

United States Patent and Trademark Office
Before the Patent Trial and Appeal Board

Juniper Networks, Inc.
(Petitioner)
v.
Portsmouth Network Corp.,
(Patent Owner)

IPR2024-00893 | **U.S. Patent No. 6,917,986**

Juniper Exhibit 1019
Juniper v. Portsmouth
IPR2024-00893

September 12, 2025

Challenged U.S. Patent No. 6,917,986

EX1001

(12) **United States Patent**
Mor et al.

(10) **Patent No.: US 6,917,986 B2**
(45) **Date of Patent: Jul. 12, 2005**

(54) **FAST FAILURE PROTECTION USING
REDUNDANT NETWORK EDGE PORTS**

6,680,906 B1 1/2004 Nguyen

OTHER PUBLICATIONS

(75) **Inventors:** Gal Mor, Rishon le Zion (IL); Shmuel
Ilan, Ra'anana (IL)

"IEEE Standard for Information Technology, Telecommu-
nications and Information Exchange between Systems,
Local and Metropolitan Area Network, Common Specifi-
cations, Part 3: Media Access Control (MAC) Bridges", Pub-
lished as ANSI/IEEE Standard 802.1D (1998). Available at:
standards.ieee.org/catalog/IEEE802.1.html.

(73) **Assignee:** Corrigent Systems Ltd., Tel Aviv (IL)

(*) **Notice:** Subject to any disclaimer, the term of this
patent is extended or adjusted under 35
U.S.C. 154(b) by 734 days.

(21) **Appl. No.:** 10/036,518

Primary Examiner—William A. Cuchlinski, Jr.
Assistant Examiner—Thanh T. Nguyen
(74) *Attorney, Agent, or Firm*—Nixon & Vanderhye P.C.

(22) **Filed:** Jan. 7, 2002

ABSTRACT

(65) **Prior Publication Data**

US 2003/020618 A1 Nov. 6, 2003

(51) **Int. Cl.** G06F 15/173

(52) **U.S. Cl.** 709/238; 709/208; 709/226;

709/209; 709/239; 707/104.1; 370/396;

370/252; 370/352; 370/389

(58) **Field of Search** 709/238; 239;

709/208; 209; 226; 707/104.1; 370/396;

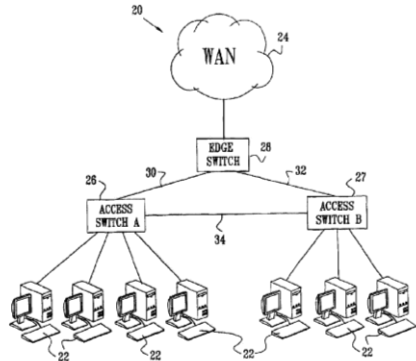
252; 352; 389

References Cited

U.S. PATENT DOCUMENTS

5,159,595 A 10/1992 Flanagan et al.
5,307,353 A 4/1994 Yamashita et al.
5,925,137 A 7/1999 Okano et al.
6,246,667 B1 6/2001 Bullitt et al.

24 Claims, 4 Drawing Sheets



Juniper Exhibit 1001
Juniper v. Portsmouth

(12) **United States Patent**
Mor et al.

(10) **Patent No.: US 6,917,986 B2**
(45) **Date of Patent: Jul. 12, 2005**

(54) **FAST FAILURE PROTECTION USING
REDUNDANT NETWORK EDGE PORTS**

(22) **Filed: Jan. 7, 2002**

EX1001

Overview of the Petition Grounds

Petition at 20

'986 Patent (IPR2024-00893)

GROUND	CLAIMS	BASIS
1. Gai in view of IEEE 1999	1 , 2-3, 6-11, 12 , 13 , 14-15, 18-23, 24	§103
2. Gai in view of IEEE 1999 and Jain	4-5, 16-17	§103

Petition at 20

Cisco IPR (IPR2024-00505), Final Written Decision

In summary:				
Claim(s)	35 U.S.C. §	Reference(s)/ Basis	Claim(s) Shown Unpatentable	Claim(s) Not Shown Unpatentable
1, 2, 4–7, 9–14, 16–19, 21–24	103(a)	Gai	1, 2, 4–7, 9, 10, 13, 14, 16–19, 21, 22	11, 12, 23, 24

Ground 1: Challenged Independent Claim 1

ID, Paper 10 at 25, 27, 28, 30, 31
POR at 9-11

'986 Patent, Claim 1

1. A method for network communication, comprising:
coupling an edge switch in a network to one or more access switches via at least first and second redundant links, respectively connected to first and second ports of the edge switch;
placing the first port in an active state, while placing the second port in a blocking state;
conveying communication traffic over the first link while the first port is in the active state, thereby causing each of the switches to build a respective database for use in forwarding the traffic;
responsive to a failure associated with the first link, placing the second port in the active state and the first port in the blocking state; and
sending dummy traffic from the edge switch over the second link to the one or more access switches, so as to cause each of the one or more access switches to modify its respective database responsive to the second port being in the active state.

**Conventional Teachings per Gai
(No Dispute)**

Only Dispute

Institution Decision (ID), Paper 10 at 25, 27, 28, 30, 31;
POR at 9-11

'986 Patent, Claim 12

12. A method for network communication, comprising:
coupling a first bridge in a network to a second bridge located downstream of the first bridge, by connecting at least first and second redundant links to respective first and second ports of the first bridge so as to communicate with the second bridge;

placing the first port in an active state, while placing the second port in a blocking state;

conveying communication traffic over the first link while the first port is in the active state, thereby causing the bridges to build respective databases for use in forwarding the traffic;

responsive to a failure associated with the first link, placing the second port in the active state and the first port in the blocking state; and

sending dummy traffic from the first bridge over the second link downstream to the second bridge, so as to cause the second bridge to modify its database responsive to the second port being in the active state.

**Conventional Teachings per Gai
(No Dispute)**

Only Dispute

Spanning Tree Protocol

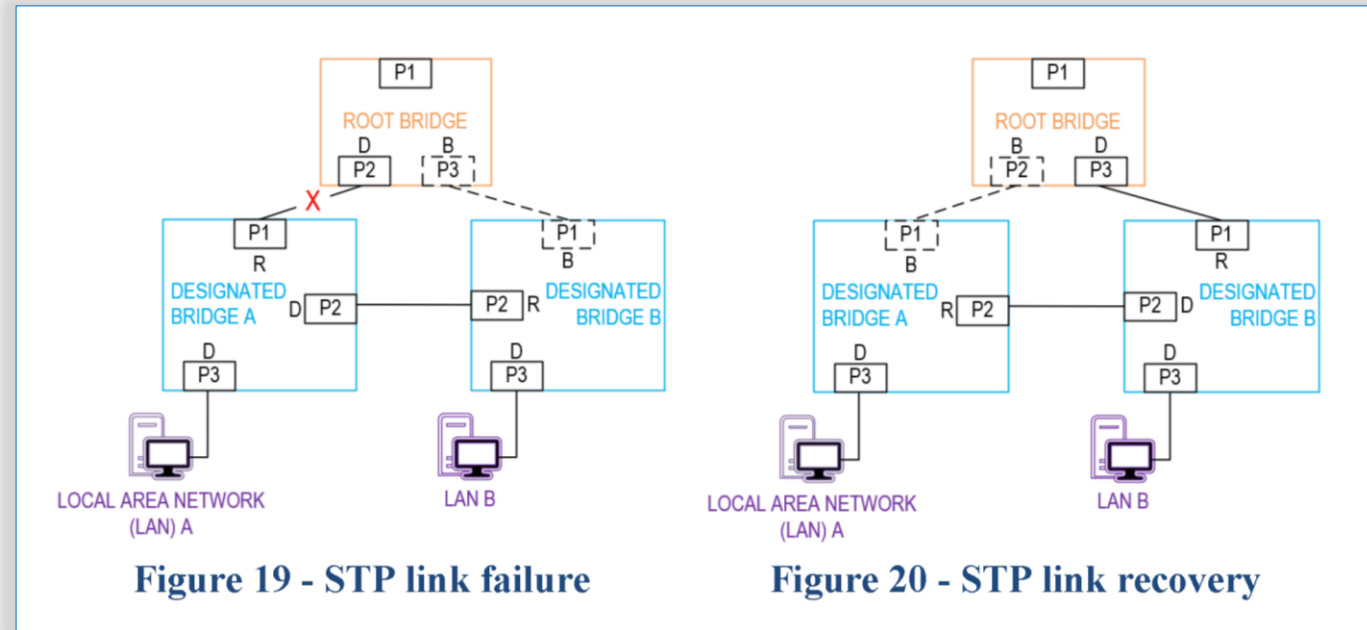
EX1003 at ¶¶38-39, 87-90

Declaration of Bruce McNair

38. Just as a forest may have a number of separate trees, a network may have a series of disconnected segments. To enable connectivity between all the nodes of the network, a single tree that “spans” all the nodes is desirable.

