RDI BIT) ON PRIMARY E-LAN
CCM (FAILURE FLAG) 542
CCM (FAILURE FLAG) 544
CCM (FAILURE FLAG) 546
CCM (FAILURE FLAG) 548
550 IN RESPONSE TO LOSS OF CCMS ON PRIMARY E-LAN TRANSMIT TRAFFIC ON SECONDARY E-LAN
552 IN RESPONSE TO LOSS OF CCMS ON PRIMARY E-LAN TRANSMIT TRAFFIC ON SECONDARY E-LAN
554 DETECT CCMS WITH FAILURE FLAG SET
556 DETECT CCMS WITH FAILURE FLAG SET
558 TRANSMIT TRAFFIC ON SECONDARY E-LAN
560 TRANSMIT TRAFFIC ON SECONDARY E-LAN

FIG. 11B

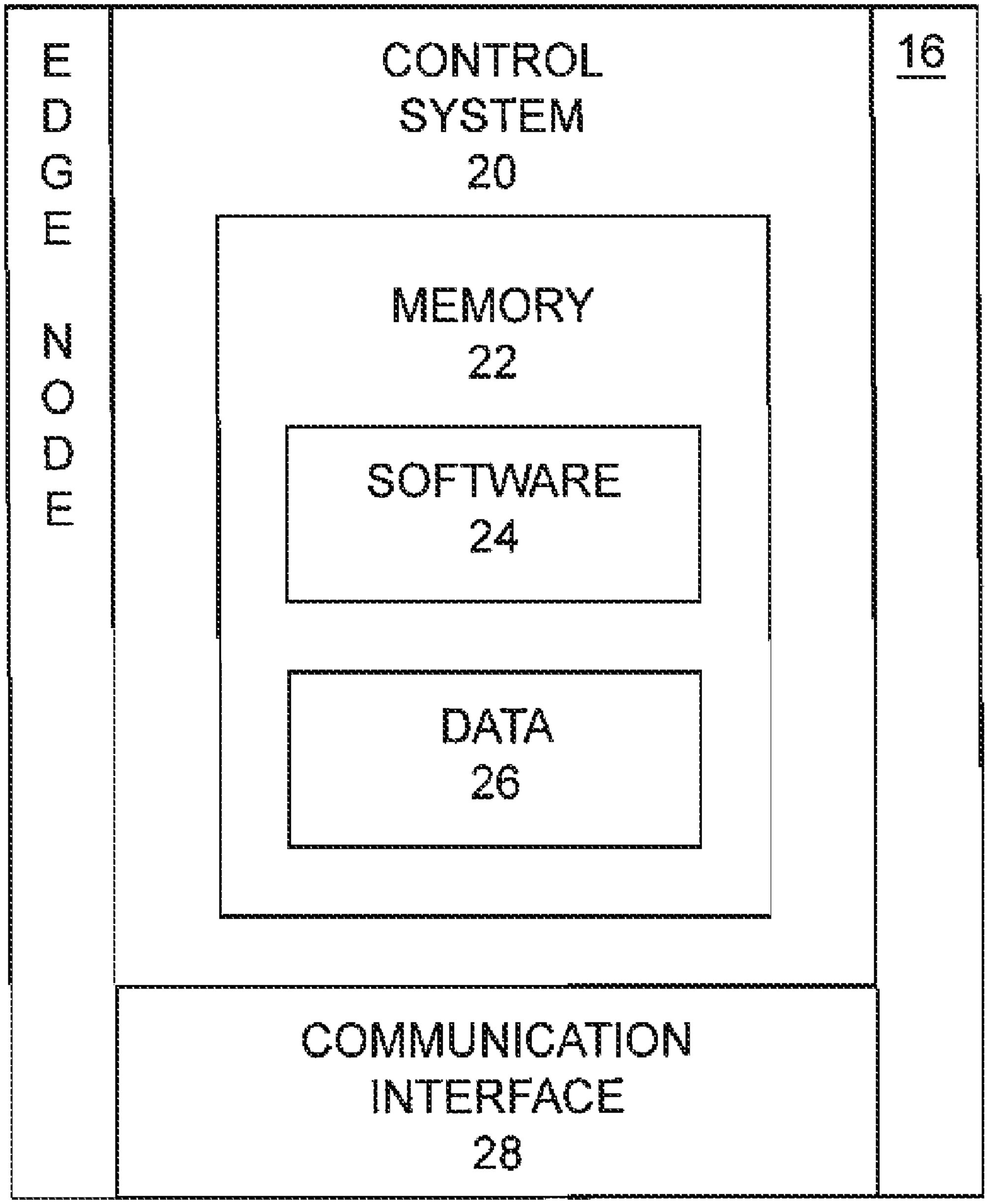

EDGE NODE
CONTROL SYSTEM 20
16
MEMORY 22
SOFTWARE 24
DATA 26
COMMUNICATION INTERFACE 28

FIG. 12

MULTI-POINT AND ROOTED MULTI-POINT PROTECTION SWITCHING

[0001] This application is a Continuation of U.S. patent application Ser. No. 12/250,266, entitled MULTI-POINT AND ROOTED MULTI-POINT PROTECTION SWITCHING, filed Oct. 13, 2008, currently pending, which claims the benefit of U.S. Provisional Patent Application Ser. No. 60/979,449, entitled PROTECTION SWITCHING FOR MULTIPOINT AND POINT-TO-MULTIPOINT, filed Oct. 12, 2007, the disclosures of which are both hereby incorporated by reference in their entireties.

FIELD OF THE INVENTION

[0002] The present invention relates to communications, and in particular to providing a substantially immediate recovery mechanism for Carrier Ethernet and like network deployments.

BACKGROUND OF THE INVENTION

[0003] Carrier Ethernet refers to the use of Ethernet frames as a transport mechanism within a backbone network, which connects any number of edge networks, such as enterprise networks, local area networks, subscriber networks, residences, and the like. The backbone networks are generally used to support wide area or metropolitan area networking between these edge networks. The rationale for using Ethernet within backbone networks is plentiful. Most local area networks (LANs), networking devices, and networked user terminals in the edge networks rely on Ethernet as a transport mechanism, which generally refers to a defined data link layer technology. As such, the use of Ethernet in the edge networks is ubiquitous and data within these edge networks is carried in Ethernet frames. By also using Ethernet frames in the backbone networks that connect the edge networks, the frames in edge and backbone networks are compatible with each other and frame conversion is avoided. If the backbone network does not employ Ethernet, the frames must be converted from one transport mechanism to another each time a boundary between an edge and backbone networks is crossed. Carrier Ethernet also supports high network access speeds, as the Ethernet-based edge networks can be coupled directly to the backbone network with relative ease. Finally, the cost associated with Ethernet-based networks is relatively low given the mature, widespread, and large scale use of Ethernet equipment. For these reasons, there is a strong desire to employ Carrier Ethernet in backbone networks that support all types of communications, including data, voice, audio, and video,

[0004] Unfortunately, Carrier Ethernet does not provide substantially immediate recovery mechanisms when a networking device or link fails. Current recovery mechanisms include rerouting and other restoration techniques, which require the affected nodes to communicate with each other extensively to identify the failure and then either reroute traffic or attempt to correct the failure. The need for the nodes to communicate with each other to such a degree to identify and address the failure injects significant delay in the recovery mechanism. Delivery sensitive services that require high quality of service levels like telephony and television services are subject to unacceptably long dropouts when substantially immediate recovery mechanisms are not available. Currently, existing recovery mechanisms being employed for Carrier Ethernet take more than a few seconds and often 30 seconds or more to recover from a failure. In contrast, most telephony and television service providers require a recovery period of less than 50 milliseconds to ensure a customer is unaffected by a failure. Other types of transport technologies, such as traditional Synchronous Optical Network (SONET) infrastructures have built-in recovery mechanisms that are capable of recovering from a failure in less than 50 milliseconds; however, these technologies are generally much more costly than Carrier Ethernet and require undesirable interworking at network entry and exit. The absence of an acceptable recovery mechanism for Carrier Ethernet is posing a major barrier to employing Carrier Ethernet for a wider and more comprehensive range of services. As such, there is a need for an effective and efficient recovery mechanism for backbone networks that employ Carrier Ethernet and like transport mechanisms.

