

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

SAP AMERICA, INC.
Petitioner,

v.

VALTRUS INNOVATIONS LTD. AND KEY PATENT INNOVATIONS LTD.
Patent Owner.

Case No. IPR2025-00414
U.S. Patent No. 7,152,182

**PETITION FOR *INTER PARTES* REVIEW
OF U.S. PATENT NO. 7,152,182**

TABLE OF CONTENTS

	<u>Page</u>
I. Introduction.....	17
II. Compliance with IPR Requirements	17
A. Grounds for Standing	17
B. Mandatory Notices	17
C. Fee Authorization	19
III. The Challenged Claims	20
IV. The '182 Patent.....	20
A. Earliest Effective Filing Date.....	20
B. Level of Ordinary Skill in the Art.....	20
C. Overview of the '182 Patent.....	21
V. Claim Construction.....	23
VI. Principal Prior Art.....	24
A. <u>Lubbers</u> (Ex-1005)	24
B. <u>Sicola</u> (Ex-1006).....	27
C. <u>Ofek</u> (Ex-1007).....	31
VII. All Claims Are Unpatentable	32
A. Ground 1: Claims 1-8 Would Have Been Obvious Over <u>Lubbers</u> and <u>Sicola</u>	34
1. Claim 1	39
a. [1preamble]: “A data redundancy system, comprising:”	39
b. [1a]: “a primary storage facility for storing a primary copy of data, the primary storage facility including;”	40
c. [1b]: a first redundancy appliance operable in a role for receiving a sequence of write requests and for storing data for the sequence of write requests in storage associated with the primary storage facility; and”	43

d.	[1c]: “a second redundancy appliance for shadowing the first redundancy appliance wherein the second redundancy appliance assumes the role of the first redundancy appliance;”	45
e.	[1d]: “a secondary storage facility for storing data that is redundant of the primary copy of the data, the secondary storage facility discrete from the primary storage facility and including;”	48
f.	[1e]: “a third redundancy appliance operable in a role for storing redundant data for the sequence of write requests in storage associated with the secondary facility; and”	50
g.	[1f]: “a fourth redundancy appliance for shadowing the third redundancy appliance wherein the fourth redundancy appliance assumes the role of the third redundancy appliance in the event of a fault at the third redundancy appliance.”	52
2.	Claim 2:	54
a.	“The data redundancy system according to claim 1, wherein the secondary storage facility assumes the role of the primary storage facility in response to a failover event,”	54
b.	“the failover event including a communication traffic condition having a greater portion of request traffic originating closer to the secondary storage facility than to the primary storage facility.”	54
3.	Claim 3: <i>The data redundancy system according to claim 1, wherein the second redundancy appliance assumes the role of the first redundancy appliance in response to a failover event selected from the group consisting of a manually initiated event and an automatically initiated event.</i>	56
4.	Claim 4:	57
a.	“The data redundancy system according to claim 1, wherein the first redundancy appliance updates the second redundancy appliance with information about the data in storage by forwarding a write	

	record to the second redundancy appliance for each write request,”	57
b.	“the second redundancy appliance sending acknowledgements to the first redundancy appliance in batches in response to receiving write records.”	58
5.	Claim 5: <i>The data redundancy system according to claim 4, wherein the first redundancy appliance sends an acknowledgement to an originator of the write request in response to receiving the acknowledgement from the second redundancy appliance.</i>	60
6.	Claim 6: <i>The data redundancy system according to claim 1, wherein the first redundancy appliance is disjoint from the second redundancy appliance.</i>	61
7.	Claim 7: <i>The data redundancy system according to claim 1, wherein the second redundancy appliance maintains a shadow copy of data stored by the first redundancy appliance wherein the second redundancy appliance accesses the shadow copy of the data when the second redundancy appliance assumes the role of the first redundancy appliance.</i>	63
8.	Claim 8: <i>The data redundancy system according to claim 1, wherein the second redundancy appliance accesses the data stored in storage by the first redundancy appliance when the second redundancy appliance acts as the first redundancy appliance in response to a failover event.</i>	65
B.	Ground 2: Claims 9-18 are Obvious Over <u>Lubbers</u> , <u>Sicola</u> and <u>Ofek</u>	66
1.	Claim 9: <i>The data redundancy system according to claim 1, wherein the first redundancy appliance forwards redundant data to the secondary storage facility as send batches.</i>	75
2.	Claim 10: <i>The data redundancy system according to claim 9, wherein a send batch size is determined according to either a time interval for collecting transactions into the send batch or a predetermined quantity of transactions to be included in the send batch.</i>	77