“Spanning” a set of nodes in networking parlance means to connect that set of nodes. In network terminology, a set of connections that spans a set of nodes is called a spanning tree.

39. For most networks, there may be several choices of how the nodes may be interconnected, allowing alternate arrangements of links to provide the desired connectivity. This is where the cost of operating the network enters. Some links may have a lower financial cost or may provide faster connections. If an effective cost to the end user is attributed to a lower speed connection, it may be desirable to minimize the overall cost of using the network. A spanning tree that connects all the nodes while minimizing cost is called a minimum cost spanning tree or a minimum spanning tree. As discussed later, IEEE standardized a protocol to find and maintain the minimum spanning tree for a given network in the presence of failures within the network. This IEEE standardized protocol is called the Spanning Tree Protocol (STP).



EX1003 at ¶¶38-39, 87-90

The '986 Patent and the Petition's Prior Art References Are All Directed to Improving the Spanning Tree Protocol

EX1001 at 4:43-49
EX1005 at Abstract
EX1006 at 5:10-13, 6:52-54
EX1009 at 161
Petition at 3, 25, 27, 47, 62

'986 Patent

do so. Therefore, preferred embodiments of the present invention maintain compatibility with STP as provided by the current 802.1D standard. The standard spanning tree algorithm will not only run transparently over the modified bridge, but actually should run faster, since the “radius” of the tree (i.e., the number of branching points over which it must run) is reduced.

EX1001 at 4:43-49

IEEE-1999

ABSTRACT

EtheReal is a real-time Fast Ethernet switch architecture that provides bandwidth guarantees to distributed multimedia applications without OS or hardware modifications on the host machines. It implements true link-layer multicast, and offers a natural match to support network-layer QoS protocols such as RSVP. Because real-time performance guarantees fundamentally require state to be installed inside the network, link/switch failures could lead to significant disruption to the QoS promised to the user applications. This paper describes the fault detection and recovery mechanism supported by the *EtheReal* architecture, and reports on the performance measurements of the initial prototype implementation. The heart of *EtheReal*'s fault detection and recovery mechanism is a fast spanning tree reconfiguration algorithm to reduce the total fault recovery time, and a delayed link inactivation scheme that allows real-time connections which are not affected by the failed links/switches to continue to exist, even though some of the links are marked as “blocked” in the new spanning tree topology. Measurements on the prototype show that the fault detection and recovery time on a network whose diameter is 10 hops are 220ms and 31ms, respectively. This combined delay corresponds to a minor jitter in real-time audio/video communication, and is a significant improvement over the standard IEEE 802.1d implementation, which takes on the order of 30 sec.

EX1009 at 161

Gai

[57]

ABSTRACT

A method that rapidly reconfigures a computer network having a plurality of devices executing the spanning tree algorithm. First, one or more devices are configured and arranged so that one port, providing connectivity to the root, is in the forwarding state and the remaining ports, providing connectivity to the root, are in the blocked state. Next, one or more of the blocked ports are designated as back-up ports.

EX1005 at Abstract

set forth in Table 1. Network diameter is defined by the IEEE 802.1D standard as the maximum number of devices between any two points of attachment of end stations and is an argument of the Become_Root_Primary command 300.

EX1005 at 9:12-15

Jain

The present invention provides for an enhancement to the spanning tree protocol based on identifying port roles, and managing updates of the forwarding databases after topology changes, according to the roles of the ports.

* * *

entity. The forwarding database also referred to as a “filtering database” in the IEEE 802.1D Standard, is typically configured as a single filtering database object per bridge.

EX1006 at 5:10-13, 6:52-54

Gai Discloses the Same Dummy Traffic as the '986 Patent

EX1001 at 3:15-33
Petition at 40

'986 Patent

U.S. Pat. No. 6,032,194, whose disclosure is incorporated herein by reference, describes a method for rapidly reconfiguring a computer network, based on certain modifications to the standard STP. The method assumes that the switches in the network, including both core and access switches, have multiple trunk ports providing connectivity to the STP root via trunk switches. (These trunk switches are comparable to the core switches and edge switches defined above, and the trunk ports are on the upstream side of each switch, i.e., the side leading toward the root.) One of the trunk ports of each switch is placed in the forwarding state, while the others are designated as back-up ports. Upon detection of a failure at the active forwarding port, the switch immediately places one of the back-up trunk ports in the forwarding state, so that it becomes the new active port. The switch then transmits dummy multicast messages toward the root (upstream) through the newly-active trunk port, each containing the source address of a network entity that is coupled to the switch itself or is downstream of the switch. Upon receiving these messages, other devices in the network learn to use the new active port. In order to implement this rapid

EX1001 at 3:15-35

Petition

The '986 patent and Gai send the same dummy traffic between switches.

The '986 patent admits that Gai discloses sending dummy multicast messages (*i.e.*, dummy traffic) upstream using a root port after detection of a link failure and, “[u]pon receiving these messages, other devices in the network learn to use the new active port.” EX1001, 3:26-35. McNair, ¶165.

Petition at 4-5, 22, 40

POR

No Dispute

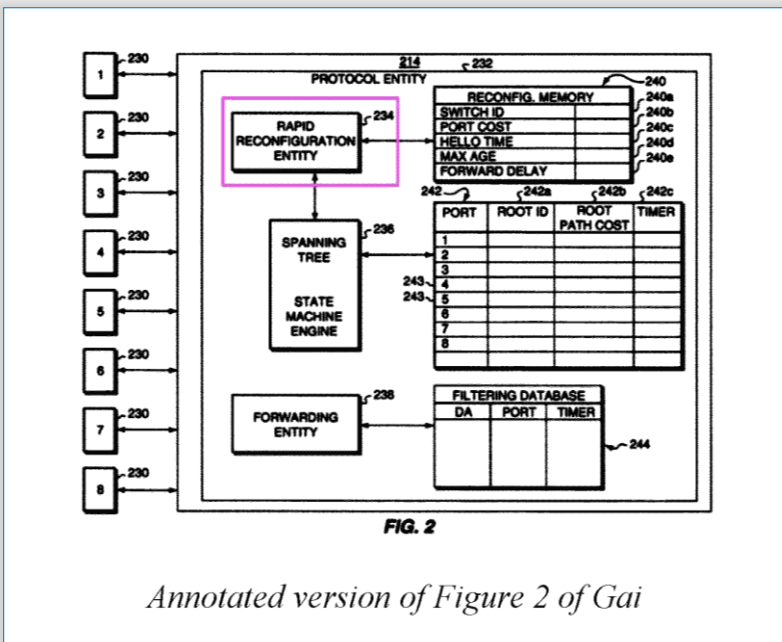
Gai Teaches That Dummy Traffic Can Be Sent From Any Switch

Petition at 40, 42-43
ID, Paper 10 at 34

Petition

using rapid reconfiguration entity 234, also applies to backbone switch 120. Thus, POSAs would have understood that the switch implementation shown in Figure 2 of Gai applies to any switch in Gai, or at least that it would have been obvious to use it with any switch in Gai. Therefore, POSAs would have understood that both access switches and backbones switches in Gai can generate and transmit dummy multicast messages. McNair, ¶169.

