

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

CISCO SYSTEMS, INC.,
Petitioner

IPR2023-00537
U.S. Patent No. 7,916,630

**PETITION FOR *INTER PARTES* REVIEW
UNDER 35 U.S.C. § 312 AND 37 C.F.R. § 42.104**

TABLE OF CONTENTS

Petitioner’s Exhibit List	5
I. Introduction.....	6
II. Grounds for standing	6
III. Note.....	7
IV. Summary of the ’630 patent	7
V. Prosecution history	11
VI. Effective priority date of the ’630 patent	12
VII. Level of ordinary skill in the art	13
VIII. Claim construction.....	13
A. “ <i>leasing method</i> ”	13
B. “ <i>Inform All method</i> ”	14
IX. Relief requested and the reasons for the requested relief.....	15
X. Identification of how the claims are unpatentable.....	15
A. Challenged Claims	15
B. Statutory ground for challenges	16
C. Ground 1	17
1. Silva.....	17
2. Claim 1	20
3. Claim 2.....	28
4. Claim 3	29

5.	Claim 4.....	32
6.	Claim 5.....	33
7.	Claim 6.....	33
8.	Claim 7.....	35
9.	Claim 8.....	37
10.	Claim 9.....	38
11.	Claim 10.....	39
12.	Claim 11.....	39
13.	Claim 12.....	40
14.	Claim 14.....	41
15.	Claim 15.....	43
XI.	Discretionary denial is inappropriate.....	43
A.	Discretionary denial under 35 U.S.C. § 325(d) is not appropriate	43
B.	Discretionary denial under the <i>Fintiv</i> factors is not appropriate.....	44
1.	Whether evidence exists that a stay may be granted if a proceeding is instituted	44
2.	Proximity of the court’s trial date to the board’s projected statutory deadline for a final written decision	45
3.	Investment in the parallel proceeding.....	46
4.	Overlap between issues raised in the petition and in the parallel proceeding.....	46
5.	Whether the petitioner is a defendant	47

6.	Other circumstances that impact the Board’s exercise of discretion, including the merits.....	48
C.	Discretionary denial under <i>General Plastic</i> is not appropriate.....	48
XII.	Conclusion	49
XIII.	Mandatory notices	50
A.	Real party-in-interest.....	50
B.	Related matters	50
C.	Lead and back-up counsel and service information.....	50
	Certificate of Word Count	52
	Certificate of Service	53

PETITIONER'S EXHIBIT LIST

Ex.1001	U.S. 7,916,630
Ex.1002	Prosecution History of U.S. 7,916,630
Ex.1003	Declaration of Nader F. Mir, Ph.D., under 37 C.F.R. § 1.68
Ex.1004	<i>Curriculum Vitae</i> of Nader F. Mir, Ph.D.
Ex.1005	Silva et al., “DIP: Distributed Diagnosis Protocol,” <i>Microprocessing and Microprogramming</i> 38 (1993)
Ex.1006	[Reserved]
Ex.1007	Declaration of Ingrid Hsieh-Yee, Ph.D., under 37 C.F.R. § 1.68
Ex.1008	Docket Control Order, <i>Lionra Techs. Ltd. v. Cisco Sys., Inc.</i> , No. 2:22-cv-00305, Paper 55-1 (E.D. Tex. Dec. 1, 2022)
Ex.1009	U.S. District Courts – Federal Court Management Statistics – Comparison Within Circuit – During the 12-Month Period Ending June 30, 2022, available at https://www.uscourts.gov/statistics/table/na/federal-court-management-statistics/2022/06/30-3
Ex.1010	Complaint, <i>Lionra Techs. Ltd. v. Cisco Sys., Inc.</i> , No. 2:22-cv-00305 (E.D. Tex. Aug. 8, 2022)

I. INTRODUCTION

Pursuant to 35 U.S.C. §§ 311, 314(a), and 37 C.F.R. § 42.100, Cisco Systems, Inc. (“Petitioner”) respectfully requests that the Board review and cancel as unpatentable under (pre-AIA) 35 U.S.C. §103(a) claims 1-12, 14, and 15 (hereinafter, the “Challenged Claims”) of U.S. 7,916,630 (the “’630 Patent,” Ex.1001).

The ’630 Patent describes monitoring the condition of a network having distributed components arranged in a logical ring structure, where each component monitors only its respective neighboring component. Ex.1001, Abstract. As shown below and confirmed in the Declaration of Nader F. Mir, Ph.D. (Ex.1003) these concepts were known in the prior art well before the priority date of the ’630 Patent. For example, *Silva* (Ex.1005) teaches a diagnosis protocol to check the status of distributed network nodes arranged in a virtual ring, where each node monitors only its respective neighbor. Therefore, the references presented in this Petition teach and render obvious the Challenged Claims.

II. GROUNDS FOR STANDING

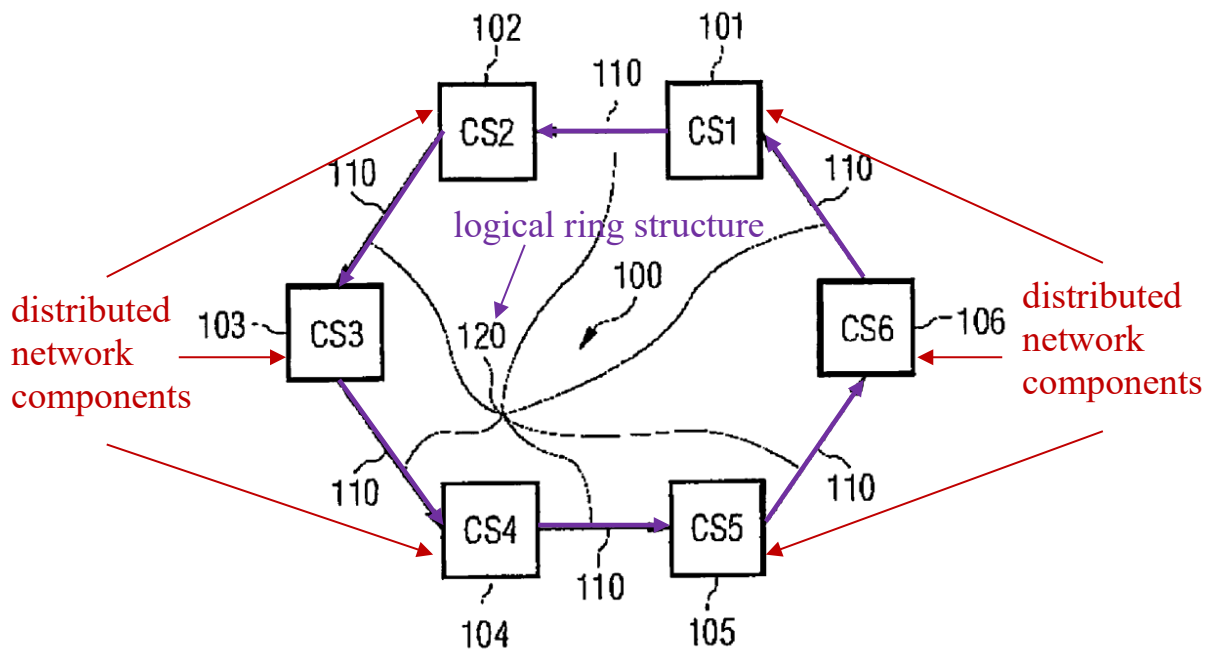
Petitioner certifies that the ’630 Patent is eligible for IPR and that Petitioner is not barred or estopped from requesting IPR challenging the patent claims. 37 C.F.R. § 42.104(a).

III. NOTE

Petitioner cites to exhibits' original page numbers. **Emphasis** in quoted material has been added. Claim terms are presented in *italics*.

IV. SUMMARY OF THE '630 PATENT

The '630 Patent relates generally to monitoring a “system condition of a system with distributed components.” Ex.1001, 1:14-17. According to the '630 Patent, distributed network components may be “organized in a logical ring structure.” Ex.1001, 2:36-39. This “logical” ring structure is “a conceptual model, in a virtual ring structure,” as opposed to a “physical concrete structuring of the system components into a ring structure.” Ex.1001, 2:36-45. Figure 1 of the '630 Patent shows distributed network components, such as communication servers (101-106), that are arranged in a logical ring structure (120):



Ex.1001 at Fig. 1 (annotated)

According to the '630 Patent, “there is as a rule a requirement for all the components of the distributed system to have a knowledge of the status or condition of every other component in the system (monitoring).” Ex.1001, 1:18-23. To reduce the volume of system monitoring messages, “a component only monitors its respective neighboring component in the logical ring structure, whereby a condition of the respective neighboring component can be determined.” Ex.1001, 3:8-11. Figure 1 “shows a distributed system with (distributed) components organized in a logical ring structure, in which each component only monitors its respective neighbor in the ring, according to an exemplary embodiment.” Ex.1001, 4:48-51.