3. Claim 11: *The data redundancy system according to claim 9, wherein a send batch size is determined adaptively based on available communication bandwidth between the primary storage facility and the secondary storage facility. 78*
4. Claim 12: *The data redundancy system according to claim 1, wherein a first write transaction in a send batch at the primary storage facility is replaced with a second write transaction where the second write transaction overwrites data written by the first write transaction. 80*
5. Claim 13: *The data redundancy system according to claim 1, wherein the third redundancy appliance stores redundant data to the storage associated with the secondary facility according to receive batches. 82*
6. Claim 14: *The data redundancy system according to claim 13, wherein the third redundancy appliance stores a receive batch after the receive batch is determined safe to apply. 84*
7. Claim 15: *The data redundancy system according to claim 14, wherein a receive batch is determined safe to apply when a data integrity test is successfully performed. 86*
8. Claim 16: *The data redundancy system according to claim 13, wherein a first write transaction in a first receive batch at the secondary storage facility is replaced with a second write transaction in a second receive batch wherein the second write transaction overwrites data written by the first write transaction and the first and second receive batches are combined. 87*
9. Claim 17 89
 - a. [17preamble]: “A data redundancy method for storing a primary copy of data at a primary storage facility and for storing data that is redundant of the primary copy at a secondary storage facility, the method comprising” 89
 - b. [17a]: “receiving a sequence of write requests at a first redundancy appliance at the primary storage facility;” 90

c.	[17b]: “storing data for the sequence of write requests in primary storage at the primary storage facility;”	91
d.	[17c]: “logging at least a sequence of write requests for transfer to the secondary storage facility;”	91
e.	[17d]: “forwarding the logged write requests as send batches to a second redundancy appliance at the secondary storage facility disjoint from the primary storage facility, the second storage facility batching the received write request as receive batches;”	92
f.	[17e]: “storing data associated with the receive batches in secondary storage at the secondary storage facility”	93
g.	[17f]: “replacing a first write transaction in a receive batch at the second storage facility with a second write transaction wherein the second write transaction overwrites data written by the first write transaction; and”	94
a.	[17g]: “the second redundancy appliance acting as the first redundancy appliance in response to a failover event.”	94
10.	Claim 18	95
a.	[18preamble]: “A data redundancy system, comprising:”	95
b.	[18a]: “a primary storage facility including at least one primary redundancy appliance for receiving a sequence of write requests and for storing data for the sequence of write requests in storage associated with the primary storage facility, the primary storage facility batching the write requests as send batches for transfer to a secondary storage facility; and”	95
c.	[18b]: “the secondary storage facility disjoint from the primary storage facility including at least one secondary redundancy appliance for receiving the sequence of write requests and for storing data for	

	the sequence of write requests in storage associated with the secondary storage facility, the second storage facility batching the received write request as receive batches, the second redundancy appliance acting as the primary redundancy appliance in response to a failover event;”	95
d.	[18c]: “wherein a first write transaction in a receive batch at the second storage facility is replaced with a second write transaction where the second write transaction over writes data written by the first write transaction.”	96
VIII.	Discretionary Denial is Not Appropriate.....	96
A.	Denial under § 325(d) is not warranted.....	96
B.	The Board Should Not Deny Institution Under 35 U.S.C. § 314(a)....	97
IX.	Conclusion	100

TABLE OF ABBREVIATIONS AND CONVENTIONS

Petitioner	SAP America, Inc.
Patent Owners	Valtrus Innovations Ltd. (“Valtrus”) and Key Patent innovations Ltd. (“KPI”)
’182 Patent	U.S. Patent No. 7,152,182 (the patent being challenged herein)
Challenged Claims	Claims 1-18
POSITA	Person of Ordinary Skill in the Art

CLAIM LISTING

Claim 1	
1[preamble]	A data redundancy system, comprising:
1[a]	a primary storage facility for storing a primary copy of data, the primary storage facility including
1[b]	a first redundancy appliance operable in a role for receiving a sequence of write requests and for storing data for the sequence of write requests in storage associated with the primary storage facility; and
1[c]	a second redundancy appliance for shadowing the first redundancy appliance wherein the second redundancy appliance assumes the role of the first redundancy appliance;
1[d]	a secondary storage facility for storing data that is redundant of the primary copy of the data, the secondary storage facility discrete from the primary storage facility and including:
1[e]	a third redundancy appliance operable in a role for storing redundant data for the sequence of write requests in storage associated with the secondary facility; and
1[f]	a fourth redundancy appliance for shadowing the third redundancy appliance wherein the fourth redundancy appliance assumes the role of the third redundancy appliance in the event of a fault at the third redundancy appliance.
Claim 2	
2	The data redundancy system according to claim 1, wherein the secondary storage facility assumes the role of the primary storage facility in response to a failover event, the failover event including a communication traffic condition having a greater portion of request traffic originating closer to the secondary storage facility than to the primary storage facility.