Petition at 43



Petition at 42

Gai

devices. Thus, the identity of the new root port (e.g., port number four) must be propagated to the other devices, such as access switches 115–116 and backbone switches 120–125 to prevent messages from being lost or misdirected. Switch 114, via rapid reconfiguration entity 234, preferably informs the other devices of its new forwarding port by transmitting dummy multicast packets through the new port, as indicated at block 348. FIG. 4 is a highly schematic illustration of a

EX1005 at 12:59-66

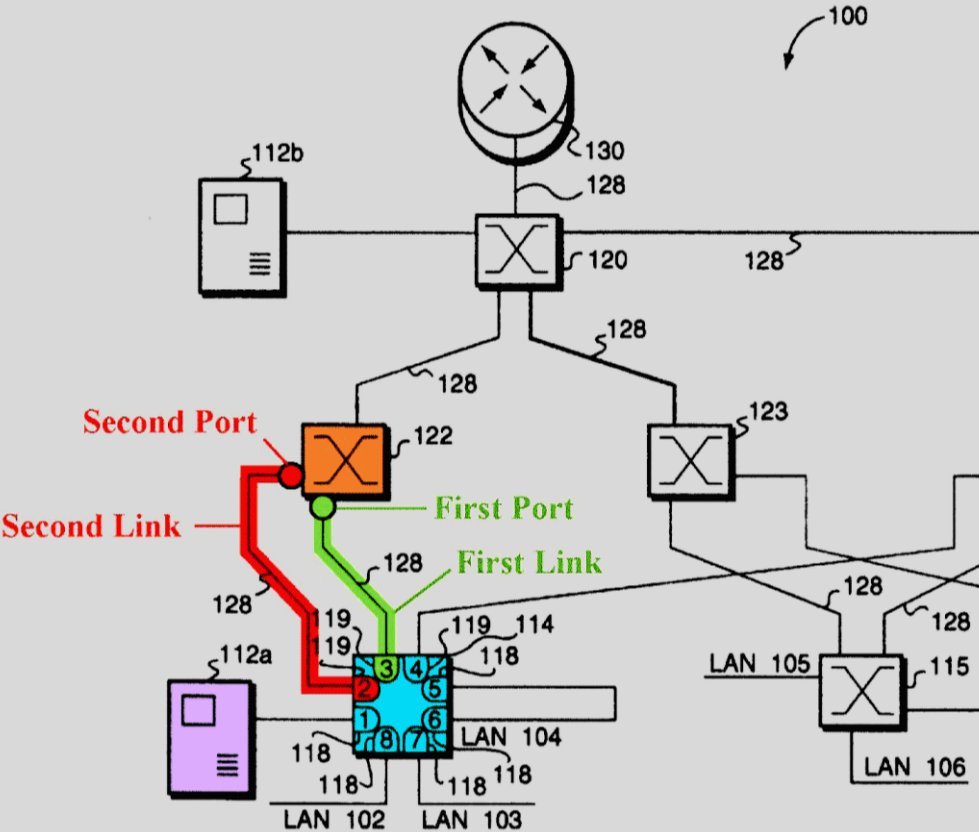
most heavily utilized server, as shown at block 310. In response to the Become_Root_Primary command 300, the rapid reconfiguration entity 234 of the selected backbone switch 120, at block 312, modifies the settable portion of the corresponding switch ID stored at reconfiguration memory 240. In particular, the value of the switch ID is significantly

EX1005 at 8:66-9:3

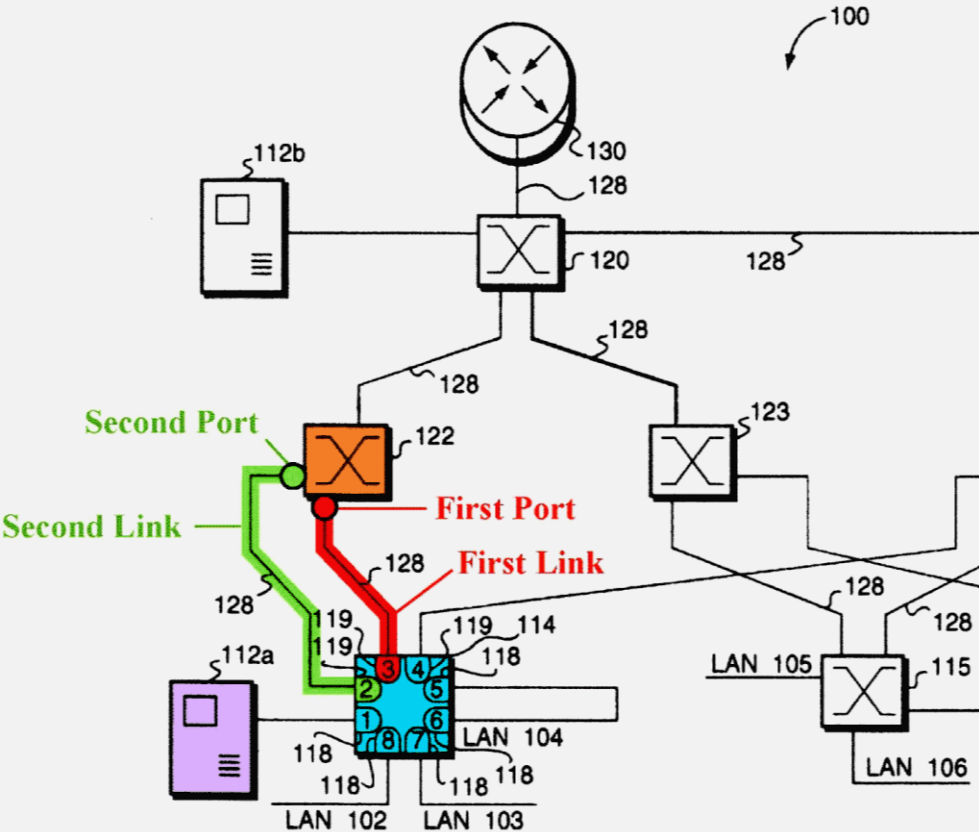
Institution Decision

As Petitioner points out, Gai describes that the rapid reconfiguration process can be implemented on both edge and access switches.¹⁰ Pet. 43 (citing Ex. 1005, 8:66–9:3, describing that rapid reconfiguration entity 234 is also implemented on backbone switch 120, which indicates it can be applied to any switch). Gai discusses how to implement fast network recovery

ID, Paper 10 at 34



Annotated Figure 1 of Gai before link failure



Annotated Figure 1 of Gai after link failure

IEEE-1999

ABSTRACT

EtheReal is a real-time Fast *Ethernet* switch architecture that provides bandwidth guarantees to distributed multimedia applications without OS or hardware modifications on the host machines. It implements true link-layer multicast, and offers a natural match to support network-layer QoS protocols such as RSVP. Because real-time performance guarantees fundamentally require state to be installed inside the network, link/switch failures could lead to significant disruption to the QoS promised to the user applications. This paper describes the fault detection and recovery mechanism supported by the *EtheReal* architecture, and reports on the performance measurements of the initial prototype implementation. The heart of *EtheReal*'s fault detection and recovery mechanism is a fast spanning tree reconfiguration algorithm to reduce the total fault recovery time, and a delayed link inactivation scheme that allows real-time connections which are not affected by the failed links/switches to continue to exist, even though some of the links are marked as "blocked" in the new spanning tree topology. Measurements on the prototype show that the fault detection and recovery time on a network whose diameter is 10 hops are 220ms and 31ms, respectively. This combined delay corresponds to a minor jitter in real-time audio/video communication, and is a significant improvement over the standard IEEE 802.1d implementation, which takes on the order of 30 sec.