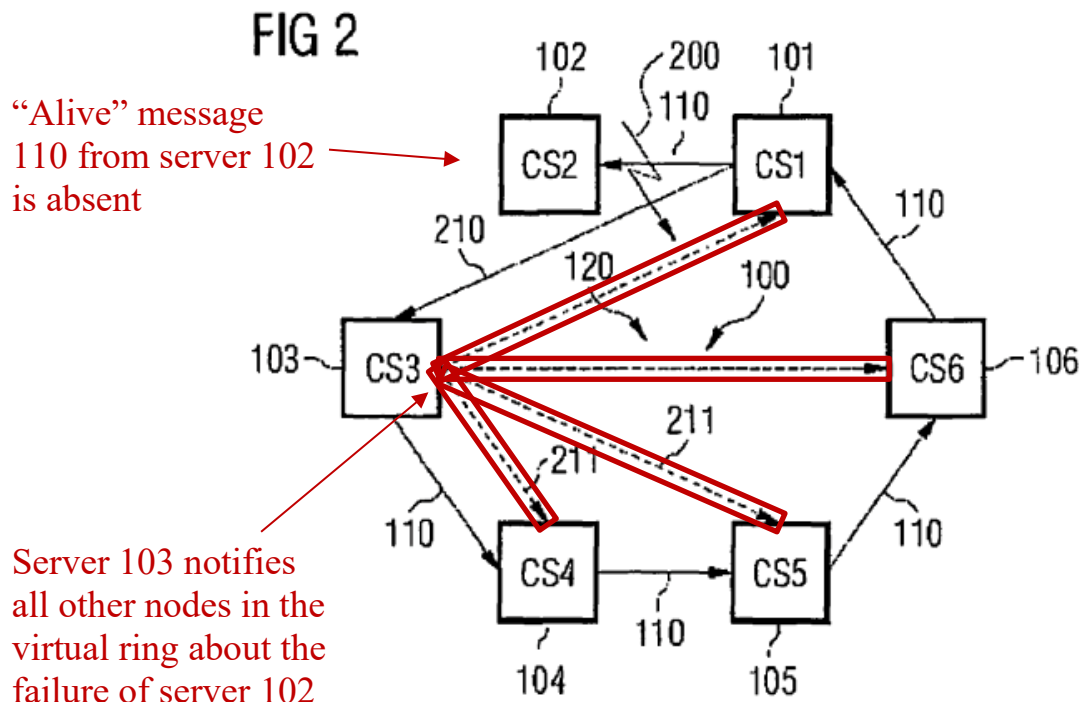
The neighboring component “can be a predecessor or also a successor in the logical ring structure.” Ex.1001, 4:30-32.

Each network component may monitor its predecessor or successor for a “predefinable condition,” Ex.1001, 3:12-16, such as “a functional incapacity, in particular an offline condition, or a functional capacity, in particular an online condition.” Ex.1001, 3:38-41. “The monitoring of the respective neighboring component and/or the determination of the condition of a component can preferably be carried out using a method based on a leasing method.” Ex.1001, 3:43-46. Specifically, “‘Alive’ information, particularly an ‘Alive’ message, can be sent or is sent from the neighboring component to the component” monitoring it. Ex.1001, 3:47-50. “On the basis of such ‘Alive’ information, which may be periodic but is not necessarily so, it is then possible to determine a functional incapacity of the neighboring component if the neighboring component does not send (no longer sends) ‘Alive’ information. This can clearly mean that – on the other hand – as long as ‘Alive’ information is being sent, an ‘online’ condition can be assumed for the respective component.” Ex.1001, 3:54-61.

If a predefinable condition is determined to exist, the network component that detected the condition may inform the other components using a method based on an “Inform All” method. Ex.1001, 3:62-65. “Here, in other words with regard to the method based on an ‘Inform All’ method applied, an acknowledgment process is

performed for each other component in the system, in which the respective other component, if it has received the information about the predefinable condition of a neighboring component, confirms the receipt of the information, in particular by using ‘Acknowledgment’ information, specifically an ‘Acknowledgment’ message.” Ex.1001, 3:66-4:6. If another component does not acknowledge the receipt of the information, “the predefinable condition is also determined for [the] another component.” Ex.1001, 4:14-18.

Figure 2, shows an embodiment of the invention where successor component 102 periodically sends an “Alive” message 110 to the preceding component 103 in the logical ring. Ex.1001, 5:30-33. “If in the case of one communication server (103 in FIG. 2) this ‘Alive’ message 110 from its successor (102 in FIG. 2) is absent (the ‘lease’ expires), the successor (102 in FIG. 2) will be categorized as ‘offline’.” Ex.1001, 5:34-37. “The communication server 103 monitoring this ‘offline’ communication server 102 then informs every other communication server 101, 104 to 106 in the system 100 about this by corresponding information messages 211, in other words about the failure 200 of its successor 102.” Ex.1001, 5:38-42.



Ex.1001 at Fig. 2 (annotated)

“Every other communication server 101, 104 to 106 in the system 100 must confirm the receipt of the information by an ‘Acknowledgement’ message.” Ex.1001, 5:43-45. “If this acknowledgment is missing from one of the other components 101, 104 to 106, then this communication server will also be categorized as ‘offline’.” Ex.1001, 5:46-48.

V. PROSECUTION HISTORY

The application which led to the '630 Patent was filed internationally on June 17, 2005, as Application No. PCT/EP2005/052828. Ex.1002, 2. During national stage prosecution at the USPTO, Applicant filed a Request for Continued

Examination with claim amendments to overcome rejections for anticipation and obviousness over admitted prior art (APA) and Ogier, noting that the independent claims recite “each component in the system monitoring only a single respective neighboring component among said distributed components that is a predecessor or successor of said each component in the logical ring structure” and “informing all other components of the system about the current condition of the respective neighboring component.” Ex.1002, 273-278. After a clarifying examiner’s amendment was entered, the claims were allowed. Ex.1002, 300-304. The ’630 Patent issued on March 29, 2011. Ex.1002, 315.

VI. EFFECTIVE PRIORITY DATE OF THE ’630 PATENT

The ’630 Patent is based on the PCT application filed June 17, 2005, and claims priority to a patent application filed in Germany on September 16, 2004. The US national phase filing of the ’630 Patent was on January 18, 2008. Determining whether the ’630 Patent is entitled to an earlier priority date is unnecessary in this proceeding, however, as all of the cited and relied-upon prior art predates September 2004. Accordingly, Petitioner has not undertaken a priority date analysis. Petitioner does not waive any right or opportunity it may have to dispute the priority date of the ’630 Patent in this or another forum where the issue is relevant.

VII. LEVEL OF ORDINARY SKILL IN THE ART

A Person of Ordinary Skill in The Art (“POSITA”) in September 2004 would have had a working knowledge of the network monitoring art that is pertinent to the ’630 Patent, including monitoring system conditions of a network with distributed components. A POSITA would have had a bachelor’s degree in computer science, computer engineering, electrical engineering, or equivalent training, and approximately two years of experience working in the field of communication networks. Additional work experience can substitute for specific educational background, and vice versa. Ex.1003, ¶¶19-21.

VIII. CLAIM CONSTRUCTION

This Petition analyzes the claims consistent with the ordinary and customary meaning as would be understood by a POSITA in light of the specification. *See* 37 C.F.R. § 42.100(b). Petitioner submits that, for the purposes of this proceeding and the grounds presented herein, no claim terms require express construction. *Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co.*, 868 F.3d 1013, 1017 (Fed. Cir. 2017). For clarity, however, Petitioner notes below example embodiments in the specification of certain terms.

A. “*leasing method*”

Claim 3 recites determining a current condition of a network component based on a “*leasing method*.” The ’630 Patent describes one example of a “leasing”

method in which a component “periodically sends a so-called ‘Alive’ message” to its neighboring communication server in the ring. Ex.1001, 5:34-37. If the ‘Alive’ message “is absent (the ‘lease’ expires),” the neighbor will be categorized as offline. Ex.1001, 5:34-37, 3:43-61. If the ‘Alive’ message is sent, “an ‘online’ condition can be assumed for the respective component.” Ex.1001, 3:54-61. To the extent Patent Owner would seek to read this example into the claim language, the references cited in this Petition teach this limitation, as outlined in the analysis below. Ex.1003, ¶¶73-80.

B. “*Inform All method*”

Claim 7 recites that the step of informing all other components about a neighboring component’s respective condition using an “*Inform All*” method. The ’630 Patent provides an example where (1) a neighboring node is categorized as “offline” when an “Alive” message is not received (Ex.1001, 5:34-37), (2) the node monitoring the “offline” node then informs each of the other nodes in the system about this failure by corresponding information messages (Ex.1001, 5:38-42), (3) each of the other nodes in the system confirms receipt of the information by an “Acknowledgment” message (Ex.1001, 5:43-45, *see also* 3:66-4:8), (4) if the acknowledgment is missing from a node, then that node is also categorized as “offline” (Ex.1001, 5:46-48, *see also* 4:14-18), (5) each of the other nodes are in turn informed about this further failure by the node monitoring the “offline” node

(i.e., the monitoring node in step (2) described above) (Ex.1001, 5:49-50, *see also* 4:19-20), and (6) after notifying each of the other nodes about the further failure, the node monitoring the “offline” node (i.e., the monitoring node in step (2) described above) may – or may not – expect any further acknowledgment (Ex.1001, 5:51-52, *see also* 4:20-21). To the extent Patent Owner would seek to read this example into the claim language, the references cited in this Petition teach this limitation, as outlined in the analysis below. Ex.1003, ¶¶94-104.