Claim 3	
3	The data redundancy system according to claim 1, wherein the second redundancy appliance assumes the role of the first redundancy appliance in response to a failover event selected from the group consisting of a manually initiated event and an automatically initiated event.
Claim 4	
4	The data redundancy system according to claim 1, wherein the first redundancy appliance updates the second redundancy appliance with information about the data in storage by forwarding a write record to the second redundancy appliance for each write request, the second redundancy appliance sending acknowledgements to the first redundancy appliance in batches in response to receiving write records.
Claim 5	
5	The data redundancy system according to claim 4, wherein the first redundancy appliance sends an acknowledgement to an originator of the write request in response to receiving the acknowledgement from the second redundancy appliance.
Claim 6	
6	The data redundancy system according to claim 1, wherein the first redundancy appliance is disjoint from the second redundancy appliance.
Claim 7	
7	The data redundancy system according to claim 1, wherein the second redundancy appliance maintains a shadow copy of data stored by the first redundancy appliance wherein the second redundancy appliance accesses the shadow copy of the data when the second redundancy appliance assumes the role of the first redundancy appliance.

Claim 8	
8	The data redundancy system according to claim 1, wherein the second redundancy appliance accesses the data stored in storage by the first redundancy appliance when the second redundancy appliance acts as the first redundancy appliance in response to a failover event.
Claim 9	
9	The data redundancy system according to claim 1, wherein the first redundancy appliance forwards redundant data to the secondary storage facility as send batches.
Claim 10	
10	The data redundancy system according to claim 9, wherein a send batch size is determined according to either a time interval for collecting transactions into the send batch or a predetermined quantity of transactions to be included in the send batch.
Claim 11	
11	The data redundancy system according to claim 9, wherein a send batch size is determined adaptively based on available communication bandwidth between the primary storage facility and the secondary storage facility.
Claim 12	
12	The data redundancy system according to claim 1, wherein a first write transaction in a send batch at the primary storage facility is replaced with a second write transaction where the second write transaction overwrites data written by the first write transaction.
Claim 13	
13	The data redundancy system according to claim 1, wherein the third redundancy appliance stores redundant data to the

	storage associated with the secondary facility according to receive batches.
Claim 14	
14	The data redundancy system according to claim 13, wherein the third redundancy appliance stores a receive batch after the receive batch is determined safe to apply.
Claim 15	
15	The data redundancy system according to claim 14, wherein a receive batch is determined safe to apply when a data integrity test is successfully performed.
Claim 16	
16	The data redundancy system according to claim 13, wherein a first write transaction in a first receive batch at the secondary storage facility is replaced with a second write transaction in a second receive batch wherein the second write transaction overwrites data written by the first write transaction and the first and second receive batches are combined.
Claim 17	
17[preamble]	A data redundancy method for storing a primary copy of data at a primary storage facility and for storing data that is redundant of the primary copy at a secondary storage facility, the method comprising:
17[a]	receiving a sequence of write requests at a first redundancy appliance at the primary storage facility;
17[b]	storing data for the sequence of write requests in primary storage at the primary storage facility;
17[c]	logging at least a sequence of write requests for transfer to the secondary storage facility;
17[d]	forwarding the logged write requests as send batches to a second redundancy appliance at the secondary storage facility