EX1009 at 161

4.2 FAULT RECOVERY

To analyze the scalability of our approach, we instrumented our prototype to measure the overhead of the various components of a fault recovery operation. From the perspective of the affected connections, the fault recovery mechanism spends time in:

1. Terminating connections going through the faulty link/switch.
2. Reconfiguring the spanning tree
3. Re-establishing the connections that were torn down.

To terminate the connections through a failed link we need to (a) identify the connection identifiers of the failed connections (b) send a teardown message to the upstream/downstream sources of outgoing connections and (c) tear the connection to the upstream/downstream destinations if incoming connections. The CPU overhead of processing the connection queue is 95 ns/connection, which includes time spent in freeing up resources used by the connection. The cost of sending a teardown message to the next upstream hop involves formulating the message and setting up a DMA transaction to transfer the message, which accounts for a total of 2us/message/hop.

EX1009 at 167

Petition

IEEE 1999 explains that switches both upstream and downstream of a link failure detect the failure. EX1009, 166 ("the fault would be independently detected by switches both upstream as well as downstream of the fault"). Each switch that detects a failure takes steps to rebuild the spanning tree. EX1009, 166 ("The fault recovery mechanism on each switch that detects the fault...[r]ebuild[s] the spanning tree"). As part of the spanning tree reconfiguration, to improve recovery time, each switch that detected a link failure sends "teardown messages" to both upstream and downstream switches to "terminate the connections through a failed link." EX1009, 167. McNair, ¶¶101-102, 134.

Petition at 25

PO's Expert Dr. Olivier

40. IEEE 1999 discloses the *EtheReal* switch architecture including fault detection and recovery mechanisms supported by the *EtheReal* architecture. *See* EX1009 (IEEE 1999) at Abstract. IEEE 1999's fault detection recovery mechanism begins with detecting a change in the network topology. *See id.* at 166. In the interior of the network, the fault would be independently detected by switches both upstream as well as downstream of the fault. *Id.* ("In the interior of the network, the fault would be independently detected by switches both upstream as well as downstream of the fault."). After detection, these switches perform the following

EX2002 at ¶40

43. Once the teardown messages are received and the connections are torn down, the fault recovery mechanism begins rebuilding the spanning tree. *See id.* at 166. The reconfiguration of the spanning tree is handled in three phases: (1) a startup phase (SP), a (2) startup completion phase (SCP), and (3) a topology discovery complete phase (TDC). *See id.* at 165, 167. In the SP phase of the spanning tree reconfiguration process, a switch in the network transmits a message through all its ports to its neighbors to join the spanning tree. *See id.* at 165. A node receiving the

EX2002 at ¶43

Both Gai and IEEE-1999 Are Directed To:

1

The same technical field



Spanning Tree networks

2

The same IEEE network standard



IEEE 802.1D

3

The same technical problem



Reducing link failure
recovery time

4

The same technical approach



Detecting link failures and sending
messages to switches to rebuild
the Spanning Tree

The Motivation to Modify Gai → To Reduce The Time Taken to Rectify Network Failures

Petition

The '986 patent admits Gai discloses sending dummy traffic upstream from a switch that detects a link failure on its root port. EX1001, 3:26-33. Gai discloses sending dummy traffic to reduce link failure recovery time. EX1005, 6:42-53, 15:21-25. Aside from the network initialization commands discussed above

In view of IEEE 1999, POSAs would have had reason to modify Gai's technique by sending Gai's dummy traffic downstream after a downstream link failure (as the '986 patent describes), in addition to sending dummy traffic upstream after an upstream link failure (as Gai describes). This has the same benefit already described in Gai, *i.e.*, reducing the time needed for network self-healing. EX1005, 6:42-53, 15:21-25. McNair, ¶137.

With reference to downstream directionality, it would have been obvious to POSAs to send dummy multicast messages both upstream, as explained above, as well as downstream, for the benefit of reducing the total link failure recovery time because the link failure can be addressed by upstream and downstream switches in parallel. As explained above in §VII.A.3, POSAs would have been motivated, by IEEE 1999, to modify Gai to send dummy traffic downstream as well as upstream to achieve this benefit, and would have enjoyed a reasonable expectation of success in doing so. McNair, ¶173.

In short, when modified in view of IEEE 1999's disclosure that it was advantageous to send self-healing messages both upstream and downstream after a link failure either upstream or downstream, Gai's backbone switches (*i.e.*, the claimed "edge switches" and "bridges") send Gai's dummy traffic both upstream and downstream, regardless of whether a link failure is detected upstream of the switch or downstream of the switch. This means that modified Gai's backbone switch / edge switch / bridge sends dummy traffic downstream to Gai's access switch / bridge, as recited in the challenged claims. Doing so achieves Gai's goal (shared by IEEE 1999) of reducing the time taken to rectify network failures, including between an edge switch and an access switch. EX1005, 11:53-55, 11:64-12:12. McNair, ¶144.

The Modification to Gai

Gai's System (Pre-Modification)

Gai and the '986 patent both transmit dummy traffic to notify other switches of port configuration changes. EX1001, 3:29-35 (describing Gai's operation), 4:17-28 (describing the '986 patent's operation). To notify upstream switches, Gai's downstream switch transmits dummy traffic upstream through the new active link to update the filtering database of the upstream switch. EX1005, 12:56-66. In summary, Gai discloses setting up a spanning tree network, performing port reconfiguration after link failure, and transmitting dummy traffic, all as described in the '986 patent. McNair, ¶¶104, 124-125.

Gai as Modified In View of IEEE-1999's Teachings

In short, when modified in view of IEEE 1999's disclosure that it was advantageous to send self-healing messages both upstream and downstream after a link failure either upstream or downstream, Gai's backbone switches (*i.e.*, the claimed "edge switches" and "bridges") send Gai's dummy traffic both upstream and downstream, regardless of whether a link failure is detected upstream of the switch or downstream of the switch. This means that modified Gai's backbone switch / edge switch / bridge sends dummy traffic downstream to Gai's access switch / bridge, as recited in the challenged claims. Doing so achieves Gai's goal (shared by IEEE 1999) of reducing the time taken to rectify network failures, including between an edge switch and an access switch. EX1005, 11:53-55, 11:64-12:12. McNair, ¶144.

Dummy Messages and Teardown Messages Are Both Part of the Spanning Tree Reconfiguration Process

POR at 7
Petition at 4, 25

POR

For example, Petitioner argues that Gai could be modified to send dummy traffic downstream because IEEE 1999 discloses sending such messages upstream and downstream. Petition at 27. However, Petitioner acknowledges that IEEE 1999 describes its messages as “teardown messages” and not dummy messages. *Id.*; see

POR at 7

Petition (Discussing Gai)

The '986 patent admits that using dummy frames to communicate topology changes immediately in a Spanning Tree network was already known from U.S. Patent No. 6,032,194 (“Gai,” EX1005), which it incorporates by reference for this very disclosure. 3:15-29. The '986 patent argues that its scheme differs from

Petition at 4

Petition (Discussing IEEE-1999)

the spanning tree”). As part of the spanning tree reconfiguration, to improve recovery time, each switch that detected a link failure sends “teardown messages” to both upstream and downstream switches to “terminate the connections through a failed link.” EX1009, 167. McNair, ¶¶101-102, 134.