IX. RELIEF REQUESTED AND THE REASONS FOR THE REQUESTED RELIEF

Petitioner asks that the Board institute a trial for *inter partes* review and cancel the Challenged Claims in view of the analysis below.

X. IDENTIFICATION OF HOW THE CLAIMS ARE UNPATENTABLE

A. Challenged Claims

Petitioner challenges claims 1-12, 14 and 15 of the '630 Patent. The Challenged Claims (1-12, 14, and 15) include (a) claims that Patent Owner has asserted in the co-pending litigation (1-8, 14, and 15), and (b) other claims (9-12) that are also rendered unpatentable by the same prior art as the claims in (a). A finding that the Challenged Claims (1-12, 14, and 15) are unpatentable in this proceeding will resolve the parties' dispute in the co-pending litigation and obviate

any need for a trial regarding the '630 Patent, substantially reducing the time and expense of that litigation for all parties.

B. Statutory ground for challenges

Ground	Claims	Basis
#1	1-12, 14, and 15	35 U.S.C. § 103 (pre-AIA) over Silva

Silva et al., “DIP: Distributed Diagnosis Protocol,” *Microprocessing and Microprogramming* 38 (1993) (Ex.1005, “Silva”) was published in 1993. See Ex.1007, ¶49. The journal, *Microprocessing and Microprogramming*, was intended for researchers with an interest in fields such as “Distributed systems” and “communications and networks.” Ex.1007, App.1005A, 4. Therefore, Silva is prior art under pre-AIA 35 U.S.C. §§ 102(a) and 102(b).¹

Silva was not before the Examiner during prosecution of the '630 Patent.

Silva is analogous art to the '630 Patent because it is at least (1) in the same field of endeavor (e.g., network monitoring); or (2) reasonably pertinent to a problem with which the inventor of the '630 Patent was involved (e.g., how to monitor the condition of components in a network, including one that utilizes a

¹ In the event that Patent Owner argues that the '630 Patent is an AIA patent, Petitioner notes that Silva would still qualify as prior art under post-AIA 35 U.S.C. §§ 102(a) and 102(b).

logical ring topology). *Compare* Ex.1001, 1:36-39 (“monitoring a system condition of a system with distributed components, the distributed components of the system are or will be organized in a logical ring structure”) *with* Ex.1005, 172 (“Detection of Failures” of network nodes “organized on a virtual ring”).

C. Ground 1

1. Silva

Silva relates to a “DIP - a Diagnosis Protocol to check the availability of processors in a distributed system.” Ex.1005, 171. Like the ’630 Patent, “network nodes are organized on a virtual ring and all the sites are numbered from 0 up to N-1, along the path of the virtual ring.” Ex.1005, 172. The virtual ring “does not reflect the physical organization of the network” but follows the logical “order of the Status Table, which is unique to all the processors in the network.” Ex.1005, 172. Figure 1 of Silva shows distributed nodes that are arranged in a virtual ring:

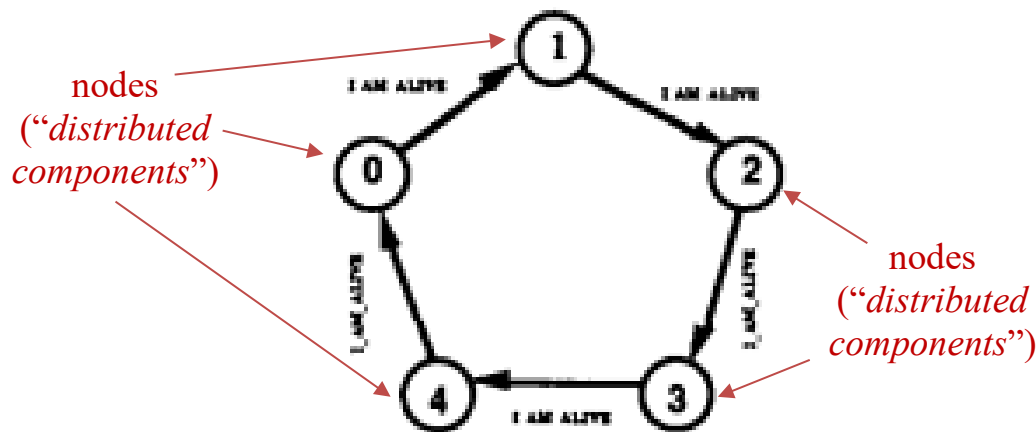


Figure 1 : Virtual Ring

Ex.1005 at Fig. 1 (annotated); Ex.1003, ¶44

According to Silva, “[e]ach node maintains a table with the status of the other nodes, called the Network Status Table (NST).” Ex.1005, 172. Like the ’630 Patent, “to reduce the overhead during normal operation,” Silva teaches a Distributed Diagnosis Protocol that requires that each node monitor its respective neighbor in the ring. Ex.1005, 172. In particular, “[e]ach processor P_i sends periodically an I_AM_ALIVE message to the next neighbor $P_{((i+1)\text{mod}N)}$ while it is checking for the messages sent by the previous processor $P_{((i-1)\text{mod}N)}$.” Ex.1005, 172. “Each processor has two independent processes to perform that job. 1) The *Heartbeat* process is periodically awakened by a timer interrupt in order to send a message to the next neighbor ... 2) The detection of failures is done by the Watchdog process, which is listening to the messages sent by the previous neighbor.” Ex.1005, 172-173.

In particular, a node detects failures of its previous neighbor by listening for its I_AM_ALIVE messages. “Every time it receives an I_AM_ALIVE message it triggers a timer T_W with $((\Omega+1).\alpha + \partial_{\max} - \partial_{\min})$, where Ω is the number of omission failures or lost messages that can be tolerated. If the timer expires, this means that the previous node has crashed and the Node Down Protocol is started for that node.”

Ex.1005, 172-173.

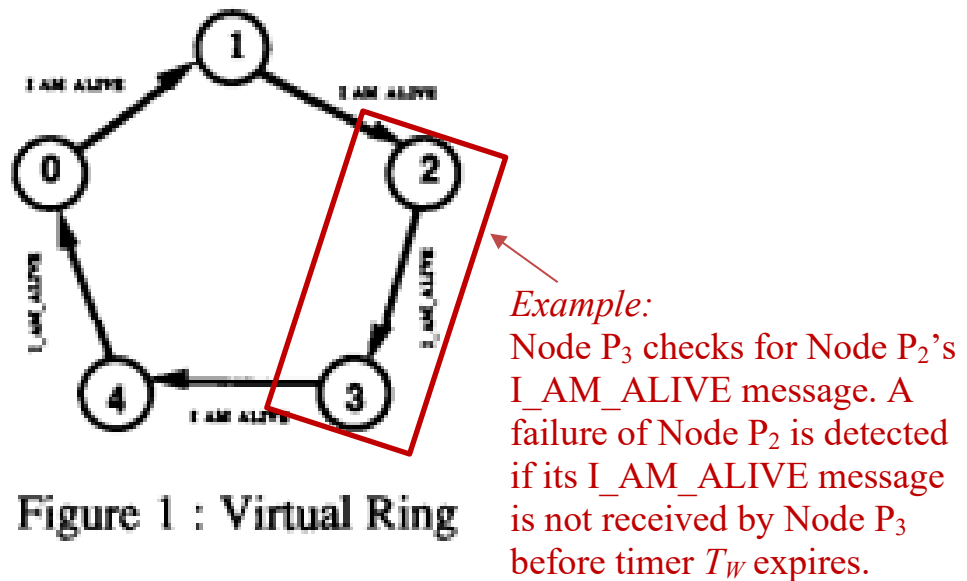
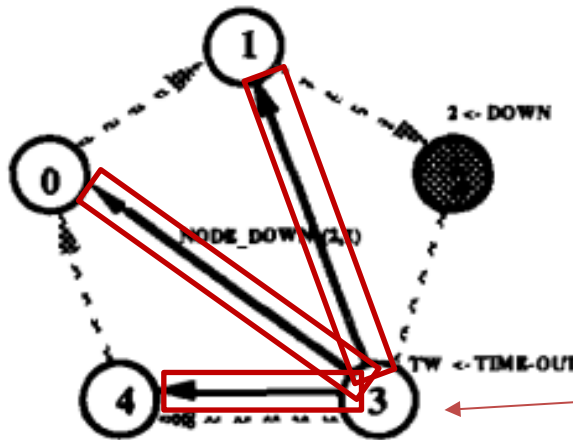


Figure 1 : Virtual Ring

Ex.1005 at Fig. 1 (annotated)

Like the '630 Patent, if a node determines a failure, “the node should notify all the others immediately by a broadcast message to the network, announcing the Node_Down event of its neighbour.” Ex.1005, 173. “Take as an example the network of figure 2. One can notice that node #2 has crashed and its tester (node #3) will detect that when its timer T_W expires. The status of node #2 is marked as

DOWN in its Network Status Table, and it will send a broadcast message to the network, which includes the number of the node and the NST(2).IN: NODE_DOWN (2,1).” Ex.1005, 173.