SUMMARY OF THE INVENTION

[0005] The present invention relates to techniques for allowing one or more edge nodes in a backbone network to quickly and efficiently switch traffic delivery from a first virtual network to a second virtual network in response to a failure occurring in association with the first virtual network. In certain embodiments, an edge node is capable of independently detecting that a failure has occurred on the first virtual network and quickly transitioning from the first virtual network to the second virtual network for receiving or delivering traffic. Failures associated with the first virtual network may be detected in a variety of ways, including detecting the loss of control messages that are continuously being provided by another edge node via the first virtual network; detecting commencement of the delivery of control messages over the second virtual network by another edge node; receiving control messages that include a failure indicator over the first virtual network, and the like. Upon detecting the failure in the first virtual network, the edge node will begin delivering traffic over the second virtual network. If control messages are not already being provided over the second network, the edge node may begin providing the control messages over the second virtual network.

[0006] A source address provided in the control messages may be used by an edge node that is receiving the control messages as the destination address for traffic being sent over the backbone network. However, the delivery and processing of control messages will vary from one embodiment another. When control messages are used, they are sent successively by a given edge node at a rate that is substantially higher than one control message every 50 milliseconds, and preferably at a rate of higher than about one control message every ten milliseconds. As these rates, failure detection procedures being employed by a given edge node will allow the edge node to quickly detect the loss or presence of control messages or failure indicators and react with sufficient speed to allow the edge node to switch from delivering traffic on the first virtual network to delivering traffic on the second virtual network in less than 50 milliseconds. With protective switching occurring in less than 50 milliseconds, time-sensitive services are not unacceptably interrupted.

[0007] In one embodiment, the first virtual network and the second virtual network are Ethernet virtual connections that connect a common group of edge nodes, and the backbone network transports traffic using a Carrier Ethernet architecture; however, other architectures may take advantage of the concepts of the present invention. The first and second virtual networks and the associated edge nodes may be configured in a multi-point or rooted multi-point configuration. In a multi-point configuration, any edge node may communicate with any other edge node. Preferably, each edge node in the multi-point configuration is associated with each of the first and second virtual networks. In a rooted multi-point configuration, a hub edge node is capable of communicating with multiple spoke edge nodes; however, the spoke edge nodes may only communicate with the hub edge nodes and are not allowed to communicate with each other. As such, each of the first and second virtual networks will have different hub edge nodes, each of which serves the same set of spoke edge nodes over the respective virtual networks.

[0008] Those skilled in the art will appreciate the scope of the present invention and realize additional aspects thereof after reading the following detailed description of the preferred embodiments in association with the accompanying drawing figures.

BRIEF DESCRIPTION OF THE DRAWING FIGURES

[0009] The accompanying drawing figures incorporated in and forming a part of this specification illustrate several aspects of the invention, and together with the description serve to explain the principles of the invention.

[0010] FIG. **1** is a block diagram of a communication environment according to one embodiment of the present invention.

[0011] FIG. **2** illustrates a rooted multi-point architecture for a backbone network according to one embodiment of the present invention.

[0012] FIG. **3** illustrates the failure of a hub edge node in a rooted multi-point architecture.

[0013] FIG. **4** illustrates the failure of an intermediate node in a rooted multi-point architecture.

[0014] FIG. **5** illustrates the failure of a link in a rooted multi-point architecture.

[0015] FIG. **6** is a communication flow illustrating a first embodiment of the present invention.

[0016] FIGS. **7**A and **7**B are a communication flow illustrating a second embodiment of the present invention.

[0017] FIGS. **8**A and **8**B are a communication flow illustrating a third embodiment of the present invention.

[0018] FIGS. **9**A and **9**B are a communication flow illustrating a fourth embodiment of the present invention.

[0019] FIG. **10** illustrates a multi-point architecture for a backbone network according to one embodiment of the present invention.

[0020] FIGS. **11**A and **11**B are a communication flow illustrating a fifth embodiment of the present invention.

[0021] FIG. **12** is a block diagram of an edge node according to one embodiment of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0022] The embodiments set forth below represent the necessary information to enable those skilled in the art to practice the invention and illustrate the best mode of practicing the invention. Upon reading the following description in light of the accompanying drawing figures, those skilled in the art will understand the concepts of the invention and will recognize applications of these concepts not particularly addressed herein. It should be understood that these concepts and applications fall within the scope of the disclosure and the accompanying claims.

[0023] Prior to delving into the details of the present invention, an overview of a Carrier Ethernet environment is provided. Although Carrier Ethernet is used to illustrate the concepts of the present invention, these concepts are applicable to other transport mechanisms, such as Multi-Protocol Label Switching (MILS), which support multi-point and rooted multi-point services, which will be defined and illustrated further below. For additional information relating to Carrier Ethernet, reference is made to the IEEE 802.1ag and 862.1ah set of standards as well as the technical specifications MEF 1 through 21 of the Metro Ethernet Forum, which are incorporated herein by reference.

[0024] An exemplary Carrier Ethernet environment **10** is illustrated in FIG. **1** and includes a number of edge networks **12** that are interconnected by a backbone network **14**. The backbone network **14** employs Carrier Ethernet as a transport mechanism, and the edge networks **12** may employ traditional Ethernet as a transport mechanism; however, other transport mechanisms may be used in the edge networks **12** and interworked with the backbone network **14**. The points at the edges of the backbone network **14** may be referred to as edge nodes **16** and provide the necessary interworking and connectivity between the backbone network **14** and the edge networks **12**. An exemplary edge node **16** may include an Ethernet bridge, a backbone edge bridge, an Ethernet routing switch, and the like. Intermediate nodes **18** provide the connectivity from one edge of the backbone network **14** to another, and as such intermediate nodes **18** may be connected to each other as well as one or more edge nodes **16**. Further, edge nodes **16** may be directly connected to each other in addition to being indirectly connected to each other via one or more intermediate nodes **18**.

[0025] Assuming the edge networks **12** use an Ethernet transport mechanism, basic Ethernet frames entering the backbone network **14** from the edge networks **12** are further encapsulated into Carrier Ethernet frames by the edge nodes **16** for transport over the backbone network **14** to a destination edge node **16**. The basic Ethernet frames are extracted from the Carrier Ethernet frames by the destination edge node **16** and delivered over the edge network **12** in traditional Ethernet fashion to their intended destination or destinations.