Petition at 25

PO Misapprehends the Petition's Use of Gai's Dummy Messages in the Gai Modification

POPR at 8
Petition at 29
ID, Paper 10 at 36-37
POR at 10

Petition

In short, when modified in view of IEEE 1999's disclosure that it was advantageous to send self-healing messages both upstream and downstream after a link failure either upstream or downstream, Gai's backbone switches (*i.e.*, the claimed "edge switches" and "bridges") send Gai's dummy traffic both upstream and downstream, regardless of whether a link failure is detected upstream of the switch or downstream of the switch. This means that modified Gai's backbone

Petition at 29

POPR

failed link." *Id.* at 25. Petitioner further asserts that a switch receiving a teardown message will update its filtering database such that the failed connection is terminated. *Id.* at 26. Therefore, Petitioner appears to equate IEEE 1999's teardown messages with the claimed dummy traffic.

POPR at 8

Institution Decision

We do not understand Petitioner to be arguing that the teardown message itself should be taken from IEEE 1999 and combined with Gai's rapid reconfiguration process, as Patent Owner seems to contend. Prelim. Resp. 7–12. Rather, Petitioner asserts that in view of IEEE 1999's teaching that the teardown message is sent both upstream and downstream in order to more rapidly recover from a link failure, a POSITA would understand that the same principle could be applied to the dummy traffic being sent out during the rapid recovery process in Gai. Pet. 26–30. So it is the dummy traffic described in Gai, rather than the teardown message in IEEE 1999, that Petitioner alleges meets limitation 1e. *Id.*; Ex. 1003 ¶¶ 54, 134–144. Therefore, at this stage, we find Patent Owner's arguments trying to distinguish the teardown message from dummy traffic in limitation 1e to be unpersuasive.¹¹

POR

Petitioner relies on IEEE 1999 to disclose the transmission of self-healing messages from an edge switch to one or more access switches and specifically identifies IEEE 1999's teardown messages as the appropriate message to modify filtering databases. See e.g., Petition at 26–29, 46.¹ However, as previously

POR at 10

These state
the ***opposite***, Reply at 4-5

PO Does Not Address the Motivations Explained by Petitioner

POR at 7
EX2002 at ¶¶46-47

POR

Petitioner argues that a POSA would be motivated to modify Gai with IEEE 1999 to send Gai’s dummy traffic downstream after a downstream link failure because such a modification “has the same benefit already described in Gai, *i.e.*, reducing the time needed for network self-healing.” Petition at 26; *see* EX2002, ¶ 47. Petitioner’s arguments are flawed. EX2002, ¶¶ 46–47.

POR at 7

PO Expert Declaration

46. In my opinion, a POSITA would not have been motivated to combine Gai’s teachings with IEEE 1999 to send dummy traffic downstream or to send dummy traffic to modify a database responsive to the second port being in the active state, as recited in the independent challenged claims of the ’986 patent.

EX2002 at ¶46

PO Expert Declaration

47. Mr. McNair appears to acknowledge that Gai does not disclose sending dummy traffic downstream and instead proposes modifying Gai with IEEE 1999 to address the deficiency. *See e.g., id.* at ¶¶ 137, 173. In my opinion, Mr. McNair’s reasoning for such modifications are unsupported and flawed. For example, Mr. McNair states that “POSAs would have had reason (*i.e.*, faster recovery from a link failure as explained in IEEE 1999) to modify Gai’s technique by sending Gai’s dummy traffic downstream after a downstream link failure (as the ’986 patent describes), in addition to sending dummy traffic upstream after an upstream link failure (as Gai describes).” *Id.* at ¶ 137. Mr. McNair acknowledges that IEEE 1999 describes its messages as “teardown messages” and not dummy messages. *Id.* at 138. However, Mr. McNair asserts that the function of the teardown messages “is the same as Gai’s ‘dummy messages.’” *Id.* Mr. McNair’s comparison is incorrect. As previously mentioned in section IX.B, IEEE 1999’s fault detection recovery mechanism detects a fault in a link and performs the following functions: (1) discard all best effort traffic, (2) terminate all connections going through the faulty link, (3)

EX2002 at ¶47

rebuild the spanning tree, and (4) reestablish the connections that were torn down. *See supra* section IX.B; EX1009 at 166. IEEE 1999’s teardown messages are transmitted to host computers to tear down connections going across the faulty link. They are not received by switches in the network. In addition, these messages are NOT used to learn MAC address, but to instead to inform host computers that the network is unstable and that their particular connections have been torn down. Also, this happens before the spanning tree is reconfigured and before each switch begins forwarding traffic along new routes. *See* EX1009 at 166, 167. This is not how Gai describes dummy messages. Gai discloses that dummy multicast messages are transmitted “[u]pon transition to the new forwarding port” and are sent so that “other devices within the network learn to utilize the new forwarding port, rather than the failed port, when directing messages to these entities.” EX1005 at 5:54–6:5. Therefore, Gai is clear that its dummy multicast messages are transmitted as part of the reconfiguration process and only transmitted once a device has already successfully transitioned to a new forwarding port. As such, Gai’s dummy messages are sent at a different time and serve a different purpose than IEEE 1999’s teardown messages.

No Dispute

Petition

In short, when modified in view of IEEE 1999's disclosure that it was advantageous to send self-healing messages both upstream and downstream after a link failure either upstream or downstream, Gai's backbone switches (*i.e.*, the claimed "edge switches" and "bridges") send Gai's dummy traffic both upstream and downstream, regardless of whether a link failure is detected upstream of the switch or downstream of the switch. This means that modified Gai's backbone switch / edge switch / bridge sends dummy traffic downstream to Gai's access switch / bridge, as recited in the challenged claims. Doing so achieves Gai's goal (shared by IEEE 1999) of reducing the time taken to rectify network failures, including between an edge switch and an access switch. EX1005, 11:53-55, 11:64-12:12. McNair, ¶144.

Petition at 29

With reference to downstream directionality, it would have been obvious to POSAs to send dummy multicast messages both upstream, as explained above, as well as downstream, for the benefit of reducing the total link failure recovery time because the link failure can be addressed by upstream and downstream switches in parallel. As explained above in §VII.A.3, POSAs would have been motivated, by IEEE 1999, to modify Gai to send dummy traffic downstream as well as upstream to achieve this benefit, and would have enjoyed a reasonable expectation of success in doing so. McNair, ¶173.

Petition at 45

The Explained Motivation

Petition

In short, when modified in view of IEEE 1999's disclosure that it was advantageous to send self-healing messages both upstream and downstream after a link failure either upstream or downstream, Gai's backbone switches (*i.e.*, the claimed "edge switches" and "bridges") send Gai's dummy traffic both upstream and downstream, regardless of whether a link failure is detected upstream of the switch or downstream of the switch. This means that modified Gai's backbone switch / edge switch / bridge sends dummy traffic downstream to Gai's access switch / bridge, as recited in the challenged claims. Doing so achieves Gai's goal (shared by IEEE 1999) of reducing the time taken to rectify network failures, including between an edge switch and an access switch. EX1005, 11:53-55, 11:64-12:12. McNair, ¶144.

Petition at 29

With reference to downstream directionality, it would have been obvious to POSAs to send dummy multicast messages both upstream, as explained above, as well as downstream, for the benefit of reducing the total link failure recovery time because the link failure can be addressed by upstream and downstream switches in parallel. As explained above in §VII.A.3, POSAs would have been motivated, by IEEE 1999, to modify Gai to send dummy traffic downstream as well as upstream to achieve this benefit, and would have enjoyed a reasonable expectation of success in doing so. McNair, ¶173.