Example:
Node P₃ detects Node P₂ has crashed, and broadcasts a NODE_DOWN (2,1) message to the other nodes in the network.

Figure 2 : Detection of a Failure

Ex.1005 at Fig. 2 (annotated)

“[W]hen a node receives a NODE_DOWN broadcast it replies with a ND_Ack message to node #3.” Ex.1005, 173. In certain cases, “some nodes may not answer” the NODE_DOWN broadcast message with a ND_Ack message, such as in the case of a network partition. Ex.1005, 173. Nodes that do not answer the NODE_DOWN broadcast with an acknowledgement message “should also be classified as DOWN.” Ex.1005, 173.

2. Claim 1

[1.0] *A method for monitoring a system condition of a network with distributed*

components organized in a logical ring structure, comprising:

To the extent the preamble is limiting, Silva renders it obvious.

First, Silva teaches a “DIP: Distributed Diagnosis Protocol” for detecting “node crashes” and “node failure[s]” in a network. Ex.1005, Title, Abstract, 172. Accordingly, Silva’s diagnosis protocol teaches the claimed “*method for monitoring a system condition.*” Ex.1003, ¶46.

Second, Silva’s nodes may be “remote” from each other and “interconnected by a LAN” where “each [node] can communicate with any other [node] in the network.” Ex.1005, 171-172; Ex.1003, ¶47. Figure 1 depicts remote network nodes communicating across a network by sending I_AM_ALIVE messages:

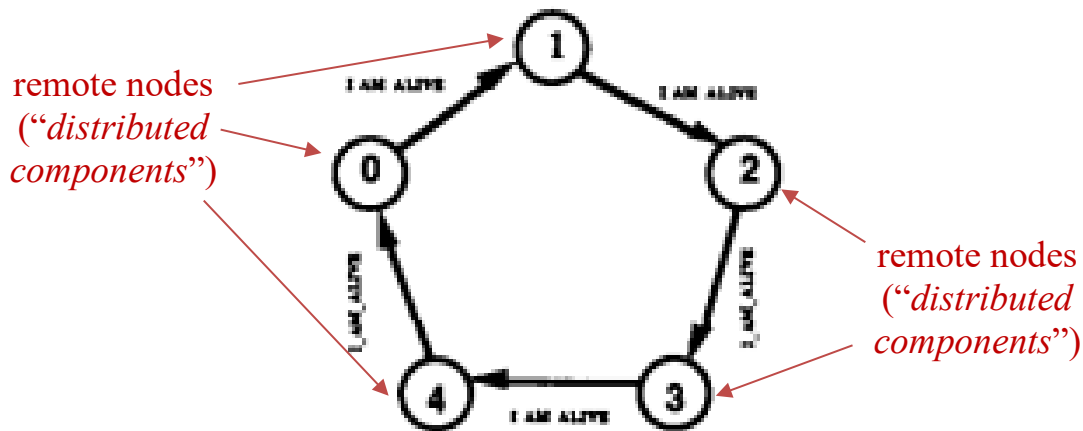


Figure 1 : Virtual Ring

Ex.1005, Fig. 1 (annotated); Ex.1003, ¶48

Accordingly, Silva teaches the claimed “*network with distributed components.*”

Ex.1003, ¶¶47-48.

Third, Silva teaches that “[t]he network nodes are organized on a **virtual ring** and all sites are numbered from 0 up to N-1, along the path of the virtual ring. It should be noted that the ring does not reflect that physical organization of the network.” Ex.1005, 172; Ex.1003, ¶49. Figure 1 depicts the remote nodes arranged in a virtual ring structure:

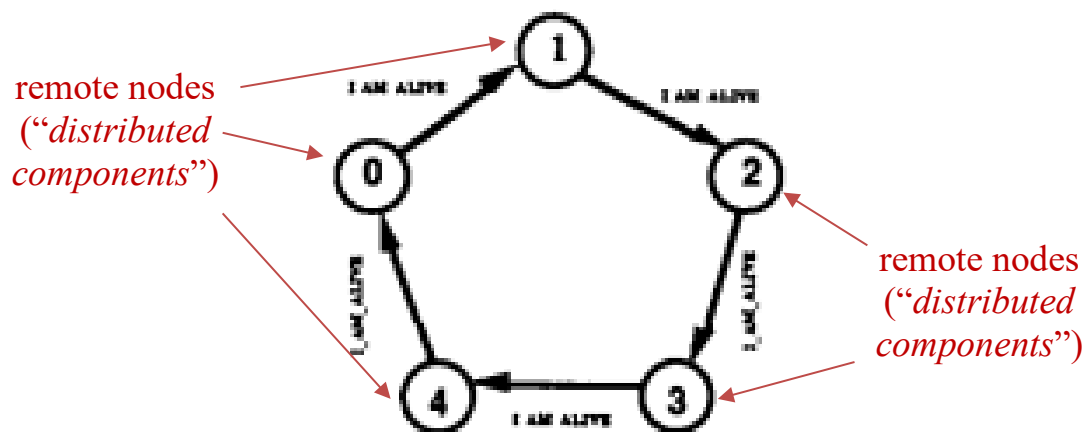


Figure 1 : Virtual Ring

Ex.1005, Fig. 1 (annotated); Ex.1003, ¶50

A POSITA would have understood Silva’s “virtual ring” structure to be a “*logical ring structure*” because it reflects the logical organization of nodes in the ring, rather than the physical organization of the network. Ex.1003, ¶51. Further, a POSITA would have understood Silva’s “virtual ring” structure to be a “*logical ring structure*” because one example of a logical ring structure provided by the

'630 Patent is “a virtual ring structure.” Ex.1001, 2:40-45; Ex.1003, ¶52.

Accordingly, Silva teaches the claimed distributed components organized in “a *logical ring structure*.” Ex.1003, ¶¶49-52.

Thus, Silva’s disclosure of a distributed diagnosis protocol for detecting crashes or failures of remote network nodes organized in a virtual ring, renders obvious a “*method for monitoring a system condition of a network with distributed components organized in a logical ring structure*.” Ex.1003, ¶53.

[1.1] each component in the system monitoring only a single respective neighboring component among said distributed components that is a predecessor or successor of said each component in the logical ring structure to determine a current condition of the respective neighboring component; and

First, Silva teaches that each node in the virtual ring monitors only its neighboring predecessor node (“*a single respective neighboring component*”) in the virtual ring for an I_AM_ALIVE message. In particular, Silva teaches that “[e]ach processor P_i sends periodically an I_AM_ALIVE message to the next neighbour $P_{((i+1)\bmod N)}$ while it is **checking for the messages sent by the previous processor $P_{((i-1)\bmod N)}$** .” Ex.1005, 172; Ex.1003, ¶55. This checking is done by a “Watchdog process” in each node “which is **listening to the messages sent by the previous neighbour**.” Ex.1005, 172-173; Ex.1003, ¶57.

A POSITA would understand that the expression, $P_{((i-1)\bmod N)}$, refers to the network node immediately preceding network node P_i . In particular, the variable

“i” indicates the identity of a particular node, the variable “N” indicates the total number of nodes in the virtual ring, and “mod” is the mathematical operation, modulo, that indicates the remainder after dividing (i-1) by N. Ex.1005, 172. Thus, for example, in a virtual ring with a total of 5 nodes, network node P_2 monitors its neighboring predecessor P_1 because $P_{((2-1) \bmod 5)} = P_1$. Ex.1003, ¶56

Figure 1 depicts how network node P_i checks for an I_AM_ALIVE message from only its neighboring predecessor node $P_{((i-1) \bmod N)}$. Ex.1005, 172, Ex.1003, ¶57. For example, in Figure 1, network node P_2 checks for an I_AM_ALIVE message sent from only network node P_1 .

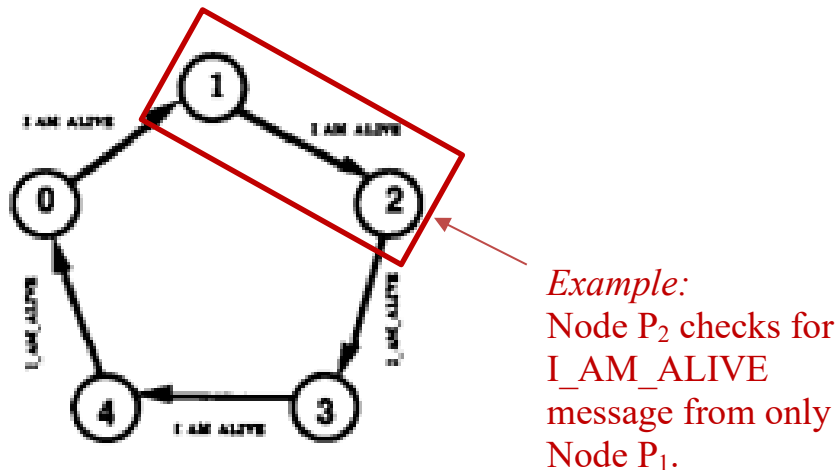


Figure 1 : Virtual Ring

Ex.1005, Fig. 1 (annotated); Ex.1003, ¶57

Accordingly, Silva teaches the claimed “each component in the system monitoring only a single respective neighboring component among said distributed components that is a predecessor or successor of said each component in the

logical ring structure.” Ex.1003, ¶54-57.