[0026] Within the backbone network **14**, an Ethernet Virtual Connection (EVC) is used to virtually connect and associate any two or more points at the edges of the backbone network **14**. In this discussion, an EVC is constructed from a Service Instance Identifier (ISID) operating over a Backbone Virtual Local Area Network (B-VLAN) per the IEEE 802.1ah standard, which is incorporated herein by reference in its entirety, although other constructions are possible. Each EVC effectively provides a virtual network for the associated points and is isolated from other EVCs. An EVC may connect and associate two or more edge nodes **16** directly or through any number of intermediate nodes **18**. The edge nodes **16** associated with an EVC may transfer Ethernet frames to one another over the EVC while the EVC prevents the transfer of Ethernet frames to or from edge nodes **16** or intermediate nodes **18** that are not associated with the EVC. The Metro Ethernet Forum defines three types of EVCs: an E-LINE, an E-LAN, and an E-TREE. An E-LINE provides a single point-to-point connection between any two edge nodes **16** and may be used to support services such as Voice over Internet Protocol (VoIP) and Internet Access. E-LAN provides multipoint connectivity between more than two edge nodes **16** and may be used to support services such as audio and video conferencing, IPTV, On-demand media, and the like. E-TREE provides rooted multi-point connectivity between a hub edge node **16** and two or more spoke edge nodes **16** through any number of intermediate nodes **18**. The hub edge nodes **16** are allowed to multicast or broadcast information to each of its associated spoke edge nodes **16**, yet the spoke edge nodes **16** are only allowed to communicate with the associated hub edge nodes **16**. The spoke edge nodes **16** are not allowed to communicate with each other. Notably, the hub edge nodes **16** may be allowed to communicate with each other. E-TREE may be used to support broadcast television and radio services among others.

[0027] With reference to FIG. **2**, a backbone network **14** includes a primary tree and a secondary tree, according to one embodiment of the present invention. The primary tree is preferably an EVC in the form of an E-TREE, which generally extends between a first hub edge node **16**H**1** and the spoke edge nodes **16S1**, **16S2**, **16S3**, and **16S4**. The primary tree is represented by the thick, solid line that extends from the first hub edge node **16**H**1** and branches out to each of the spoke edge nodes **16S1**, **16S2**, **16S3**, and **16S4** via the intermediate nodes **18**A, **18**C, and **18**F. The primary tree also includes a branch connected to a second hub edge node **16**H**2**. Similarly, the secondary tree is preferably an EVC in the form of an E-TREE, which generally extends between a second hub edge node **16**H**2** and the spoke edge nodes **16S1**, **16S2**, **16S3**, and **16S4**. The secondary tree is represented by the thick, dashed line that extends from hub edge node **16**H**2** and branches out to each of the spoke edge nodes **16S1**, **16S2**, **16S3**, and **16S4** via the intermediate nodes **18**B, **18**D, and **18**E. The secondary tree also includes a branch connected to first hub edge node **16**H**1**. As such, the primary and second trees serve the same spoke edge nodes **16S1**, **16S2**, **16S3**, and **16S4** from the different first and second hub edge nodes **16**H**1** and **16**H**2**.

[0028] The first and second hub edge nodes **16**H**1** and **16**H**2** are coupled to the same edge network **12** and are capable of receiving traffic from a given source and delivering the traffic over the respective primary and secondary trees to the one or more of the spoke edge nodes **16S1**, **16S2**, **16S3**, and **16S4**. Further, any spoke edge nodes **16S1**, **16S2**, **16S3**, and **16S4** are capable of delivering traffic over the respective primary and secondary trees to the first and second hub edge nodes **16**H**1** and **16**H**2**. In addition to delivering traffic, the nodes associated with the primary or secondary trees may exchange control messages over the trees, such as Continuity Check Messages (CCMs), which may be used to determine whether a failure has occurred within the primary and secondary trees, in this embodiment, control messages and traffic being sent to the hub edge nodes **16**H**1** and **16**H**2** from a spoke edge node **16S1**, **16S2**, **16S3**, **16S4** is delivered using an OD, which is referred to as a spoke ISID. Control messages and traffic being sent to the spoke edge nodes **16S1**, **16S2**, **16S3**, **16S4** from a hub edge node **16**H**1**, **16**H**2** is delivered using a second ISID, which is referred to as a hub ISID and is different than the spoke ISID. The Ethernet frames carrying the control messages or traffic will have source and destination addresses associated with the edge nodes **16** and will be tagged according to the tree, primary or secondary, being used for transport.

[0029] The primary and secondary trees may be established dynamically or statically through manual provisioning or an appropriate control protocol, such as some version of Spanning Tree Protocol (xSTP) or Provider Link State Bridging (PLSB). When established, steps may be taken to ensure that the primary and secondary trees are not part of a shared risk link group (SRLG), and as such, a single link or nodal failure will not affect both the primary and secondary trees. The goal is to ensure that a path exists between at least one of the hub edge nodes **16**H**1**, **16**H**2** and each the spoke edge nodes **16S1**, **16S2**, **16S3**, **16S4** over either the primary tree or the secondary tree, even when there is a failure of a link or node in the backbone network **14**. When configured in this manner, the primary and secondary trees are considered independent, even though they serve common spoke edge nodes **16S1**, **16S2**, **16S3**, and **16S4** from different hub edge nodes **16**H**1** and **16**H**2**.

[0030] FIGS. **3**, **4**, and **5** illustrate different types of failures that can occur in the backbone network **14** and affect a primary or secondary tree. Each of the three different failures that are illustrated affect the primary tree and do not affect the secondary tree. As such, these failures will prevent the exchange of traffic and control messaging between the first hub edge node **16**H**1** and one or more of the spoke edge nodes **16S1**, **16S2**, **16S3**, **16S4**, With reference to FIG. **3**, the first hub edge node **16**H**1** fails, and as such, the first hub edge node **16**H**1** is not available for receiving or delivering control messages to the spoke edge nodes **16S1**, **16S2**, **16S3**, **16S4**, or to the second hub edge node **16**H**2**, Further, the first hub edge node **16**H**1** will not be able to exchange traffic between the associated edge network **12** (not shown) and the backbone network **14**, along with the other nodes of the backbone network **14**. Even though the first hub edge node **16**H**1** has failed, traffic and control messaging may be exchanged between the second hub edge node **16**H**2** and the spoke edge nodes **16S1**, **16S2**, **16S3**, **16S4** over the secondary tree. As will be described further below, upon detecting such a failure the present invention provides techniques for quickly switching over to the secondary tree from the primary tree when such a failure occurs, according to one embodiment of the present invention.

[0031] With reference to FIG. **4**, assume intermediate node **18**C fails. As illustrated, intermediate node **18**C is a critical node in the primary tree, and when it fails, traffic and control messaging cannot be exchanged between the first hub edge node **16**H**1** and the spoke edge nodes **16S1**, **16S2**, **16S3**, **16S4** over the primary tree. However, traffic and control messaging may be exchanged between the second hub edge node **16**H**2** and the spoke edge nodes **16S1**, **16S2**, **16S3**, **16S4** over the secondary tree.