Petition at 45

Institution Decision

A similar mechanism is described in IEEE 1999 where, as part of the reconfiguration process, a teardown message is sent both upstream and downstream from the switch where the failure is detected. Ex. 1009 at 166. According to IEEE 1999, "each switch that detects the fault" takes steps to "[r]ebuild the spanning tree." *Id.* As part of the spanning tree reconfiguration that takes place after a failure, each switch that detects the failure sends teardown messages to both upstream and downstream switches to "terminate the connection[] through a failed link." Ex. 1009 at 167. By sending teardown messages in both directions, along with optimizations in rebuilding the spanning tree protocol, the network downtime in response to a link failure is substantially reduced. *Id.* at 164–167. We are persuaded, based on the current record, that a POSITA would apply the teaching from IEEE 1999 that sending messages for reconfiguring the network after a link failure both upstream and downstream can reduce the network downtime to the rapid reconfiguration process in Gai when that process is implemented on an edge switch, such as on switch 122. See Pet. 26–30, 40–46; Ex. 1003 ¶¶ 165–175.

ID, Paper 10 at 36

PO's Sur-Reply

I. INTRODUCTION

Petitioner's Reply does not remedy the deficiencies of its Petition. Instead, Petitioner attempts to gloss over the issues raised in the Patent Owner Response ("POR") and fails to provide a reason for why a POSA would be motivated to combine IEEE 1999 and Gai and/or an explanation for how a POSA would combine IEEE 1999 and Gai to render the challenged claims obvious. The Federal Circuit, however, has established that a Petitioner must provide evidence-supported reasoning for why and how references should be combined. Petitioner has failed to provide such reasoning. For the reasons in this Sur-Reply and in the POR, Petitioner failed to satisfy its burden of demonstrating invalidity of any of the challenged claims.

PO's Sur-Reply,
Paper 23 at 1



Petitioner Explained How POSAs Would Have Modified Gai in View of IEEE-1999

Petition at 27-28, 30

Petition: How to Modify Gai

switches.” EX1005, 6:9-11. Gai discloses entering an Enable_Uplinkfast command at an access switch (*e.g.*, access switch 114) to block a failed port (*e.g.*, the root port) and configure the access switch to send dummy traffic upstream through a new active port. EX1005, 6:39-46. When modified in view of IEEE 1999’s disclosure that self-healing network traffic can advantageously be sent both upstream and downstream, the command would be entered at a backbone switch (*e.g.*, backbone switch 122) to block a failed port (*e.g.*, a designated port connected to access switch 114) and configure the backbone switch to send dummy traffic downstream to an access switch (*e.g.* access switch 114) through a new active port.

While Gai uses its protocol signaling command, Enable_Uplinkfast, to send dummy traffic through a new root port, POSAs would deem it an obvious implementation detail to modify the parameters of the command to be applicable to other ports, such as a designated port on a backbone switch. McNair, ¶139.

Petition at 27-28

Petition: Reasonable Expectation of Success

Finally, a POSA would have had a reasonable expectation of success in modifying Gai to have its backbone switches / edge switches / bridges send dummy traffic both upstream and downstream because signaling protocols are routinely modified to provide for different control signaling, such that there would have been no additional technical challenge in sending those messages both upstream and downstream. Further, Gai’s mechanism for sending dummy traffic (*i.e.*, rapid reconfiguration entity 234) is implemented by both its backbone switches and its access switches. *Infra* §VII.A.4.f. Since Gai’s modified technique applies to backbone switches, which qualify as the claimed “edge switches” and “bridges,” the combination of Gai with IEEE 1999 renders obvious the ’986 patent’s claims as explained below. McNair, ¶145.

Petition at 30

Petitioner Explained How POSAs Would Have Modified Gai in View of IEEE-1999

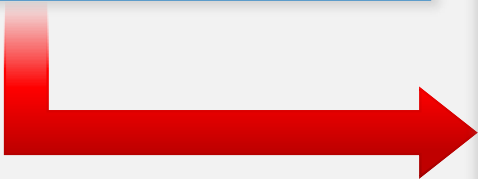
Petition at 27-28, 30
PO's Sur-Reply, Paper 23 at 1

Petition: How to Modify Gai

switches.” EX1005, 6:9-11. Gai discloses entering an Enable_Uplinkfast command at an access switch (e.g., access switch 114) to block a failed port (e.g., the root port) and configure the access switch to send dummy traffic upstream through a new active port. EX1005, 6:39-46. When modified in view of IEEE 1999’s disclosure that self-healing network traffic can advantageously be sent both upstream and downstream, the command would be entered at a backbone switch (e.g., backbone switch 122) to block a failed port (e.g., a designated port connected to access switch 114) and configure the backbone switch to send dummy traffic downstream to an access switch (e.g. access switch 114) through a new active port.

While Gai uses its protocol signaling command, Enable_Uplinkfast, to send dummy traffic through a new root port, POSAs would deem it an obvious implementation detail to modify the parameters of the command to be applicable to other ports, such as a designated port on a backbone switch. McNair, ¶139.

Petition at 27-28



PO's Sur-Reply



I. INTRODUCTION

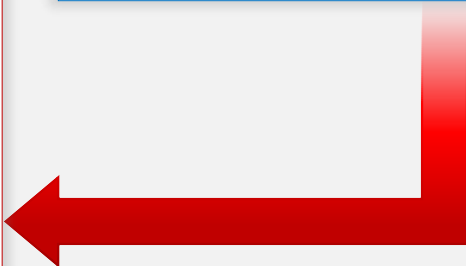
Petitioner’s Reply does not remedy the deficiencies of its Petition. Instead, Petitioner attempts to gloss over the issues raised in the Patent Owner Response (“POR”) and fails to provide a reason for *why* a POSA would be motivated to combine IEEE 1999 and Gai and/or an explanation for *how* a POSA would combine IEEE 1999 and Gai to render the challenged claims obvious. The Federal Circuit, however, has established that a Petitioner must provide evidence-supported reasoning for why and how references should be combined. Petitioner has failed to provide such reasoning. For the reasons in this Sur-Reply and in the POR, Petitioner failed to satisfy its burden of demonstrating invalidity of any of the challenged claims.

PO's Sur-Reply, Paper 23 at 1

Petition: Reasonable Expectation of Success

Finally, a POSA would have had a reasonable expectation of success in modifying Gai to have its backbone switches / edge switches / bridges send dummy traffic both upstream and downstream because signaling protocols are routinely modified to provide for different control signaling, such that there would have been no additional technical challenge in sending those messages both upstream and downstream. Further, Gai’s mechanism for sending dummy traffic (i.e., rapid reconfiguration entity 234) is implemented by both its backbone switches and its access switches. *Infra* §VII.A.4.f. Since Gai’s modified technique applies to backbone switches, which qualify as the claimed “edge switches” and “bridges,” the combination of Gai with IEEE 1999 renders obvious the ’986 patent’s claims as explained below. McNair, ¶145.

Petition at 30



The Petition's Gai Modification is Based on Directionality

Petition at 25, 29

IEEE-1999's Directionality

IEEE 1999 explains that switches both upstream and downstream of a link failure detect the failure. EX1009, 166 (“the fault would be independently detected by switches both upstream as well as downstream of the fault”). Each switch that detects a failure takes steps to rebuild the spanning tree. EX1009, 166 (“The fault recovery mechanism on each switch that detects the fault...[r]ebuild[s] the spanning tree”). As part of the spanning tree reconfiguration, to improve recovery time, each switch that detected a link failure sends “teardown messages” to both upstream and downstream switches to “terminate the connections through a failed link.” EX1009, 167. McNair, ¶¶101-102, 134.

Petition at 25

Gai Modified in view of IEEE-1999's Directionality Teachings

In short, when modified in view of IEEE 1999's disclosure that it was advantageous to send self-healing messages both upstream and downstream after a link failure either upstream or downstream, Gai's backbone switches (*i.e.*, the claimed “edge switches” and “bridges”) send Gai's dummy traffic both upstream and downstream, regardless of whether a link failure is detected upstream of the switch or downstream of the switch. This means that modified Gai's backbone switch / edge switch / bridge sends dummy traffic downstream to Gai's access switch / bridge, as recited in the challenged claims. Doing so achieves Gai's goal (shared by IEEE 1999) of reducing the time taken to rectify network failures, including between an edge switch and an access switch. EX1005, 11:53-55, 11:64-12:12. McNair, ¶144.