Second, Silva teaches that each node in the virtual ring determines whether its neighboring predecessor node is down or alive (“*determine a current condition*”) using the Watchdog process to detect the absence or presence of an I_AM_ALIVE message. “[T]he Watchdog process ... listen[s] to the messages sent by the previous neighbor. Every time it receives an I_AM_ALIVE message it triggers a timer T_W ” Ex.1005, 172-173. “If the timer expires, this means that the previous node has crashed and the **Node Down Protocol** is started for that node.” Ex.1005, 173. Conversely, if an I_AM_ALIVE message is received from the neighboring predecessor node before the timer expires, the timer is reset and the Node Down Protocol is not started because the node is alive. Ex.1005, 172-173; Ex.1003, ¶58. Accordingly, Silva teaches the claimed “*to determine a current condition of the respective neighboring component.*” Ex.1003, ¶58.

Therefore, Silva’s disclosure of each node monitoring only its neighboring predecessor node in the virtual ring for the absence or presence of an I_AM_ALIVE message, to detect whether a node is down or alive using a Watchdog process, renders obvious “*each component in the system monitoring only a single respective neighboring component among said distributed components that is a predecessor or successor of said each component in the logical ring structure to determine a current condition of the respective*

neighboring component.” Ex.1003, ¶59.

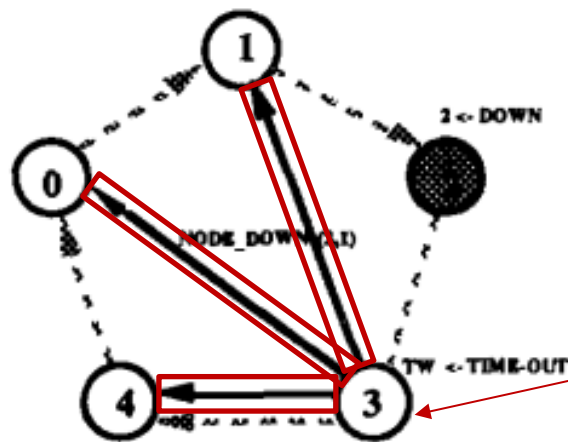
[1.2] each component in the system informing all other components of the system about the current condition of the respective neighboring component when the current condition corresponds to at least one predefined condition.

First, as discussed in [1.1], Silva teaches that a node determines its neighboring predecessor node is down when its Watchdog process does not receive an I_AM_ALIVE message from the neighboring predecessor node before a timer, T_W , expires. This condition is referred to as a “Node Down” condition (“*at least one predefined condition*”). Ex. 1005, 172-173, Ex.1003, ¶61.

Second, a POSITA would understand that Silva’s Node Down condition is “*predefined*” because its requirements are known prior to detecting a node is actually down. Ex.1003, ¶62. As discussed in [1.1], a Node Down condition requires that no I_AM_ALIVE message is received before “timer T_W expires.” Ex.1005, 172-173; Ex.1003, ¶¶62-63.

Third, Silva teaches that when a node detects a Node Down condition (“*the current condition*”) of its neighboring predecessor node, it initiates a Node Down Protocol to notify all other nodes in the virtual ring of the Node Down condition. Upon detection of the Node Down condition, **“the node should notify all others immediately by a broadcast message to the network, announcing the Node_Down event of its neighbour.”** Ex.1005, 173; Ex.1003, ¶64. Figure 2 depicts an example of the Node Down Protocol, where a node (i.e., node P_3) has

detected the failure of its neighboring predecessor node (i.e., node P_2), and immediately notifies all other nodes in the virtual ring (i.e., nodes P_0 , P_1 , and P_4) about the Node Down condition by broadcasting a “NODE_DOWN (2,1)” message:

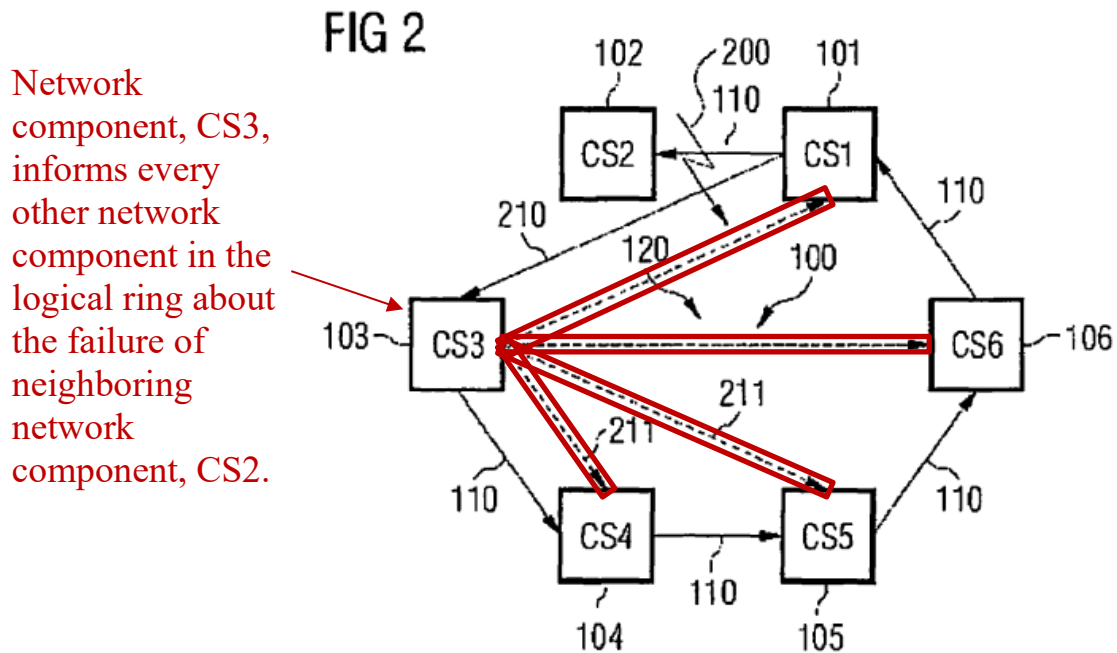


Example:
Node P_3 notifies all other nodes in the virtual ring about the Node Down condition of Node P_2 by broadcasting a NODE_DOWN (2,1) message.

Figure 2 : Detection of a Failure

Ex.1005 at Fig. 2 (annotated); Ex.1003, ¶65

The broadcast message (e.g., NODE_DOWN (2,1)) sent to each node in Silva is much like the information message (211) sent to each network component described in the '630 Patent to “inform every other” network component that a failure is detected. *See* Ex.1001, 5:38-42, Ex.1003, ¶66. Figure 2 depicts an example provided in the '630 Patent of a network component (i.e., CS3) notifying all other network components (i.e., CS1, CS4, CS5, and CS6) about the failure of its neighboring networking component (i.e., CS2).



Ex.1001 at Fig. 2 (annotated), Ex.1003, ¶66

Therefore, Silva's disclosure of a Watchdog process in each node to detect a Node Down condition of its neighboring predecessor node, and initiating a Node Down Protocol that broadcasts a message to notify all other nodes about the Node Down condition, renders obvious "*each component in the system informing all other components of the system about the current condition of the respective neighboring component when the current condition corresponds to at least one predefined condition.*" Ex.1003, ¶67.

3. Claim 2

[2.0] *The method as claimed in claim 1, wherein the at least one predefined condition is at least one of a functional incapacity corresponding to an offline condition and a functional capacity corresponding to an online condition.*

As discussed in [1.2], Silva teaches a Watchdog process for detecting a Node Down condition (“*predefined condition*”). The Node Down condition is a node failure or crash condition (“*functional incapacity corresponding to an offline condition*”): “[t]he **detection of failures** is done by the Watchdog process, which is listening to the messages sent by the previous neighbour.” Ex.1005, 173. If an I_AM_ALIVE message is not received before “the timer expires, this means that **the previous node has crashed** and the Node Down Protocol is started for that node.” Ex.1005, 173, Ex.1003, ¶69. A POSITA would have understood a node “failure” and a node “crash” are each examples of “*an offline condition*,” because a node that has failed or crashed is unavailable to other nodes for communication. Ex.1003, ¶70.

Therefore, Silva’s teaching of a Node Down condition that indicates a node failure or crash renders obvious, “*wherein the at least one predefined condition is at least one of a functional incapacity corresponding to an offline condition and a functional capacity corresponding to an online condition.*” Ex.1003, ¶¶71-72.

4. Claim 3

[3.0] The method as claimed in claim 2, wherein at least one of said monitoring of the respective neighboring component and determination of the current condition of the respective neighboring component is carried out based on a leasing method.