[0032] With reference to FIG. **5**, a link failure is illustrated, The illustrated link failure involves the physical link between intermediate node **18**C and the spoke edge node **16S2**. With such a failure, traffic and control messaging cannot be exchanged between the first hub edge node **16**H**1** and the second spoke edge node **16S2** using the primary tree. However, traffic and control messaging may be exchanged between the first hub edge node **16**H**1** and the first, third, and fourth spoke edge nodes **16S1**, **16S3**, **16S4** using the primary tree. Using the secondary tree, traffic and control messaging may be exchanged between the second hub edge node **16**H**2** and the second spoke edge node **16S2**, as well as the first, third, and fourth spoke edge nodes **16S1**, **16S3**, **16S4**. In each of these examples, since the first and second hub edge nodes **16**H**1** and **16**H**2** handle the same traffic for the same spoke edge nodes **16S1**, **16S2**, **16S3**, **16S4**, a failure associated with one tree can be compensated for by employing another tree.

[0033] Although the examples provided herein only illustrate two hub edge nodes **16**H**1** and **16**H**2** cooperating with a given subset of spoke edge nodes **16S1**, **16S2**, **16S3**, and **16S4**, any number of hub edge nodes **16** may be associated with such a subset.

[0034] In the following examples, these different types of failures may be handled in different ways using different protection mechanisms. With a first embodiment, failure protection is provided for an E-TREE configuration with primary and secondary trees configured as described above. In this embodiment, assume that traffic is simultaneously broadcast or unicast from both of the first and second hub edge nodes **16**H**1** and **16**H**2** over the primary and secondary trees to the appropriate spoke edge nodes **16S1**, **16S2**, **16S3**, **16S4**. Notably, in the following communication flows, only spoke edge nodes **1631** and **1632** are illustrated for conciseness and clarity. These spoke edge nodes **1631** and **1632** will only deliver traffic, preferably in a unicast form, to one of the hub edge nodes **16**H**1** and **16**H**2** over a selected one of the primary and secondary trees. Accordingly, a given spoke edge node **16S1**, **16S2** will only transmit traffic over one tree at a time. The primary tree may be identified or selected through an appropriate provisioning process. Again, the first hub edge node **16**H**1** is associated with the primary tree and the second hub edge node **16**H**2** is associated with the secondary tree. In this embodiment, the spoke edge nodes **16S1** and **16S2** will select an appropriate one of the primary and secondary trees to use at any given time. When there is no failure associated with the primary tree, the primary tree will be used. When a failure is detected on the primary tree, the spoke edge nodes **16S1**, **16S2** will independently switch to using the secondary tree for delivery of traffic to the second hub edge node **16**H**2** instead of delivering traffic to the first hub edge node **16**H**1** over the primary tree.

[0035] Failures are detected by monitoring control messages, such as CCMs, which are sent to each of the spoke edge nodes **16S1**, **16S2** by both of the hub edge nodes **16**H**1**, **16**H**2**. The first hub edge node **16**H**1** will send its CCMs to the first and second spoke edge nodes **16S1**, **16S2** over the primary tree, while the second hub edge node **16**H**2** will send the CCMs to the first and second spoke edge nodes **16S1**, **16S2** over the secondary tree. To ensure switching can take place in less than 50 ms, the first and second hub edge nodes **16**H**1** and **16**H**2** will send the CCMs at a rate equal or greater to one every 10 ms.

[0036] The CCMs will have a source address (SA) and a destination address (DA). The source address will correspond to the first or second hub edge node **16**H**1**, **16**H**2** from which the CCM was originated. The spoke edge nodes **16S1**, **16S2** will receive the CCMs from the different hub edge nodes **16**H**1**, **16**H**2** over the different trees. Assuming CCMs are being received over the selected or primary tree, the spoke edge node **16S1**, **16S2** will continue to use the primary tree for delivering traffic to the first edge node **16**H**1**. The destination address for the Ethernet frames carrying the traffic is preferably set to the source address, which was provided in the CCMs received over the primary tree. When a spoke edge node **16S1**, **16S2** stops receiving CCMs from the first hub edge node **16**H**1**, the spoke edge node **16S1**, **16S2** will detect a failure on the primary tree and switch over to the secondary tree. This switching will entail changing the destination address for the Ethernet frames carrying the traffic to an address associated with the second hub edge node **16**H**2**. This address is preferably obtained from the CCMs that are received from the second hub edge node **16**H**2** over the secondary tree. To complete the switch to the secondary tree, the spoke edge node **16S1**, **16S2** will tag the Ethernet frames that are carrying traffic to the second hub edge node **16**H**2** with a tag corresponding to the secondary tree and deliver the traffic toward the second hub edge node **16**H**2** via the secondary tree. Notably, the traffic that was originally being sent to the first hub edge node **16**H**1** before the failure occurred was tagged with a tag associated with the primary tree.

[0037] Since the spoke edge nodes **16S1**, **16S2** can independently and quickly detect a failure on a primary or selected tree, a switch from the primary tree to the secondary tree may occur well within 50 ms to avoid unacceptable service interruptions. Further, when CCMs begin to arrive over the primary tree from the first hub edge node **16**H**1**, the spoke edge nodes **16S1**, **16S2** may determine that the failure has been resolved, and effect a switch from the secondary tree back to the primary tree, wherein the Ethernet frames are sent to the first hub edge node **16**H**1** using an address associated with the first hub edge node **16**H**1** and tagged to identify the primary tree. Again, the address for the first hub edge node **16**H**1** may be derived from the CCMs provided by the first hub edge node **16**H**1**. This is just one example of a possible reversion policy; other policies are possible, such as holding the current state for a certain period of time before reverting or requiring manual reversion commands or instructions.

[0038] With reference to FIG. **6**, a communication flow is provided to illustrate the switchover process outlined above. Initially, assume the first hub edge node **16**H**1** is delivering CCMs to the hub edge node **16**H**2**, the first spoke edge node **16S1**, and the second spoke edge node **16S2** every 10 ms over the primary tree (steps **100**, **102**, and **104**). The source address (SA**1**) for the first hub edge node **16**H**1** is included in the CCMs. The second hub edge node **16**H**2** is delivering CCMs every 10 ms to the first hub edge node **16**H**1**, the first spoke edge node **16S1**, and the second spoke edge node **16S2** over the secondary tree (steps **106**, **108**, and **110**). The CCMs will have a source address (SA**2**) corresponding to the address of the second hub edge node **16**H**2**. The first spoke edge node **16S1** will receive the CCMs from the first hub edge node **16**H**1** over the primary tree, and will set the unicast destination address (DA) for the hub traffic based on the source address (SA**1**) provided in the CCMs (step **112**). The hub traffic is then sent to the first hub edge node **16**H**1** using the address (SA**1**) that is associated with the first hub edge node **16**H**1** (step **114**). Various mechanisms such as configuration or dynamic selection could be used to assign spokes to the primary.

[0039] The second spoke edge node **16S2** will receive the CCMs from the first hub edge node **16**H**1** over the primary tree, and will set the unicast destination address (DA) for the hub traffic based on the source address (SA**1**) provided in the CCMs (step **116**). The hub traffic is then sent to the first hub edge node **16**H**1** using the address (SA**1**) that is associated with the first hub edge node **16**H**1** (step **118**). Again, the traffic will be delivered in Ethernet frames and will be tagged based on the primary or secondary tree over which the traffic is delivered.