Petition at 29

PO Does Not Explain Why Gai Sending Dummy Messages Downstream Fails to Meet [1E]

POR at 10-11
PO's Sur-Reply, Paper 23 at 10
Reply, Paper 18 at 2-4

POR

Petitioner relies on IEEE 1999 to disclose the transmission of self-healing messages from an edge switch to one or more access switches and specifically identifies IEEE 1999's teardown messages as the appropriate message to modify filtering databases. *See e.g.*, Petition at 26-29, 46.¹ However, as previously mentioned in section II.B, IEEE 1999's teardown messages are sent to the sources of the connections (e.g. the hosts) and are not sent to modify filtering databases of any switches or reconfigure the spanning tree. *See supra* section II.A; EX2002, ¶ 55. IEEE 1999 instead teaches that teardown messages are sent to host computers to tear down existing connections. *See* EX2002, ¶ 55.

POR at 10-11

Reply

The POR Does Not Dispute that, If Gai's Dummy Messages Are Sent Downstream, Gai Meets Every Limitation of the Independent Claims

Reply, Paper 18 at 2-4

PO's Sur-Reply

Petitioner argues that "[t]he POR does not dispute that, if Gai's dummy messages are sent downstream, Gai meets every limitation of the independent claim." Reply at 2-3. But this point is irrelevant and incorrect. It is undisputed that Gai does not disclose sending dummy messages downstream. POR at 13-14. Nor does IEEE 1999 disclose sending dummy messages downstream. *Id.* The argument is also wrong, because even if Gai's messages are sent downstream (which they are not), Petitioner failed to meet its burden of showing that other recited limitations are met, such as the limitation "to cause the second bridge to modify its database responsive to the second port being in the active state." POR at 14. Moreover, the argument is misleading, because it assumes that the limitation of the independent claims can be met without providing the requisite motivation to combine. They cannot.

PO's Sur-Reply, Paper 23 at 10

Gai Modified in View of IEEE-1999 Meets Limitation [1E]

Petition at 40-42

'986 Patent, Claim 1

[1E]

sending dummy traffic from the edge switch over the second link to the one or more access switches,

so as to cause each of the one or more access switches to modify its respective database responsive to the second port being in the active state.

Petition (Gai Disclosures)

Gai's dummy multicast messages meet [1E]'s "dummy traffic."

Petition at 40

Gai discloses that dummy multicast messages can be sent by any switch.

Petition at 42

EX1005, 13:1-4. Gai discloses generating a dummy multicast message by "load[ing] SA field 412 with the address of an entity directly coupled to switch 114." EX1005, 13:14-16. The address of an entity directly coupled to switch 114

Petition at 41

source address of the dummy frame of the '986 patent. Gai also discloses loading the DA field "with a multicast address that causes message 400 to be received by all devices (e.g., access switches 114-116 and backbone switches 120-125) within the network 100." EX1005, 13:9-13. This matches the '986 patent, which says

Petition at 41

Petition Explaining How '986 Patent Matches Gai

of network 100"). Loading SA field 412 in this way matches the '986 patent, which describes "[f]or every entry in the filtering database of the edge switch that is not associated with the failed port...the edge switch sends a dummy frame...with a source MAC address equal to the destination MAC address listed for that entry." EX1001, 9:44-49. Put another way, the source MAC address of the dummy frame

Petition at 41

the network 100." EX1005, 13:9-13. This matches the '986 patent, which says "[t]he destination MAC address of the dummy frame is a multicast address or other special address selected to ensure that the dummy frame is flooded to all the switches downstream from the edge switch." EX1001, 9:53-56. McNair, ¶166.

Petition at 41

Gai Modified in View of IEEE-1999 Meets Limitation [1E]

Petition at 44

'986 Patent, Claim 1

[1E]

sending dummy traffic from the edge switch over the second link to the one or more access switches,

so as to cause each of the one or more access switches to modify its respective database responsive to the second port being in the active state.

Petition (Gai Disclosures)

algorithm.” EX1005, 5:65-6:5; 6:50-54. In Gai, when other devices within the network receive dummy multicast messages, they enter this new information in their respective filtering databases. EX1005, 13:34-38. This matches the '986

Petition at 44

Petition Explaining How '986 Patent Matches Gai

their respective filtering databases. EX1005, 13:34-38. This matches the '986 patent, which says “[u]pon receiving these dummy frames, the access switches use the standard MAC learning algorithm to rebuild their own filtering databases so that upstream traffic is immediately directed to the new active link,” purportedly completing network reconfiguration in less than one second. EX1001, 4:24-30. McNair, ¶171.

Petition at 44

No Dispute that Gai's Dummy Messages Cause Recipients to Update Their Databases

POR at 8
EX2002 at ¶47

POR

Gai's dummy messages are different. EX2002, ¶ 47. Gai discloses that dummy multicast messages are transmitted "[u]pon transition to the new forwarding port" and are sent so that "other devices within the network learn to utilize the new forwarding port, rather than the failed port, when directing messages to these entities." EX1005 at 5:54–6:5. Therefore, Gai is clear that its dummy multicast messages are transmitted as part of the reconfiguration process and only transmitted once a device has already successfully transitioned to a new forwarding port. *See*

POR at 8

PO Expert Declaration

describes dummy messages. Gai discloses that dummy multicast messages are transmitted "[u]pon transition to the new forwarding port" and are sent so that "other devices within the network learn to utilize the new forwarding port, rather than the failed port, when directing messages to these entities." EX1005 at 5:54–6:5. Therefore, Gai is clear that its dummy multicast messages are transmitted as part of the reconfiguration process and only transmitted once a device has already successfully transitioned to a new forwarding port. As such, Gai's dummy messages

EX2002 at ¶47

From PO: “a commonsense argument”

PO’s Sur-Reply

So here, Petitioner appears to proffer a commonsense argument that because IEEE 1999 messages are sent downstream, a POSA could simply send Gai’s dummy messages downstream too. Reply at 2-4. But the Federal Circuit has rejected similar

in the active state.” *Id.* at 14. At bottom, the record lacks “any explanation as to how or why [IEEE 1999’s disclosures regarding teardown messages and Gai’s sending of dummy messages upstream] would be combined to produce the claimed invention.” *Trivascular*, 812 F.3d at 1066. Accordingly, Petitioner fails to establish

PO’s Sur-Reply, Paper 23 at 11-12

How

Why

Petition

switches.” EX1005, 6:9-11. Gai discloses entering an Enable_Uplinkfast command at an access switch (e.g., access switch 114) to block a failed port (e.g., the root port) and configure the access switch to send dummy traffic upstream through a new active port. EX1005, 6:39-46. When modified in view of IEEE 1999’s disclosure that self-healing network traffic can advantageously be sent both upstream and downstream, the command would be entered at a backbone switch (e.g., backbone switch 122) to block a failed port (e.g., a designated port connected to access switch 114) and configure the backbone switch to send dummy traffic downstream to an access switch (e.g. access switch 114) through a new active port.

Petition at 27

Petition

With reference to downstream directionality, it would have been obvious to POSAs to send dummy multicast messages both upstream, as explained above, as well as downstream, for the benefit of reducing the total link failure recovery time because the link failure can be addressed by upstream and downstream switches in parallel. As explained above in §VII.A.3, POSAs would have been motivated, by IEEE 1999, to modify Gai to send dummy traffic downstream as well as upstream to achieve this benefit, and would have enjoyed a reasonable expectation of success in doing so. McNair, ¶173.