First, as discussed in [1.1], Silva teaches that each node listens for I_AM_ALIVE messages from only its neighboring predecessor node (“*said monitoring of the respective neighboring component*”) to determine whether or not the predecessor node has failed (“*said determination of the current condition of the respective neighboring component*”). Ex.1003, ¶74.

Second, Silva teaches that each node uses a “Heartbeat process” to send I_AM_ALIVE messages “periodically” to its neighboring successor node, and each node uses a “Watchdog process” to listen for these messages from its neighboring predecessor node. Ex.1005, 172-173; Ex.1003, ¶75. As discussed further below, the combination of the Heartbeat process and the Watchdog process render obvious the claimed “*leasing method*.”

In particular, “[e]ach **processor P_i sends periodically an I_AM_ALIVE message to the next neighbour $P_{((i+1)\bmod N)}$** while it is checking for the messages sent by the previous processor $P_{((i-1)\bmod N)}$. Each processor has two independent processes to perform that job. 1) The **Heartbeat process** is periodically awakened by a timer interrupt in order to send a message to the next neighbor.” Ex.1005, 172. “2) The detection of failures is done by the **Watchdog process**, which is listening to the messages sent by the previous neighbor.” Ex.1005, 173. Every time an “I_AM_ALIVE” message is received from a neighboring predecessor node, a Watchdog timer T_W is triggered. Ex.1005, 173. “**If the timer expires, this means**

that the previous node has crashed and the Node Down Protocol is started for that node.” Ex.1005, 173. However, if an I_AM_ALIVE message is received before the timer has expired, no crash or failure is detected. Ex.1005, 173; Ex.1003, ¶76.

Thus, like the leasing method described in the '630 Patent, if an I_AM_ALIVE message is not received or “is absent” from the neighboring predecessor node for a certain period, the “lease” expires and the neighboring node will be categorized as down. *See* Ex.1001, 5:34-37. If an I_AM_ALIVE message is received from the neighboring predecessor node, then the lease does not expire, no crash or failure is detected, and an “online” condition can be assumed. *See* Ex.1001, 3:58-61; Ex.1003, ¶¶77-78. Accordingly, a POSITA would have recognized that the interaction between the Heartbeat process in each node for sending I_AM_ALIVE messages, and the Watchdog process in a neighboring successor node for listening for these messages to determine whether its neighboring predecessor node has crashed or not, teaches a “*leasing method*.” Ex.1003, ¶79.

Therefore, Silva’s disclosure of a Heartbeat process in each node for sending I_AM_ALIVE messages periodically to a neighboring successor node, and a Watchdog process in each neighboring successor node to listen for these messages (“*monitoring of the respective neighboring component*”) before a leasing period

expires, to determine whether or not a respective predecessor node has failed (*“determination of the current condition of the respective neighboring component”*), renders obvious *“wherein at least one of said monitoring of the respective neighboring component and determination of the current condition of the respective neighboring component is carried out based on a leasing method.”* Ex.1003, ¶80.

5. Claim 4

[4.0] *The method as claimed in claim 3, wherein with regard to the leasing method, an “Alive” message is sent from the respective neighboring component.*

As discussed in [1.1] and [3.0], each node uses a “Heartbeat process” to send “I_AM_ALIVE” messages periodically to its neighboring successor node, and each node uses a “Watchdog process” to listen for these messages from its neighboring predecessor node to determine whether it is down or alive. Ex.1005, 172-173; Ex.1003, ¶82. Silva’s teaching of such I_AM_ALIVE messages is much like the example of “Alive” messages described in the ’630 Patent that may be sent periodically from a network component to its neighbor for determining whether or not the network component is offline. Ex.1001, 5:34-37; Ex.1003, ¶83. Therefore, Silva’s disclosure of each node sending its neighboring successor node an “I_AM_ALIVE” message renders obvious *“wherein with regard to the leasing*

method, an ‘Alive’ message is sent from the respective neighboring component.”

Ex.1003, ¶84.

6. Claim 5

[5.0] *The method as claimed in claim 4, wherein the “Alive” information is sent periodically.*

As discussed in [4.0], Silva teaches that each node, “sends **periodically** an I_AM_ALIVE message to the next neighbour.” Ex.1005, 172-173. “The Heartbeat process is periodically awakened by a timer interrupt in order to send a message to the next neighbour.” Ex.1005, 172-173; Ex.1003, ¶86. Therefore, Silva renders obvious, “*method as claimed in claim 4, wherein the ‘Alive’ information is sent periodically.*” Ex.1003, ¶87.

7. Claim 6

[6.0] *The method as claimed in claim 5, wherein the functional incapacity of the neighboring component is determined if the respective neighboring component does not send any “Alive” information for a predetermined period of time.*

First, as described in [1.2], Silva teaches that each node’s Watchdog process detects a Node Down condition (“*current condition*” is “*functional incapacity*”) when an I_AM_ALIVE message (“*any ‘Alive’ information*”) from its neighboring predecessor node is not received before Watchdog Timer, T_W , expires. Ex.1003, ¶89.

Second, Silva teaches that the time period of Watchdog Timer, T_W , is calculated according to the mathematical equation $T_W = ((\Omega+1).\alpha + \partial_{\max} - \partial_{\min})$. Ex.1005, 173; Ex.1003, ¶90. According to Silva, the parameter α (i.e., value of heartbeat interrupt timer, T_H) and the parameter Ω (i.e., number of omission failures or lost messages that can be tolerated) are each values “chosen” or “set” as part of the distributed diagnosis protocol (DIP). Ex.1005, 172, 173, 177; Ex.1003, ¶91. A POSITA would also understand that the parameter $\partial_{\max}$ (i.e., maximum message delay between two nodes) and the parameter $\partial_{\min}$ (i.e., minimum message delay between two nodes) are each operational characteristics of the virtual ring network itself. Ex. 1005, 172; Ex.1003, ¶92. A POSITA would have thus recognized that the parameters, α , Ω , $\partial_{\max}$, and $\partial_{\min}$, are determined or set before calculating the value of the Watchdog Timer, T_W . Ex.1003, ¶92. Thus, the time period of Watchdog Timer, T_W , is “predetermined.”

Therefore, Silva’s teaching of each node’s Watchdog process detecting a Node Down condition when an I_AM_ALIVE message from its neighboring predecessor node is not received before Watchdog Timer, T_W , expires, renders obvious “*wherein the functional incapacity of the neighboring component is determined if the respective neighboring component does not send any ‘Alive’ information for a predetermined period of time.*” Ex.1003, ¶93.

8. Claim 7

[7.0] *The method as claimed in claim 6, wherein said informing all the other components about the predefined condition of the respective neighboring component is carried out using an “Inform All” method.*

As discussed in VIII.B, the '630 Patent provides an example of an “Inform All” method where (1) a neighboring node is categorized as “offline” when an “Alive” message is not received, (2) the node monitoring the “offline” node then informs each of the other nodes in the system about this failure by corresponding information messages, (3) each of the other nodes in the system confirms receipt of the information by an “Acknowledgment” message, (4) if the acknowledgment is missing from a node, then that node is also categorized as “offline,” (5) each of the other nodes are in turn informed about this further failure by the node monitoring the “offline” node (i.e., the monitoring node in step (2) described above), and (6) after notifying each of the other nodes about the further failure, the node monitoring the “offline” node (i.e., the monitoring node in step (2) described above) may – or may not – expect any further acknowledgment. To the extent Patent Owner would seek to read this example into the claim language, the references cited in this Petition teach this limitation, as outlined in the analysis below.

First, as discussed in [1.2] and [2.0], Silva teaches that each node’s Watchdog process detects a Node Down condition (“*predefined condition*”)

corresponding to an “*offline condition*” when an I_AM_ALIVE message from its neighboring predecessor node is not received before Watchdog Timer, T_w , expires. Ex.1003, ¶95.

Second, as discussed in [1.2], Silva teaches that each node, upon detection of a Node Down condition of its neighboring predecessor node, initiates a Node Down Protocol to broadcast a “NODE_DOWN” message to each of the other nodes in the virtual ring to notify them about the failure. Ex.1003, ¶96.

Third, Silva teaches that each node that receives a “NODE_DOWN” message replies with an “acknowledge” (e.g. “ND_Ack”) message. Ex.1005, 173; Ex.1003, ¶97.

Fourth, Silva teaches that if an “acknowledge” message is not received by the node that broadcasted the “NODE_DOWN” message before a timer, T_{ack} , expires, the node whose acknowledgment is missing “should also be classified as DOWN.” Ex.1005, 173; Ex.1003, ¶98.

Fifth, Silva teaches that after the node that broadcasted the “NODE_DOWN” message classifies a node whose acknowledgment is missing as DOWN, the node that broadcasted the “NODE_DOWN” message “announc[es] the unavailability” of the DOWN node to each of the other nodes. Ex. 1005, 173; Ex.1003, ¶99.