[0040] At this point, assume there is a failure preventing the CCMs from the first hub edge node **16**H**1** from reaching the first and second spoke edge nodes **16S1**, **16S2** (step **120**). When the failure occurs, the CCMs provided by the first hub edge node **16**H**1** over the primary tree will not be received by the first and second spoke edge nodes **16S1**, **16S2**. As such, each of the first and second spoke edge nodes **16S1**, **16S2** will detect a loss of CCMs from the first hub edge node **16**H**1** and will set the unicast destination address for the hub traffic based on the source address in the CCMs received from the second hub edge node **16**H**2** via the secondary tree (steps **122** and **124**). The source address in the CCMs received from the second hub edge node **16**H**2** is SA**2**. Accordingly, the first and second spoke edge nodes **16S1**, **16S2** will substantially immediately begin sending hub traffic over the secondary tree to the second hub edge node **16**H**2** using the address (SA**2**) of the second hub edge node **16**H**2** (steps **126** and **128**). Again, the Ethernet frames being transported over the various trees may be tagged accordingly, such that the Ethernet frames will be delivered over the appropriate tree. Further, when the first and second spoke edge nodes **16S1**, **16S2** begin receiving CCMs via the primary tree from the first hub edge node **16**H**1**, they may switch back to using the primary tree for delivering traffic to the first hub edge node **16**H**1**.

[0041] In a second embodiment, a recovery mechanism is provided wherein only one of the primary or secondary trees is primarily used at any given time. Preferably, the secondary tree is only used when there is a failure associated with the primary tree. Accordingly, assuming the first hub edge node **16**H**1** is the active node when there is no failure associated with the primary tree, the first hub edge node **16**H**1** will transmit CCMs over the primary tree; however, the inactive or second hub edge node **16**H**2** will not transmit CCMs over the secondary tree until it detects a failure on the primary tree. In this example, the second hub edge node **16**H**2** will detect a failure on the primary tree when it stops receiving CCMs from the first hub edge node **16**H**1** via the primary tree. The first and second spoke edge nodes **16S1**, **16S2** will only receive CCMs from the first hub edge node **16**H**1** over the primary tree until there is a failure, and after the failure, the first and second spoke edge nodes **16S1**, **16S2** may begin receiving CCMs from the second hub edge node **16**H**2** over the secondary tree.

[0042] The first and second spoke edge nodes **16S1**, **16S2** will select a tree and determine the hub edge node **16**H**1** or **16**H**2** to use for delivering traffic based on receiving the CCMs. When CCMs are received from the first hub edge node **16**H**1**, traffic will be sent to the first hub edge node **16**H**1** over the primary tree. When CCMs are being received from the second hub edge node **16**H**2** over the secondary tree, the first and second spoke edge nodes **16S1**, **16S2** will send traffic to the second hub edge node **16**H**2** over the secondary tree. Accordingly, the first and second spoke edge nodes **16S1**, **16S2** are operating substantially independently, and are determining how to transmit traffic based on the CCMs being received. Similarly, the first and second hub edge nodes **16**H**1** and **16**H**2** are also operating independently based on the delivery of CCMs. When the CCMs are being delivered at a rate substantially higher than one CCM every 50 ms, traffic delivery can be switched from the primary tree to the secondary tree within 50 ms and avoid unacceptable service disruptions.

[0043] A communication flow illustrating this embodiment is provided in FIGS. **7**A and **7**B. Initially, the first hub edge node **16**H**1** will send CCMs over the primary tree to the second hub edge node **16**H**2** and the first and second spoke edge nodes **16S1**, **16S2** (steps **200**, **202**, and **204**). The CCMs will include the address of the first hub edge node **16**H**1** as the source address (SA**1**). Assuming there is no detected failure in the primary tree, the secondary tree is not used, and as such, the second hub edge node **16**H**2** will not send CCMs to the first hub edge node **16**H**1**, the first spoke edge node **16S1**, or the second spoke edge node **16S2**. The first hub edge node **16**H**1** will send spoke traffic toward the first spoke edge node **16S1** and the second spoke edge node **16S2** (steps **206** and **208**). The second hub edge node **16**H**2** will not be providing traffic to the first and second spoke edge nodes **16S1**, **16S2**.

[0044] During this time, the first and second spoke edge nodes **16S1**, **16S2** will receive the CCMs from the first hub edge node **16**H**1** and set the unicast destination address for hub traffic based on the source address (SA**1**) provided in the CCMs that were received from the first hub edge node **16**H**1** via the primary tree (steps **210** and **212**). Accordingly, the first and second spoke edge nodes **16S1**, **16S2** will deliver hub traffic to the first hub edge node **16**H**1** via the primary tree (steps **214** and **216**). The destination address in the Ethernet frames of the hub traffic is set to the address (SA**1**) associated with the first hub edge node **16**H**1** and derived from the CCMs received from the first hub edge node **16**H**1**.

[0045] At this point, assume there is a failure of hub edge node **16**H**1** (step **218**). When the first hub edge node **16**H**1** fails, the second hub edge node **16**H**2** will detect a loss of the CCMs from the first hub edge node **16**H**2** (step **220**) and immediately initiate delivery of CCMs and spoke traffic via the secondary tree (step **222**). As such, the second hub edge node **16**H**2** will deliver CCMs to the first hub edge node **16**H**1**, the first spoke edge node **16S1**, and the second spoke edge node **16S2** (steps **224**, **226**, and **228**). The CCMs will have a source address (SA**2**) corresponding to the second hub edge node **16**H**2**. As illustrated, the second hub edge node **16**H**2** will send spoke traffic to the first and second spoke edge nodes **16S1**, **16S2**, respectively, over the secondary tree (steps **230** and **232**).

[0046] The first and second spoke edge nodes **16S1**, **16S2** may detect the failure to receive CCMs via the primary tree, or may detect the receipt of CCMs from the second hub edge node **16**H**2** via the secondary tree. Upon such detection, the first and second spoke edge nodes **16S1**, **16S2** will set the unicast destination address for the hub traffic based on the source address (SA**2**) provided in the CCMs received from the second hub edge node **16**H**2** (steps **234** and **236**). Accordingly, the first and second spoke edge nodes **16S1**, **16S2** will deliver hub traffic to the second hub edge node **16**H**2**, wherein the destination address of the Ethernet frames carrying the hub traffic is set to the address (SA**2**) associated with the second hub edge node **16**H**2** (steps **238** and **240**). The second hub edge node **16**H**2** as well as the first and second spoke edge nodes **16S1**, **16S2** may switch back to the primary tree once CCMs from the first hub edge node **16**H**1** reappear on the primary tree.

[0047] In a third embodiment of the present invention, once again assume that the secondary tree is only used when the primary tree is associated with a failure. While the previous example was particularly beneficial when one of the first and second hub edge nodes **16**H**1**, **16**H**2** failed, this embodiment is particularly beneficial when the failure occurs in an intermediate node **18** or a physical link in the primary or secondary trees. As with the previous embodiment, assume that the first hub edge node **16**H**1** is active and there are no failures detected on the primary tree. In operation, the first hub edge node **16**H**1** will transmit control messages including the CCMs, as well as traffic over the primary tree, and the first and second spoke edge nodes **16S1**, **16S2** transmit their traffic over the primary tree as well. Notably, the first and second spoke edge nodes **16S1**, **16S2** will transmit CCMs, preferably in a unicast format, toward the active hub, which is the first hub edge node **16**H**1**.