Petition at 45

Claim 6 Is Representative

'986 Patent, Claim 6

6. A method according to claim 1, wherein sending the dummy traffic comprises sending data frames originating at the edge switch while having source addresses corresponding to network addresses of other entities in the network that are accessible to the access switches via the edge switch.

EX1001 at Claim 6

PO Attacks References Individually for Dependent Claims 6 and 18

POR at 20
Petition at 48
Reply at 8

POR

EX2002, ¶ 73. Furthermore, because Gai itself does not disclose sending data frames originating at the edge switch while having source addresses corresponding to other network entities accessible to the access switches via the edge switch, a POSA would understand that Gai does not disclose claims 6 or 18. EX2002, ¶ 74. Furthermore, IEEE 1999 does not cure the deficiencies in Gai for the same reasons as previously discussed in sections II.A and II.B.1–2. See *supra* sections II.A and II.B.1–2; EX2002, ¶ 74.

POR at 20

Petition

As explained above in [1A] (*supra* §VII.A.4.b), backbone switch 122 is an edge switch. As explained above in [1E] (*supra* §VII.A.4.f), it would have been obvious for backbone switch 122 (*i.e.*, the edge switch) to send dummy multicast messages and thereby originate them. Such “dummy multicast messages carry the source address of each entity that is directly coupled to the device with the new active port or downstream thereof (relative to the root) and are received by other devices in the computer network.” EX1005, 5:54-60. Since such entities directly coupled to the device (*e.g.*, backbone switch 122) or downstream thereof are entities in the network and are accessible to access switches via the device, Gai’s backbone switch 122 sends the dummy traffic as claimed. McNair, ¶179.

Petition at 48

Petitioner’s Reply

through the new port”); EX1003, ¶169. In view of the motivation from IEEE 1999 to send dummy messages downstream as well as upstream upon detecting a link failure, Gai’s backbone switch would execute its built-in capability to do exactly that: send dummy messages both upstream and downstream. Petition, 45-46; ID, 35-36. The POR simply ignores this evidence and argument.

Reply at 8

PO Ignores Gai's Disclosure of "Frames" for Dependent Claims 6 and 18

PO's Sur-Reply, Paper 23 at 14
Petition at 53
EX1003 at ¶104

PO's Sur-Reply

not disclose sending data frames.” POR at 20. The Reply fails to identify any disclosure in the prior art teaching the “data frames” limitation of dependent claims 6 and 18. To the extent Petitioner is arguing that “dummy multicast messages [that]

PO's Sur-Reply, Paper 23 at 14

Petition

[access switch 114](#). Per the '986 patent, the claimed “dummy traffic” refers to dummy data frames. EX1001, 5:24-25, 12:51-52. Data frames are messages. EX1005, 11:16. Therefore, Gai's “dummy messages” are “dummy traffic” comprising “data frames,” meeting claim 11. McNair, ¶¶104, 188.

Petition at 53

Petitioner's Expert Declaration

104. The '986 patent claims use the phrase “dummy traffic.” The '986 patent specification uses the phrase “dummy message” as a synonym for “dummy traffic.” EX1001, 3:29-30 (“The switch then transmits dummy multicast messages toward the root...”). Claims 6, 11, 18 and 23 state that “dummy traffic comprises data frames.” Gai discloses “dummy multicast messages” (EX1005, 5:54-60) and also discloses that messages and frames are essentially synonymous. EX1005, 7:31-34. Accordingly, the dummy messages of Gai are synonymous with the dummy messages discussed in the '986 patent specification and the dummy traffic claimed in the '986 patent claims.

EX1003 at ¶104

Claims 4-5 Representative

'986 Patent, Claims 4-5

4. A method according to claim 1, and comprising, responsive to placing the second port in the active state, updating the database of the edge switch so that entries in the database point to the second port instead of the first port.

5. A method according to claim 4, wherein updating the database comprises changing the entries in the database to point to the second port before receiving the communication traffic from the one or more access switches on the second port.

EX1001 at Claims 4-5

An Overview

Petition

Gai and Jain are both directed to addressing link failures in spanning tree networks while reducing link failure recovery time. As discussed previously, Gai

Jain discloses managing forwarding databases (*i.e.*, filtering databases) in case of link failure. EX1006, Abstract, 6:52-54. Like Gai, Jain discloses detecting a failure and transitioning a backup port to take over for the failed port. EX1006, 5:15-18. To reduce recovery time, the forwarding databases are updated by transferring information from the failed port to the backup port, which is now active. The updates are made before the port transition to maintain high network availability. EX1006, 4:28-33; 5:15-20. McNair, ¶211.

The Motivation to Modify Gai

Petition

Gai expresses concern that messages may be misdirected or lost during the transition process. EX1005, 12:59-62, 15:18-21. POSAs would have been motivated to modify Gai by incorporating Jain's technique of updating a filtering database responsive to a link failure for the benefit of reducing the possibility of misdirected/lost data messages during the self-healing operation. As modified,

Petition at 63

The Modification to Gai

Petition

misdirected/lost data messages during the self-healing operation. As modified,

Gai's protocol entity would update the filtering database of a switch (including backbone switch 122) after detecting a failure at a port (*e.g.*, a downstream port).

The update would be implemented by remapping entries from the failed port to a backup port (*e.g.*, a back-up downstream port). The update would be implemented

before receiving communication traffic on the backup port to reduce the possibility of the backup port misdirecting or losing received messages using outdated

database information. The modification would result in a backbone switch's

protocol entity transitioning a failed port in the forwarding state to the blocking state and a backup port from the blocking state to the forwarding state, all with a

reduced possibility of data messages being misdirected or lost because the

backbone switch's database would be updated prior to notifying other network

switches of the newly-activated backup port. McNair, ¶214.

The POR Ignores the Petition's Argument

POR at 18

Petition at 63

Reply at 10

PO's Sur-Reply, Paper 23 at 15

Petition

database information. The modification would result in a backbone switch's protocol entity transitioning a failed port in the forwarding state to the blocking state and a backup port from the blocking state to the forwarding state, all with a reduced possibility of data messages being misdirected or lost because the backbone switch's database would be updated prior to notifying other network switches of the newly-activated backup port. McNair, ¶214.

Petition at 63

POR

during the self-healing operation.” Petition at 63. The Petition specifies that the modifications to Gai “would be implemented by remapping entries from the failed port to a backup port” and that the update “would be implemented before receiving communication traffic on the backup port to reduce the possibility of the backup port misdirecting or losing received messages using outdated database information.” *Id.* But Gai already describes a solution to this problem by sending dummy multicast messages. EX1005 at 12:62–66; EX2002, ¶ 50.

POR at 17-18

Gai's transmission of dummy multicast messages records the location information of the new root port and propagates that information to other devices in the system so that filtering databases can be updated. See EX2002, ¶ 51. As a result, Gai already provides its own solution so that messages are not lost or misdirected when a switch transitions to a new port. *Id.*

PO's Sur-Reply

Reply at 16. Moreover, Petitioner failed to show there was a motivation to combine, because, as Patent Owner argued, Gai already describes a solution to the problem Petitioner raised regarding why a POSA would be motivated to modify Gai in view of IEEE 1999 and further in view of Jain. POR at 17. Petitioner ignores this and instead uses impermissible hindsight to argue “the Petition argued that Jain's technique would have been obvious to use in the switch that sends Gai's dummy messages – as claims 4, 5, 16, and 17 require.” Reply at 10 (emphasis added). See

PO's Sur-Reply, Paper 23 at 15

Petitioner's Reply

However, the POR misses the mark because the disclosure in Gai to which it points pertains to updating the filtering database of the switch that receives Gai's dummy message, whereas the Petition argued that Jain's technique would have been obvious to use in the switch that sends Gai's dummy messages—as claims 4, 5, 16, and 17 require. Petition, 63 (“The modification would result in a

Reply at 10