	disjoint from the primary storage facility, the second storage facility batching the received write request as receive batches;
17[e]	storing data associated with the receive batches in secondary storage at the secondary storage facility;
17[f]	replacing a first write transaction in a receive batch at the second storage facility with a second write transaction wherein the second write transaction overwrites data written by the first write transaction; and
17[g]	the second redundancy appliance acting as the first redundancy appliance in response to a failover event.
Claim 18	
18[preamble]	A data redundancy system, comprising:
18[a]	a primary storage facility including at least one primary redundancy appliance for receiving a sequence of write requests and for storing data for the sequence of write requests in storage associated with the primary storage facility, the primary storage facility batching the write requests as send batches for transfer to a secondary storage facility; and
18[b]	the secondary storage facility disjoint from the primary storage facility including at least one secondary redundancy appliance for receiving the sequence of write requests and for storing data for the sequence of write requests in storage associated with the secondary storage facility, the second storage facility batching the received write request as receive batches, the second redundancy appliance acting as the primary redundancy appliance in response to a failover event;
18[c]	wherein a first write transaction in a receive batch at the second storage facility is replaced with a second write transaction where the second write transaction over writes data written by the first write transaction.
18[d]	wherein a first write transaction in a receive batch at the second storage facility is replaced with a second write

	transaction where the second write transaction over writes data written by the first write transaction.
--	---

PETITIONER’S EXHIBIT LIST

No.	Description
1001	U.S. Patent No. 7,152,182 (“’182 Patent”)
1002	File History of U.S. Patent No. 7,152,182
1003	Declaration of Kevin C. Almeroth, Ph.D.
1004	<i>Curriculum Vitae</i> of Kevin C. Almeroth, Ph.D.
1005	U.S. Pub. No. 2003/0187847 (“ <u>Lubbers</u> ”)
1006	U.S. Patent No. 6,601,187 (“ <u>Sicola</u> ”)
1007	U.S. Pat. No. 5,901,327 (“ <u>Ofek</u> ”)
1008	Ramez Elmasri and Shamkant B. Navathe, FUNDAMENTALS OF DATABASE SYSTEMS (2d ed.1994)
1009	Original Complaint for Patent Infringement in <i>Valtrus Innovations Ltd. v. SAP America, Inc. et al</i> , 2:24-cv-00021-JRG (E.D. Tex.)
1010	Original Complaint for Patent Infringement in <i>Valtrus Innovations Ltd. v. SAP America, Inc. et al</i> , 2:24-cv-00533-JRG (E.D. Tex.)
1011	Director Vidal’s Memorandum Regarding Discretionary Denial Procedure, Dated June 21, 2022
1012	First Amended Docket Control Order in <i>Valtrus Innovations Ltd. v. SAP America, Inc. et al</i> , 2:24-cv-00021-JRG (E.D. Tex.)
1013	Motion to Dismiss for Lack of Standing (Public Version) in <i>Valtrus Innovations Ltd. v. SAP America, Inc. et al</i> , 2:24-cv-00021-JRG (E.D. Tex.)

No.	Description
1014	Motion to Reconsider Ruling on Motion to Dismiss for Lack of Standing (Public Version) in <i>Valtrus Innovations Ltd. v. SAP America, Inc. et al</i> , 2:24-cv-00021-JRG (E.D. Tex.)
1015	Plaintiffs' Notice Regarding <i>Valtrus II</i> in <i>Valtrus Innovations Ltd. v. SAP America, Inc. et al</i> , 2:24-cv-00021-JRG (E.D. Tex.)
1016	SAP's Invalidity Contentions (Cover Pleading) in <i>Valtrus Innovations Ltd. v. SAP America, Inc. et al</i> , 2:24-cv-00021-JRG (E.D. Tex.)

I. INTRODUCTION

Petitioner requests *inter partes* review (“IPR”) of claims 1-18 (“Challenged Claims”) of U.S. Patent No. 7,152,182 (“’182 Patent”). The claims of the ’182 Patent are directed to a data redundancy system (claims 1 and 18) or method (claim 17). Data redundancy was well known in the industry years before the ’182 Patent’s earliest claimed priority date.

II. COMPLIANCE WITH IPR REQUIREMENTS

A. Grounds for Standing

As required by 37 C.F.R. § 42.104(a), Petitioner certifies that the ’182 Patent “is available for inter partes review and that the Petitioner is not barred or estopped from requesting an inter partes review challenging the patent claims on the grounds identified in the petition.” Neither Petitioner nor any real party in interest has filed a civil action challenging the validity of a claim of the ’182 Patent. This petition is being filed no more than 1 year after the date on which Petitioner or real parties in interest or privies of Petitioner were served with a complaint alleging infringement of the ’182 Patent.

B. Mandatory Notices

Real Party-in-Interest: Petitioner identifies SAP America, Inc. and SAP SE (collectively, “SAP”) as real parties-in-interest.