[0048] When a failure occurs, the first hub edge node **16H1** will stop receiving CCMs from one or more of the first and second spoke edge nodes **16S1**, **16S2**. In response to detecting the loss of CCMs from the first or second spoke edge nodes **16S1**, **16S2**, the first hub edge node **16**H**1** will begin transmitting CCMs that include a failure flag. In one embodiment, the failure flag corresponds to setting the Remote Defect Indication (RDI) bit of a CCM. Upon receipt of a CCM that includes the failure flag, the second hub edge node **16H2** will start delivering traffic and CCMs over the secondary tree toward the first and second spoke edge nodes **16S1**, **16S2**. The first and second spoke edge nodes **16S1**, **16S2** will continue to use the primary tree as long as CCMs are received over the primary tree from the (active) first hub edge node **16H1**. When CCMs are no longer being received via the primary tree, the first and second spoke edge nodes **16S1**, **16S2** will immediately transition to using the secondary tree, wherein traffic and the CCMs are delivered to the second hub edge node **16H2** via the secondary tree. When the failure is corrected, the affected ones of the first hub edge node **16**H**1**, the first spoke edge node **16S1**, and the second spoke edge node **16S2** will transition back to the original operating state.

[0049] A communication flow illustrating this example is provided in FIGS. **8**A and **8**B. Initially, assume that the first hub edge node **16H1** is the active node and is delivering CCMs over the primary tree to the second hub edge node **16**H**2** as well as the first and second spoke edge nodes **16S1**, **16S2** (steps **300**, **302**, and **304**). The first hub edge node **16H1** is also delivering spoke traffic over the primary tree to the first and second spoke edge nodes **16S1**, **16S2** (steps **306** and **308**). While CCMs are being received via the primary tree from the first hub edge node **16**H**1**, the first and second spoke edge nodes **16S1**, **16S2** will deliver CCMs to the first hub edge node **16**H**1** (steps **310** and **312**), as well as deliver hub traffic to the first hub edge node **16**H**1** via the primary tree (steps **314** and **316**). Notably, the CCMs are preferably delivered every 10 ms, and the hub traffic will have a destination address corresponding to the source address (SA**1**) that is provided in the CCMs received from the first hub edge node **16**H**1** and associated with the first hub edge node **16**H**1**.

[0050] At this point, assume a failure occurs that isolates the spoke edge node **16S1** from the hub edge node **16**H**1** (step **318**). As such, the first hub edge node **16H1** will detect the loss of CCMs from the first spoke edge node **16S1** via the primary tree (step **320**). The first hub edge node **16**H**1** will then set a failure flag in the CCMs (step **322**) that it is consistently delivering toward the second hub edge node **16**H**2**, the first spoke edge node **16S1**, and the second spoke edge node **16S2** (steps **324**, **326**, and **328**). Again, the failure flag may be represented by setting the RDI bit in the CCM message. Notably, the CCMs are only received by the second spoke edge node **16S2** and the second hub edge node **16**H**2**. The CCMs sent toward the first spoke edge node **16S1** are not received.

[0051] The second hub edge node **16**H**2** will detect the failure flag in the CCMs received via the primary tree from the first hub edge node **16**H**1** (step **330**). In response to detecting the failure flag in the CCMs, the second hub edge node **16**H**2** will begin sending CCMs via the secondary tree to the first hub edge node **16**H**1**, as well as to the first and second spoke edge nodes **16S1**, **16S2** (steps **332**, **334**, and **336**). The CCMs will include the address (SA**2**) associated with the second hub edge node **16H2** as the source address. Further, the second hub edge node **16H2** will initiate delivery of spoke traffic toward the first and second spoke edge nodes **16S1**, **16S2** via the secondary tree (steps **338** and **340**).

[0052] During this time, the second spoke edge node **16S2** will continue receiving CCMs from the first hub edge node **16**H**1** via the primary tree, and as such, will continue operation as normal. In particular, the second spoke edge node **16S2** will continue sending hub traffic and CCMs to the first hub edge node **16**H**1** on the primary tree (steps **342**, **344**, and **346**). However, the first spoke edge node **16S1** will detect the loss of CCMs that were being provided by the first hub edge node **16**H**1** over the primary tree (step **348**). Further, the first spoke edge node **16S1** will begin receiving CCMs from the second hub edge node **16**H**2** and set the unicast destination address for hub traffic based on the source address in the CCMs received via the secondary tree (step **350**). In this instance, the source address in the CCMs from the secondary tree (SA**2**) corresponds to the second hub edge node **16**H**2**. Accordingly, the first spoke edge node **16S1** will begin sending CCMs and hub traffic toward the second hub edge node **16**H**2** via the secondary tree (steps **352** and **354**). Notably, if the second spoke edge node **16S2** is affected by the failure in a similar way as the first spoke edge node **16S1**, the second spoke edge node **16S2** may also switch to the secondary tree in a similar fashion. Further, when the failure is resolved, the affected ones of the first and second hub edge nodes **16**H**1**, **16**H**2** as well as the first and second spoke edge nodes **16S1**, **16S2** may revert back to using the primary tree and return to the original operating state.

[0053] In yet another embodiment employing redundant E-TREEs, different E-TREEs may be used to share or coordinate loads for different spoke edge nodes **16S1**, **16S2**. For example, a non-failure state may call for traffic involving the first spoke edge node **16S1** being carried over the primary tree, while traffic associated with the second spoke edge node **16S2** is carried by the secondary tree. The CCMs or other control messages are provided by each of the first and second hub edge nodes **16**H**1**, **16**H**2** to each of the first and second spoke edge nodes **16S1**, **16S2** over the corresponding primary and secondary trees. As such, the first and second spoke edge nodes **16S1**, **16S2** receive CCMs from each of the first and second hub edge nodes **16**H**1**, **16**H**2**, over the primary and secondary trees, respectively, even though traffic is only being exchanged over a corresponding one of the primary and secondary trees. Although this example contemplates a distinct allocation of traffic for different spoke edge nodes **16S1**, **16S2** to different ones of the primary and secondary trees, the traffic may be shared in various ways based on traffic type, quality of service, or the like, wherein certain traffic is associated with the primary tree and other traffic is associated with the secondary tree for a given spoke edge node **16S1**, **16S2**. The spoke edge nodes **16S1**, **16S2** will again use the source address provided in the CCMs received over the allocated tree, unless a failure occurs. The spoke edge nodes **16S1**, **16S2** will respectively send CCMs toward the associated one of the first and second hub edge nodes **16**H**1**, **16**H**2**. The first and second spoke edge nodes **16S1**, **16S2**, in this embodiment, do not send CCMs over both the primary and secondary trees.

[0054] Upon a failure in receiving CCMs from an active one of the first and second hub edge nodes **16**H**1**, **16**H**2**, the corresponding first and second spoke edge nodes **16S1**, **16S2** will switch to the other one of the first and second hub edge nodes **16**H**1**, **16**H**2** and begin transmitting CCMs toward the newly selected hub edge node **16**H**1** or **16**H**2**. Upon failing to receive CCMs from the corresponding spoke edge node **16S1** or **16S2**, the associated hub edge node **16**H**1** or **16**H**2** will stop serving the associated spoke edge node **16S1** or **16S2**. When a hub edge node **16**H**1** or **16**H**2** begins receiving CCMs from a spoke edge node **16S1**, **16S2** that it was not previously serving, it will begin serving that spoke edge node **16S1**, **16S2**.