Sixth, it would have been an obvious implementation choice whether the node that “announc[es] the unavailability” of the DOWN node to each of the other nodes may – or may not – expect any further acknowledgment messages. Silva teaches that the node announcing the unavailability of a DOWN node will “**start again**” the “protocol announcing the unavailability of those nodes” classified as DOWN. Ex.1005, 105, Ex.1003, ¶100. Thus, a POSITA would have found an obvious implementation would be to repeat or “start again” the protocol announcing the unavailability of nodes that do not send acknowledgment messages to the broadcasted “NODE_DOWN” message, and to listen for subsequent acknowledgments to those announcements to detect other DOWN nodes. Ex.1003, ¶¶101-102.

Therefore, Silva renders obvious, “*wherein said informing all the other components about the predefined condition of the respective neighboring component is carried out using an ‘Inform All’ method.*” Ex.1003, ¶¶103-104.

9. Claim 8

[8.0] The method as claimed in claim 7, further comprising, with regard to the “Inform All” method, performing an acknowledgment process by each of the other components, in which each of the other components, if it has received information about the predefined condition of the respective neighboring component, confirms receipt of the information, by sending an “Acknowledgment” message.

As discussed in [7.0], Silva teaches that each node that receives a “NODE_DOWN” message replies with an “acknowledge” (e.g. “ND_Ack”) message. Ex.1005, 173. A POSITA would understand that an “acknowledge” and an “ND_Ack” message are each examples of an “Acknowledgment” message because they are sent to confirm receipt of the “NODE_DOWN” message. Ex.1003, ¶106.

10. Claim 9

[9.0] *The method as claimed in claim 8, wherein the “Acknowledgment” message is sent to the component which has determined the predefined condition of the respective neighboring component.*

As described in [7.0], Silva teaches that each node that receives a “NODE_DOWN” message replies with an “acknowledge” (e.g. “ND_Ack”) message. Ex.1005, 173. In particular, the “acknowledge” message is sent to the node that sent the NODE_DOWN message. Ex.1005, 173; Ex.1003, ¶108. For example, “when a node receives a NODE_DOWN broadcast [from node #3] **it replies with a ND_Ack message to node #3.**” Ex.1005, 173.

Therefore, Silva’s disclosure that a node that receives a NODE_DOWN message from the notifying node (e.g. node #3), sends an acknowledgment message to that node (e.g. node #3), renders obvious “*wherein the ‘Acknowledgment’ message is sent to the component which has determined the predefined condition of the respective neighboring component.*” Ex.1003, ¶109.

11. Claim 10

[10.0] *The method as claimed in claim 9, further comprising determining the predefined condition for any component which does not acknowledge the receipt of the information about the predefined condition of the respective neighboring component.*

As described in [7.0], Silva teaches that if an “acknowledge” message is not received by the node that broadcast the “NODE_DOWN” message before a timer, T_{ack} , expires, the node whose “acknowledge” message is missing “**should also be classified as DOWN.**” Ex.1005, 173; Ex.1003, ¶111; *see also supra*, limitation [1.2] (identifying “Node Down” as the “*predefined condition*”). Therefore, Silva renders obvious the “*method as claimed in claim 9, further comprising determining the predefined condition for any component which does not acknowledge the receipt of the information about the predefined condition of the respective neighboring component.*” Ex.1003, ¶112.

12. Claim 11

[11.0] *The method as claimed in claim 10, further comprising in each component storing a local list of conditions of the other components.*

Silva teaches that “[e]ach node maintains a table with the status of the other nodes, called the Network Status Table (NST). Each entry of the NST has three fields: the host name, its status and the Incarnation Number (IN).” Ex.1005, 172. The Network Status Table is included in a node’s “stable storage” and read into “main memory.” Ex.1005, 173-174; Ex.1003, ¶114.

Therefore, Silva's disclosure that each node stores in its own storage or memory, a table with the status of other nodes, renders obvious "*in each component storing a local list of conditions of the other components.*" Ex.1003, ¶115.

13. Claim 12

[12.0] *The method as claimed in claim 11, wherein the respective neighboring component is one of a predecessor component and a successor component in the logical ring structure.*

As discussed in [1.1], Silva teaches that each node in the virtual ring monitors only its neighboring predecessor node. In particular, each processor " P_i " checks for I_AM_ALIVE messages "**sent by the previous processor $P_{((i-1) \bmod N)}$.**" Ex.1005, 172; Ex.1003, ¶117. Figure 1 depicts how each node P_i (e.g., node P_2) monitors only its neighboring predecessor node $P_{((i-1) \bmod N)}$ (e.g., node P_1) for an I_AM_ALIVE message:

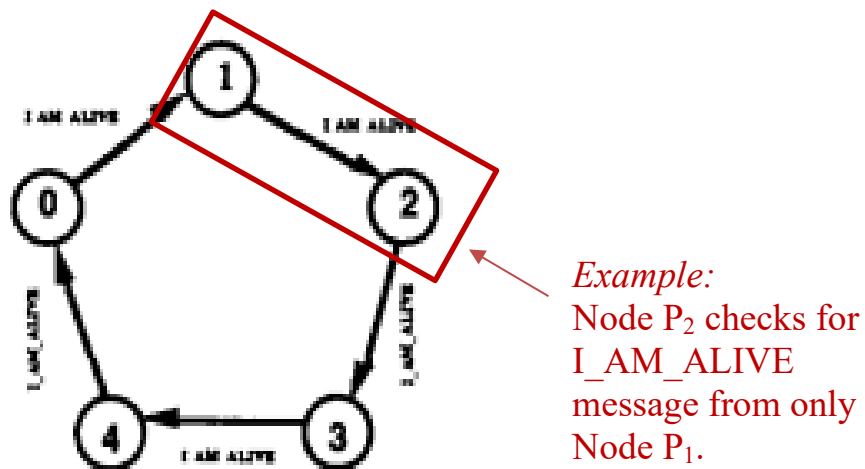


Figure 1 : Virtual Ring

Ex.1005 at Fig. 1 (annotated), Ex.1003, ¶118

Therefore, Silva’s disclosure of a node monitoring its neighboring predecessor node by listening for its I_AM_ALIVE messages renders obvious “wherein the respective neighboring component is one of a predecessor component and a successor component in the logical ring structure.” Ex.1003, ¶119.

14. Claim 14

[14.0] At least one non-transitory computer-readable medium encoded with at least one computer program that when executed by at least one processor performs a method comprising:

To the extent the preamble is limiting, Silva renders it obvious.

Silva teaches that the distributed diagnosis protocol is “implemented on a set of SUN workstations” and runs “as a background process.” Ex.1005, 177. In particular, the distributed diagnosis protocol is “written in C” programming

language and the workstations each include a “processor” and “stable storage.” Ex.1005, 171, 177; Ex.1003, ¶121. A POSITA would have found it obvious that Silva’s distributed diagnosis protocol written in C code would be stored in the stable storage of each network device such that when executed by the processor, carries out the operations of the distributed diagnosis protocol, because it was common for computer software written in C to be compiled into an executable file that is then stored on persistent computer media, such as a SUN workstation’s hard drive. Ex.1003, ¶122.

Therefore, Therefore, Silva renders obvious, “[a]t least one non-transitory computer-readable medium encoded with at least one computer program” executed by a processor. Ex.1003, ¶123.

[14.1] *each component in a logical ring structure*

See analysis at [1.0].

[14.2] *monitoring only a single neighboring component among distributed components that is a predecessor or successor of said each component in the logical ring structure to determine a current condition of the neighboring component; and*

See analysis at [1.1].

[14.3] *each component in the logical ring structure informing all other components of the system about the current condition of the neighboring component when the current condition corresponds to a predefined condition.*

See analysis at [1.2].

15. Claim 15

[15.0] *A network with distributed components, comprising:*

To the extent the preamble is limiting, Silva renders it obvious. See analysis at [1.0].

[15.1] *components organized in a logical ring structure,*

See analysis at [1.0].

[15.2] *each component monitoring only a single respective neighboring component among components that is a predecessor or successor of said each component in the logical ring structure to determine a current condition of the respective neighboring component and*

See analysis at [1.1].

[15.3] *informing all other components of said network when the current condition of the respective neighboring component corresponds to a predefined condition.*

See analysis at [1.2].

XI. DISCRETIONARY DENIAL IS INAPPROPRIATE

A. Discretionary denial under 35 U.S.C. § 325(d) is not appropriate

Denial under § 325(d) is not warranted because the challenges presented in this petition are neither cumulative nor redundant to the prosecution of the '630 Patent. None of the asserted prior art references were cited or considered by the Examiner during prosecution of the '630 Patent, thus the references in this petition

were neither relied upon during prosecution nor are cumulative of art relied upon during prosecution. *See* Ex.1002.

B. Discretionary denial under the *Fintiv* factors is not appropriate

The Board balances six factors in considering denial under § 314. *Apple Inc. v. Fintiv, Inc.*, IPR2020-00019, Paper 11 (Mar. 20, 2020) (precedential). These factors strongly favor institution. The district court case is in its very early stages. Patent Owner filed its complaint on August 8, 2022. Patent Owner provided its disclosure of asserted claims and infringement contentions on November 3, 2022. Ex.1008, 5. Invalidity contentions are not due until February 10, 2023. Ex.1008, 4. Petitioner has since worked diligently to prepare this Petition, which is being filed well within the one-year timeframe allowed by Congress.