Related Matters: No previous IPR or post-grant proceeding has been filed challenging the ’182 Patent. Valtrus Innovations Limited (“Valtrus”) has asserted the

'182 Patent in litigation against SAP in *Valtrus Innovations Ltd. v. SAP America, Inc. et al*, 2:24-cv-00021-JRG (E.D. Tex.), Valtrus and Key Patent Innovations LTD. ("KPI") (collectively, "Patent Owners") have asserted the '182 Patent in litigation against SAP in *Valtrus Innovations Ltd. v. SAP America, Inc. et al*, 2:24-cv-00533 (E.D. Tex.), and SAP filed a declaratory judgment action of non-infringement of the '182 Patent against Valtrus, KPI, and Patent Platform Services, LLC in *SAP America, Inc. v. Valtrus Innovations Limited et al*, 1:24-cv-00756-GBW (D. Del.).

Lead and Back-up Counsel:

Lead Counsel	Back-up Counsel
Louis L. Campbell (Reg. # 59,963) WINSTON & STRAWN LLP 255 Shoreline Drive, Suite 520 Redwood City, CA 94065 Tel: (650) 858-6500 LLcampbell@winston.com	Scott M. Border (Reg. # 77,744) WINSTON & STRAWN LLP 1901 L Street NW Washington, DC 20036 Tel: (202) 282-5000 SBorder@winston.com
	Samuel W. Riebe (Reg. # 78,044) WINSTON & STRAWN LLP 2121 N. Pearl Street, Suite 900 Dallas, TX 75201 Tel: (214) 453-6500 SRiebe@winston.com
	Michael Bittner (<i>Pro hac vice</i> to be submitted) WINSTON & STRAWN LLP 2121 N. Pearl Street, Suite 900 Dallas, TX 75201 Tel: (214) 453-6500 MBittner@winston.com

Lead Counsel	Back-up Counsel
	Matt R. McCullough (<i>Pro hac vice</i> to be submitted) WINSTON & STRAWN LLP <i>Pro hac vice</i> to be submitted Winston & Strawn LLP 255 Shoreline Drive, Suite 520 Redwood City, CA 94065 Tel: (650) 858-6500 MRMcCullough@winston.com
	William Logan (<i>Pro hac vice</i> to be submitted) <i>Pro hac vice</i> to be submitted WINSTON & STRAWN LLP 800 Capitol Street, Suite 2400 Houston, TX 77002 Tel: (713) 651-2600 WLogan@winston.com
	Patrick Clark (<i>Pro hac vice</i> to be submitted) WINSTON & STRAWN LLP <i>Pro hac vice</i> to be submitted 2121 N. Pearl Street, Suite 900 Dallas, TX 75201 Tel: (214) 453-6500 PClark@winston.com

Service Information: Service information is provided in the designation of counsel above. Petitioner consents to service of all documents via electronic mail at IPR-SAP-Val@winston.com.

C. Fee Authorization

The Office is authorized to charge all fees due in connection with this proceeding to Deposit Account No. 50-1814.

III. THE CHALLENGED CLAIMS

Under 37 C.F.R. §§ 42.104(b) and 42.22, Petitioner requests that the Board institute this IPR on the Challenged Claims of the '182 Patent and cancel those claims as unpatentable on the following grounds:

Ground	Claims	Basis for Unpatentability
1	1-8	Obviousness under 35 U.S.C. § 103 by U.S. Patent Publication No. 2003/0187847 (“ <u>Lubbers</u> ”) and U.S. Patent No. 6,601,187 (“ <u>Sicola</u> ”)
2	9-18	Obviousness under 35 U.S.C. § 103 by <u>Lubbers</u> , <u>Sicola</u> , and U.S. Pat. No. 5,901,327 (“ <u>Ofek</u> ”)

IV. THE '182 PATENT

A. Earliest Effective Filing Date

The '182 Patent issued from Application No. 10/456,053 (“'053 Application”), filed June 6, 2003, which is a standalone non-provisional application with no priority claim. Consequently, the earliest possible effective filing date for the '182 Patent is June 6, 2003.