[0055] A communication flow illustrating this embodiment is provided in FIGS. **9**A and **9**B. Initially, assume that first hub edge node **16**H**1** is assigned to spoke edge node **16S1**, and traffic is provided over the primary tree. Further assume that second hub edge node **16**H**2** is assigned to spoke edge node **16S2**, and traffic is provided over the secondary tree. As such, the first hub edge node **16**H**1** will begin sending CCMs via the primary tree to the second hub edge node **16**H**2** as well as the first and second spoke edge nodes **16S1**, **16S2** (steps **400**, **402**, and **404**). The CCMs provided by the first hub edge node **16**H**1** will include a source address (SA**1**) that is associated with the first hub edge node **16**H**1**. The first hub edge node **16**H**1** is also providing spoke traffic for the first spoke edge node **16S1** to the first spoke edge node **16S1** (step **406**). In response to receiving the CCMs via the primary tree, the first spoke edge node **16S1** will provide hub traffic and CCMs toward the first hub edge node **16**H**1** (steps **408** and **410**).

[0056] Concurrently, the second hub edge node **16**H**2** is sending CCMs via the secondary tree to the first hub edge node **16**H**1** as well as the first and second spoke edge nodes **16S1**, **16S2** (steps **412**, **414**, and **416**). The CCMs provided by the second hub edge node **16**H**2** will include a source address (SA**2**) that is associated with the second hub edge node **16**H**2**. The second hub edge node **16**H**2** will deliver spoke traffic for the second spoke edge node **16S2** via the secondary tree (step **418**). Assuming there is no failure in the secondary tree, the second spoke edge node **16S2** will deliver hub traffic and CCMs via the secondary tree toward the second hub edge node **16**H**2** (steps **420** and **422**). The hub traffic is directed to the address associated with the second hub edge node **16**H**2** and provided in the CCMs received from the second hub edge node **16**H**2**.

[0057] Next, assume that a failure occurs that isolates the first spoke edge node **16S1** from the first hub edge node **16**H**1** (step **424**). As such, the traffic being exchanged between the first spoke edge node **16S1** and the first hub edge node **16**H**1** via the primary tree is affected. The first spoke edge node **16S1** will detect the loss of CCMs from the first hub edge node **16**H**1** via the primary tree (step **426**), and will begin processing the CCMs being received from the second hub edge node **16**H**2** via the secondary tree (step **428**). In response, the first spoke edge node **16S1** will set the unicast destination address for the hub traffic based on the source address (SA**2**) in the CCMs being received from the second hub edge node **16**H**2** over the secondary tree (step **430**). As such, the first spoke edge node **16S1** switches from sending hub traffic to the first hub edge node **16**H**1** via the primary tree to sending hub traffic to the second hub edge node **16**H**2** via the secondary tree using the address associated with the second hub edge node **16**H**2** (step **432**).

[0058] As a result of the failure, the first hub edge node **16**H**1** will also detect a loss of CCMs being provided by the first spoke edge node **16S1** via the primary tree, and will stop delivery of spoke traffic toward the spoke edge node **16S1** (step **434**). Meanwhile, the first spoke edge node **16S1** will switch from sending CCMs to the first hub edge node **16**H**1** via the primary tree to sending CCMs to the second hub edge node **16**H**2** via the secondary tree (step **436**). Upon detecting the CCMs of the first spoke edge node **16S1** via the secondary tree, the second hub edge node **16**H**2** will initiate delivery of spoke traffic that is intended for the first spoke edge node **16S1** to the spoke edge node **16S1** via the secondary tree (steps **438** and **440**). As such, the second hub edge node **16**H**2** will begin receiving and delivering traffic associated with the first spoke edge node **16S1** when the first hub edge node **16**H**1** is no longer capable of doing so due to a failure. As with the other embodiments, when the failure condition is resolved, normal operation may be resumed. Assuming that the CCMs are delivered at a rate higher than one every 10 ms, switches from a primary tree to a secondary tree may take place in less than 50 ms, and therefore, prevent unacceptable interruptions in service.

[0059] While the prior examples related to the use of primary and secondary trees, such as E-TREEs, certain concepts of the present invention are applicable to E-LAN configurations as well. With reference to FIG. **10**, an exemplary E-LAN configuration is depicted. As stated above, an ELAN configuration is a multi-point configuration wherein any edge node **16**A-**16**F is capable of communicating with any other edge node **16**A-**16**F. The multi-point architecture is not rooted at any particular hubs, and each edge node **16**A-**16**F may be an equivalent node and be used to support different sources, such as edge networks **12**. As illustrated, each of the edge nodes **16**A-**16**F is connected through a primary E-LAN, which is illustrated by a thick solid line, as well as a secondary E-LAN, which is illustrated by a thick dashed line. A failure in one E-LAN will preferably not affect the other E-LAN, and as such, operation in a primary ELAN may be switched to a secondary E-LAN upon detecting a failure in the primary E-LAN. As depicted, intermediate nodes **18**A, **18**C, and **18**F are interconnected and facilitate connections to each of the edge nodes **16**A-**16**F to form the primary ELAN. Similarly, intermediate nodes **18**B, **18**D, and **18**E are interconnected and facilitate connections to each of the edge nodes **16**A-**16**F to provide the secondary ELAN. Again, the primary and secondary E-LANs are considered separate EVCs, and as such Ethernet frames may be tagged to facilitate transport over a particular E-LAN. The E-LANs may be dynamically or statically provisioned in the same fashion as described above. Preferably, the primary and secondary E-LANs are substantially independent from one another, such that a failure on one does not affect the other, and shared risk link groups are avoided. Although the avoidance of shared risk link groups is preferred, it is not necessary in the ELAN or E-TREE examples.

[0060] In one embodiment, each of the edge nodes **16**A-**16**D are providing CCMs to other ones of the edge nodes **16**A-**16**D on both the primary and secondary E-LANs. When a nodal or link failure occurs, one or more edge nodes **16**A-**16**F will detect a loss of CCMs from other edge nodes **16**A-**16**F, The edge node **16**A-**16**F that detects a loss of CCMs may set a failure flag in its CCMs and deliver the CCMs to the other edge nodes **16**A-**16**F. Upon receiving a CCM with a failure flag, an edge node **16**A-**16**F will transition from the primary ELAN to the secondary ELAN for transmitting and receiving traffic. Those edge nodes **16**A-**16**F that detect a failure, such as a loss of CCMs from any of the edge nodes **16**A-**16**F over the primary ELAN, will also switch to the secondary ELAN for delivering and receiving traffic. As such, the edge nodes **16**A-**16**F will switch to the secondary ELAN after detecting the failure condition through different means, and importantly, all of the edge nodes **16**A-**16**F will ultimately transition from the primary E-LAN to the secondary E-LAN, permanently or until the failure condition is resolved.

[0061] A communication flow illustrating the above embodiment is provided in FIGS. **11**A and **11**B. Initially, assume that each of the edge nodes **16**A-**16**D are providing CCMs to other ones of the edge nodes **16**A-**16**D on both the primary and secondary E-LANs. As such, edge node **16**C will send CCMs to the edge nodes **16**D, **16**B, and **16**A via the primary and secondary E-LANs (steps **500**, **502**, and **504**). Edge node **16**D will send CCMs to edge nodes **16**C, **16**B, and **16**A via the primary and secondary E-LANs (steps **506**, **508**, and **510**). Edge node **16**B will send CCMs to edge nodes **16**C, **16**D, and **16**A via the primary and secondary E-LANs (steps **512**, **514**, and **516**). Edge node **16**A will send CCMs to edge nodes **16**B, **16**C, and **16**D via the primary and secondary E-LANs (steps **518**, **520**, and **522**). Only four edge nodes **16**A-**16**D are provided in the communication flow for conciseness and clarity. In addition to providing the CCMs on both the primary and secondary E-LANs, each of the edge nodes **16**A-**16**D will transmit traffic only on the primary E-LAN (steps **524**, **526**, **528**, and **530**).