1. Whether evidence exists that a stay may be granted if a proceeding is instituted

If an IPR is instituted, a stay of the co-pending litigation would be appropriate. However, given that a motion to stay has not yet been filed, the Board should not infer or attempt to predict the outcome if such a motion is later filed. *Sand Revolution II LLC v. Continental Intermodal Group – Trucking LLC*, IPR2019-01393, Paper 24 at 7 (June 16, 2020) (informative); *see also Dish Network L.L.C. v. Broadband iTV, Inc.*, IPR2020-01359, Paper 15 (Feb. 12, 2021)

(“It would be improper to speculate, at this stage, what the Texas court might do regarding a motion to stay...”). Thus, this factor is neutral.

2. Proximity of the court’s trial date to the board’s projected statutory deadline for a final written decision

This factor weighs strongly against discretionary denial because the projected trial date – based on median time-to-trial statistics – is in August of 2024, which is contemporaneous with (within a range measured in weeks) the expected timing of the Board’s Final Written Decision.² While trial is currently scheduled for May 20, 2024 (Ex.1008, 1), the Board recognizes “that scheduled trial dates are unreliable and often change.” *See* Director’s June 21, 2022 Memorandum on Discretionary Denials (“Memo”), 8. “The PTAB will weigh this factor [factor 2] against exercising discretion to deny institution under *Fintiv* if the median time-to-trial is around the same time or after the projected statutory deadline for the PTAB’s final written decision.” Memo, 9.

The co-pending district court case was filed in the Eastern District of Texas on August 8, 2022. *See* Ex.1010. The most recent statistics show a median time-to-trial in the Eastern District of Texas as 24.5 months. Ex.1009, 5. Accordingly, the

² August 2024 is 18 months after February 2023, when Petitioner expects a notice of accorded filing date for this petition.

projected trial date for purposes of *Fintiv* is August 2024 – approximately 24 months after August 2022, and contemporaneous with the Board’s Final Written Decision expected in August 2024. Because the projected trial date is “around the same time or after” the Board’s expected final written decisions, this factor weighs in favor of institution.

3. Investment in the parallel proceeding

The co-pending litigation is in its early stages, and the investment in it has been minimal. The parties have not exchanged proposed claim terms or constructions, and will not complete claim construction briefing until October 27, 2023. Ex.1008, 2. The *Markman* hearing is not until November 17, 2023. Ex.1008, 3. Fact discovery – which has not yet opened – is not set to close until December 8, 2023, and expert discovery is not set to close until February 5, 2024. Ex.1008, 3; *see PEAG LLC v. Varta Microbattery GMBH*, IPR2020-01214, Paper 8 at 17 (Jan. 6, 2021) (finding that since no claim construction hearing had yet been held and discovery was not completed, the little investment in the parallel proceeding weighed against discretionary denial).

Accordingly, the lack of investment in the parallel district court proceeding favors institution.

4. Overlap between issues raised in the petition and in the

parallel proceeding

There is no overlap of prior art issues at this time. Invalidity contentions are not due until February 10, 2023. Ex.1008, 4. Instituting a proceeding will allow the Board to address the asserted prior art, and the issues will be narrowed in the co-pending litigation due to the estoppel provisions of 35 U.S.C. § 315(e)(2).

Moreover, there will be no overlap of prior art issues. As noted above in section X.A., Petitioner is challenging claims not asserted in the pending litigation. Thus, there will be no overlap of prior art issues with regard to those claims.

Additionally, if the Board institutes trial, Petitioner will cease asserting in the co-pending litigation the combination of references on which trial is instituted for the claims on which trial is instituted, to the extent Petitioner even asserts the same combination in the district court. *See Verizon v. Huawei*, IPR2020-01079, Paper 10 at 38 (finding a similar stipulation “mitigates the concern about overlapping issues” and weighs “against discretionary denial of the Petition”).

Accordingly, this factor weighs against discretionary denial.

5. Whether the petitioner is a defendant

Petitioner is a defendant in the litigation. Ex.1008. This is true of most Petitioners in IPR proceedings. While the *Fintiv* case indicates that a difference between the district court defendant and the petitioner may weigh against discretionary denial, nothing within the *Fintiv* case suggests that the same party

between the proceedings weighs in favor of discretionary denial. *See HP Inc. v. Slingshot Printing LLC*, IPR2020-01084, Paper 13 at 9 (having the “same parties as parallel proceeding” makes factor 5 “neutral”). Accordingly, this factor is neutral and should not be a basis for denying institution.

6. Other circumstances that impact the Board’s exercise of discretion, including the merits

As discussed in detail above, the prior art presented in this Petition renders the Challenged Claims unpatentable as obvious. The merits of Petitioner’s arguments are compelling because they lead to a strong conclusion that one or more claims are unpatentable by a preponderance of the evidence. This factor weighs against discretionary denial.

As such, because the *Fintiv* factors are either neutral or weigh against discretionary denial, and because this Petition was filed approximately six months before the statutory bar date, institution should not be denied on discretionary factors.

C. Discretionary denial under *General Plastic* is not appropriate

The ’630 Patent has not been challenged in any prior IPR petition, so none of the *General Plastic* discretionary institution factors apply to this Petition. *See General Plastic Indus. Co., Ltd. v. Canon Kabushiki Kaisha*, IPR2016-01357, Paper 19 at 16 (Sept. 6, 2016) (Section II.B.4.i. precedential).

XII. CONCLUSION

Petitioner has established a reasonable likelihood that the Challenged Claims are unpatentable.

Respectfully submitted,

Dated: February 7, 2023
HAYNES AND BOONE, LLP
2323 Victory Avenue, Suite 700
Dallas, Texas 75219
Customer No. 27683

/Theodore M. Foster/
Theodore M. Foster
Lead Counsel for Petitioner
Registration No. 57,456

XIII. MANDATORY NOTICES

A. Real party-in-interest

Pursuant to 37 C.F.R. § 42.8(b)(1), Petitioner certifies that the real party-in-interest is Cisco Systems, Inc.

B. Related matters

Pursuant to 37 C.F.R. § 42.8(b)(2), to the best knowledge of the Petitioner, the '630 Patent is or was involved in the following cases:

Case Heading	Number	Court	Filed
<i>Lionra Techs. Ltd. v. Cisco Systems, Inc.</i>	2:22-cv-00305	EDTX	Aug. 8, 2022
<i>Lionra Techs. Ltd. v. Hewlett Packard Enterprise Co. et al.</i>	2:22-cv-00319	EDTX	Aug. 19, 2022

C. Lead and back-up counsel and service information

Lead Counsel

Theodore M. Foster
HAYNES AND BOONE, LLP
2323 Victory Ave. Suite 700
Dallas, TX 75219

Phone: (303) 382-6205
Fax: (214) 200-0853
ipr.theo.foster@haynesboone.com
USPTO Reg. No. 57,456

Back-up Counsel

Brian C. Kwok
HAYNES AND BOONE, LLP
2323 Victory Ave. Suite 700
Dallas, TX 75219

Phone: (415) 293-8916
Fax: (214) 200-0853
brian.kwok.ipr@haynesboone.com
USPTO Reg. No. 58,828

David L. McCombs
HAYNES AND BOONE, LLP
2323 Victory Ave. Suite 700
Dallas, TX 75219

Phone: (214) 651-5533
Fax: (214) 200-0853
david.mccombs.ipr@haynesboone.com
USPTO Reg. No. 32,271

Please address all correspondence in this proceeding to lead and back-up counsel. Petitioner consents to service in this proceeding by email at the addresses above.

CERTIFICATE OF WORD COUNT

Pursuant to 37 C.F.R. § 42.24(d), Petitioner hereby certifies, in accordance with and reliance on the word count provided by the word-processing system used to prepare this Petition, that the number of words in this paper is 8,075. Pursuant to 37 C.F.R. § 42.24(d), this word count excludes the table of contents, table of authorities, mandatory notices under § 42.8, certificate of service, certificate of word count, appendix of exhibits, and any claim listing.

Dated: February 7, 2023

/Theodore M. Foster/
Theodore M. Foster
Lead Counsel for Petitioner
Registration No. 57,456

CERTIFICATE OF SERVICE

The undersigned certifies that, in accordance with 37 C.F.R. § 42.6(e) and 37 C.F.R. § 42.105, service was made on Patent Owner as detailed below.

Date of service February 7, 2023

Manner of service FEDERAL EXPRESS

Documents served Petition for *Inter Partes* Review Under 35 U.S.C. § 312 and 37 C.F.R. § 42.104 of U.S. 7,916,630;
Petitioner's Exhibit List;
Exhibits 1001-1005, & 1007-1010.

Persons served Staas & Halsey LLP
1201 New York Avenue, N.W.
Suite 700
Washington, DC 20005

/ Theodore M. Foster /
Theodore M. Foster
Lead Counsel for Petitioner
Registration No. 57,456