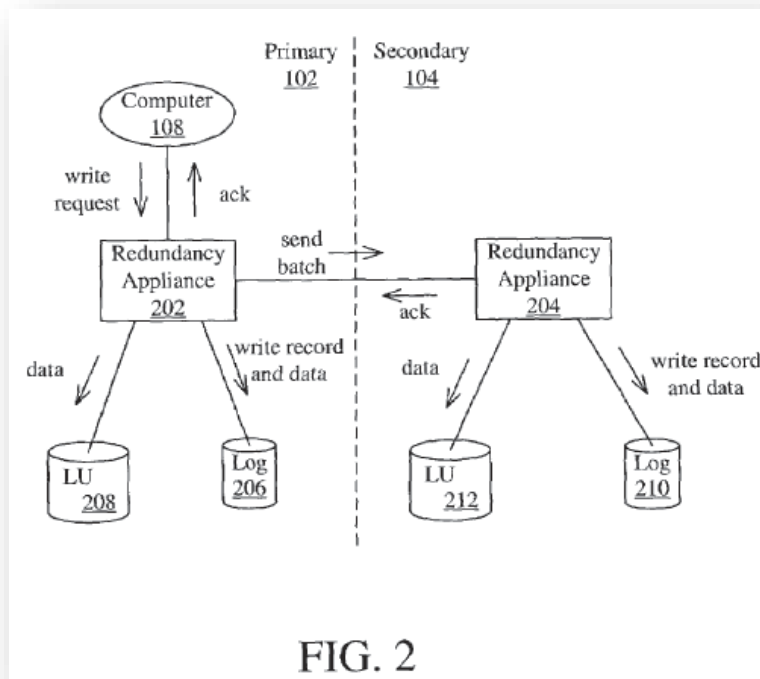
B. Level of Ordinary Skill in the Art

As of June 6, 2003, a POSITA in the '182 Patent's field would have a working knowledge of data storage, including data redundancy systems and methods used in a fault-tolerant manner. Such a POSITA would have had a bachelor's degree in electrical engineering, computer science, or equivalent training, and approximately

two years of work experience in software development in the field of network data processing and storage. Additional education could compensate for work experience, and vice versa. Ex-1003, ¶ 33. Based on his education and experience, Dr. Almeroth has at least the knowledge of and qualifies as a POSITA. Ex-1003, ¶¶ 7-31.