[0062] At this point, assume there is a failure that prevents the delivery of CCMs from the edge node **16**A to the edge node **16**B, and vice versa (step **532**). As a result, edge node **16**A will detect a loss of CCMs from edge node **16**B on the primary E-LAN (step **534**). Similarly, edge node **16**B will detect a loss of CCMs from edge node **16**A on the primary E-LAN (step **536**). Both edge nodes **16**A and **16**B will set a failure flag in the CCMs being provided on the primary E-LAN (step **538** and **540**) and will transmit the CCMs toward the other edge nodes **16**A-**16**D. Due to the failure, the CCMs will not be received by the edge nodes **16**A and **16**B, but the CCMs may be received by the edge nodes **16**C and **16**D (steps **542**, **544**, **546**, and **548**). Concurrently, the edge nodes **16**A and **16**B will also begin transmitting traffic on the secondary ELAN (steps **550** and **552**). At this point, the edge nodes **16**A and **16**B are sending CCMs with the failure flag via the primary ELAN and transmitting traffic, and perhaps regular CCMs, on the secondary ELAN.

[0063] Edge nodes **16**C and **16**D will detect the CCMs with the failure flag received from edge nodes **16**A and **16**B (steps **554** and **556**). In response to receiving a CCM with a failure flag, edge nodes **16**C and **16**D will begin transmitting traffic, and perhaps regular CCMs, on the secondary ELAN (steps **558** and **560**). As such, edge nodes **16**A and **16**B will rapidly detect a failure and switch from one ELAN to another for delivering traffic, as well as regular CCMs, while avoiding excessive and supplementary control messaging to detect the failure and facilitate a transition from one ELAN to another,

[0064] Turning now to FIG. **12**, a block representation of an edge node **16** is provided according to one embodiment of the present invention. The edge node **16** may include a control system **20** with sufficient memory **22** for the requisite software **24** and data **26** to operate as described above. The control system **20** may be associated with at least one communication interface **28** to facilitate communications over the backbone network **14** as well as associated edge networks **12**, which may facilitate Ethernet or other transport mechanisms.

[0065] Although the above embodiments are focused on Carrier Ethernet implementations, the present invention is applicable to various transport technologies that use multipoint or rooted multi-point topologies, which are analogous to E-TREE and E-LAN topologies in a Carrier Ethernet architecture.

[0066] Those skilled in the art will recognize improvements and modifications to the preferred embodiments of the present invention. All such improvements and modifications are considered within the scope of the concepts disclosed herein and the claims that follow.

What is claimed is:

1. A backbone network for serving a plurality of edge nodes, a first edge node being associated with a first virtual network and a second virtual network that are supported by the backbone network, the first edge node being operable to:

forward traffic over the first virtual network while receiving control messages from a first source over the first virtual network before a failure associated with the first virtual network; and

forward traffic over the second virtual network while receiving control messages from a second source over the second virtual network after the failure associated with the first virtual network.

2. The backbone network of claim **1**, wherein the first edge node is operable to:

detect the failure associated with the first virtual network based on loss of control messages received from the first source over the first virtual network; and

switch from forwarding traffic over the first virtual network to forwarding traffic over the second virtual network in response to detection of the failure associated with the first virtual network.

3. The backbone network of claim **1**, wherein the first edge node is operable to:

detect the failure associated with the first virtual network based on receipt of the control messages received from the second source over the second virtual network; and

switch from forwarding traffic over the first virtual network to forwarding traffic over the second virtual network in response to detection of the failure associated with the first virtual network.

4. The backbone network of claim **3**, wherein the control messages received from the second source over the second virtual network are provided only in response to the failure associated with the first virtual network.

5. The backbone network of claim **1**, wherein at least one of the control messages comprises a failure indicator and the first edge node is operable to:

detect the failure indicator in a received control message; and

switch from forwarding traffic over the first virtual network to forwarding traffic over the second virtual network in response to detection of the failure indicator.

6. The backbone network of claim **1**, wherein the backbone network is an Ethernet-based network and the first and second virtual networks are Ethernet virtual connections of the Ethernet-based network.

7. The backbone network of claim **1**, wherein the first edge node is further operable to:

receive traffic over the first virtual network while receiving the control messages from the first source over the first virtual network; and

receive traffic over the second virtual network while receiving the control messages from the second source over the second virtual network.

8. The backbone network of claim **1**, wherein the first edge node is operable to:

exchange traffic with a second edge node of the backbone network over the first virtual network; and

exchange traffic with a third edge node of the backbone network over the second virtual network.

9. The backbone network of claim **8**, wherein the first edge node is a spoke edge node and the second and third edge nodes are hub edge nodes in a rooted multi-point architecture

10. The backbone network of claim **9**, wherein the first virtual network and the second virtual network represent separate tree-configured virtual Ethernet connections of an Ethernet-based network.

11. The backbone network of claim **9**, wherein the first virtual network and the second virtual network represent separate E-LAN configured virtual Ethernet connections of an Ethernet-based network.

12. The backbone network of claim **1**, wherein each of the plurality of edge nodes is an edge node in a multi-point architecture.

13. The backbone network of claim **1**, wherein:

the first edge node is operable to receive a series of control messages over at least one of the of the first virtual network and the second virtual network, the series of control messages comprising a source address that is associated with a source of the series of control messages; and

forwarding the traffic comprises addressing the traffic to the source address and sending the traffic over the at least one of the first virtual network and the second virtual network toward the source of the series of control messages.

14. The backbone network of claim **13**, wherein:

the series of control messages comprises a first series of control messages received over the first virtual network from a first source prior to the failure and a second series of control messages received over the second virtual network from a second source after the failure; and

the source address of the first series of control messages received over the first virtual network is different from the source address for the second series of control messages received over the second virtual network, such that the traffic sent over the first virtual network prior to the detecting the failure is addressed to the first source and the traffic sent over the second virtual network after the detecting the failure is addressed to the second source.

15. The backbone network of claim **13**, wherein the control messages of the series of control messages from the source are normally received at a rate substantially higher than one every 50 milliseconds.

16. The backbone network of claim **13**, wherein the control messages of the series of control messages from the source are normally received at a rate around about one every 10 milliseconds.

17. The backbone network of claim **1**, wherein the first edge node is operable to send control messages to the at least one of the plurality of edge nodes at a rate substantially higher than one every 50 milliseconds.

18. The backbone network of claim **1**, wherein the first edge node is operable to:

send a series of control messages to at least one of the plurality of edge nodes over the first virtual network; and

upon detecting the failure associated with the first virtual network, provide a failure indicator in control messages of the series of control messages.

19. The backbone network of claim **1**, wherein the first virtual network and the second virtual network are configured to not share intermediate nodes, such that no intermediate node of either of the first virtual network and the second virtual network is part of a shared risk link group.

* * * * *