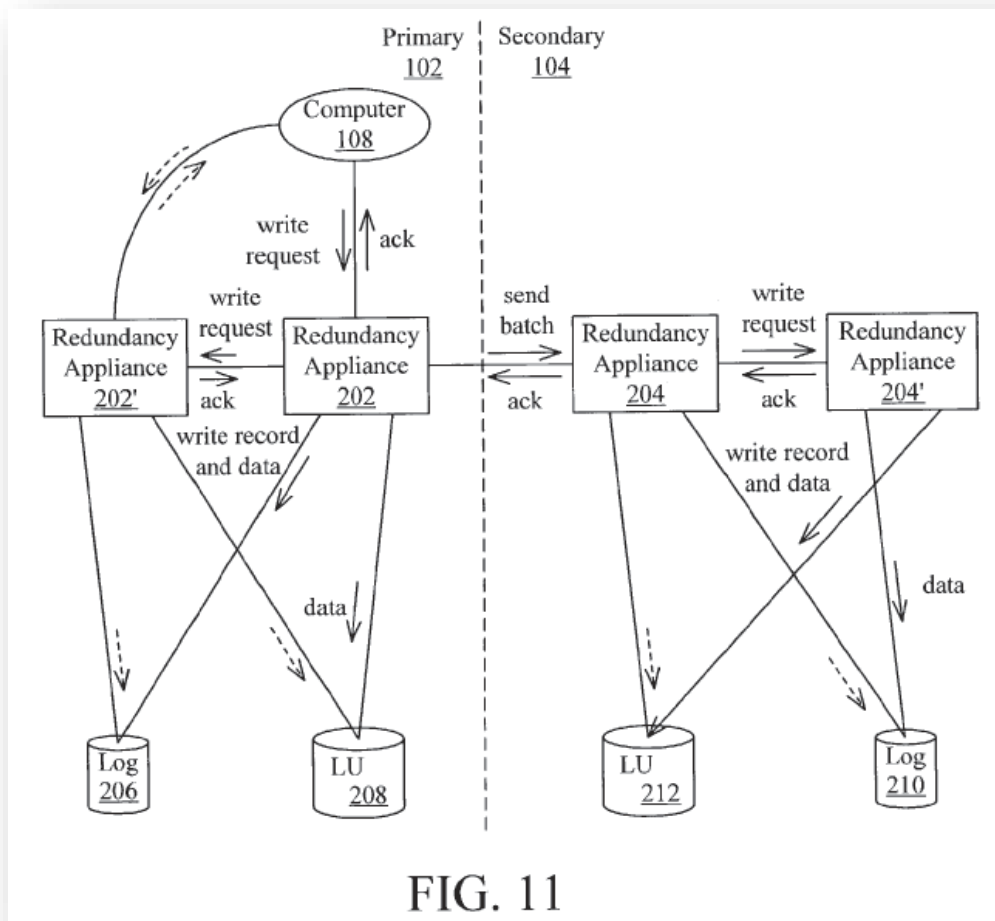
C. Overview of the '182 Patent

The '182 Patent describes techniques for fault-tolerant data redundancy operations. Ex-1001, Abstract, 1:29-31, 2:64-66. The computer system includes a primary storage facility and a secondary storage facility at different locations. Ex-1001, 1:55-59, 2:66-3:1, 3:18-31. The secondary storage facility stores data that is redundant of the primary storage facility. Ex-1001, 1:55-59, 2:66-3:1, 4:1-5. Figure 2 shows operation of the primary and secondary storage facilities.



Ex-1001, FIG. 2, 29:29-31. The computer system has two operating modes: normal operation, and operation in the event of a fault. Ex-1001, 4:17-28, 4:50-68, 12:66-13:9, 13:32-33.

The primary and secondary storage facilities include data redundancy appliances, which can be implemented by hardware or software. Ex-1001, 1:57-67, 3:1-9, 4:29-41. A first data redundancy appliance of the primary storage facility receives a sequence of write requests and stores data for the sequence of write requests in storage associated with the primary storage facility. Ex-1001, 1:57-67, 3:1-9. A data redundancy appliance of the secondary storage facility shadows the first redundancy appliance and assumes its role in the event of a fault at the first redundancy appliance. Ex-1001, 1:57-67, 3:1-9. In certain arrangements, multiple redundancy appliances can be included in each storage facility. In contrast to Figure 2 above, Figure 11 shows multiple redundancy appliances at each storage facility.



As demonstrated below, all the features recited in the challenged claims of the '182 Patent were known in the prior art before the alleged invention date of the '182 Patent. *Infra* § VII; Ex-1003, ¶¶ 50-264.

V. CLAIM CONSTRUCTION

Claims are given their "ordinary and customary meaning" as understood by a POSITA and the prosecution history pertaining to the patent. 37 C.F.R. § 42.100(b). Because a POSITA would find the challenged claims unpatentable under any interpretation consistent with their plain and ordinary meaning in the context of the

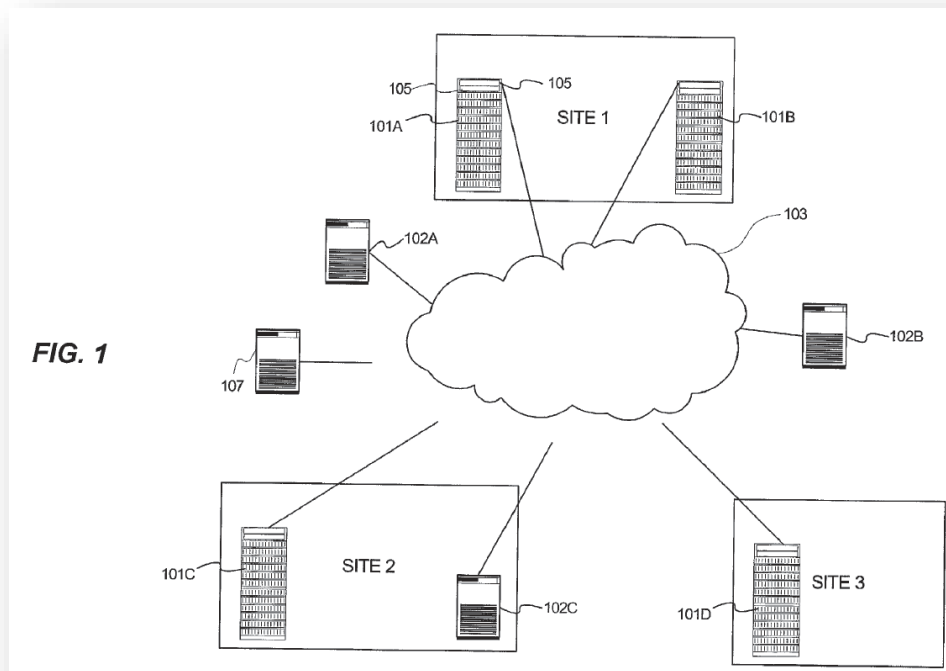
'139 Patent, *see* Ex-1003, ¶¶ 48-49., the Board need not expressly construe the claim terms. *See Vivid Techs., Inc. v. Am. Sci. & Eng'g. Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999).

VI. PRINCIPAL PRIOR ART

A. Lubbers (Ex-1005)

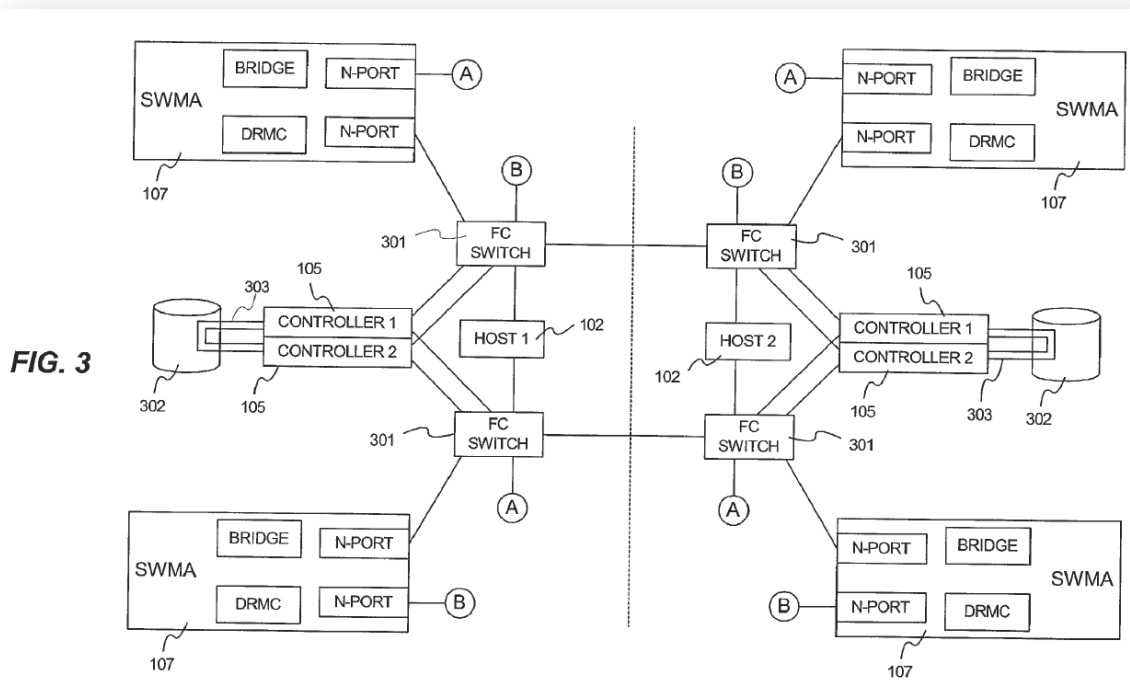
The application underlying U.S. Pub. No. 2003/0187847 ("Lubbers") was filed on March 26, 2002, and published on October 2, 2003, and thus qualifies as prior art at least under pre-AIA 35 U.S.C. § 102(e).

Lubbers discloses a multi-site storage area network (SAN) with multiple sites and storage cell(s) 101 accessible by host computer(s) 102 (shown in Figure 1):



Ex-1005, [0028]-[0029]; FIG. 1. Storage cells 101 and hosts 102 couple to a data communication network 103. Ex-1005, [0030]. Storage cells 101 can simultaneously function as primary storage for any host 102 and secondary storage for a replica of data from another storage cell 101. Ex-1005, [0036].

Lubbers discloses a disaster tolerant configuration with redundant components in Figure 3:



Ex-1005, [0037]; FIG. 3.

The physical storage devices 302 are coupled to controllers 105 by a high-speed redundant connection such as fibre channel arbitrated loop (FCAL) connections 303. Ex-1005, [0040]. Each of the tens or hundreds of individual hard

disk drives can be accessed by either controller 105 in a storage cell. Ex-1005, [0040].

Lubbers discloses a registration and discovery process that allows each controller 105 to connect to other controllers, as well as a “failover” condition in which one controller 105 fails. Ex-1005, [0045]-[0046]. Redundant components can operate simultaneously and in parallel to improve throughput as well as serving during failover to replace functionality of failed systems, components, and connections. Ex-1005, [0064]. Lubbers explains redundant components provide great flexibility in implementing a particular system to emphasize any balance of low latency, high availability, disaster tolerance, and cost. Ex-1005, [0064].

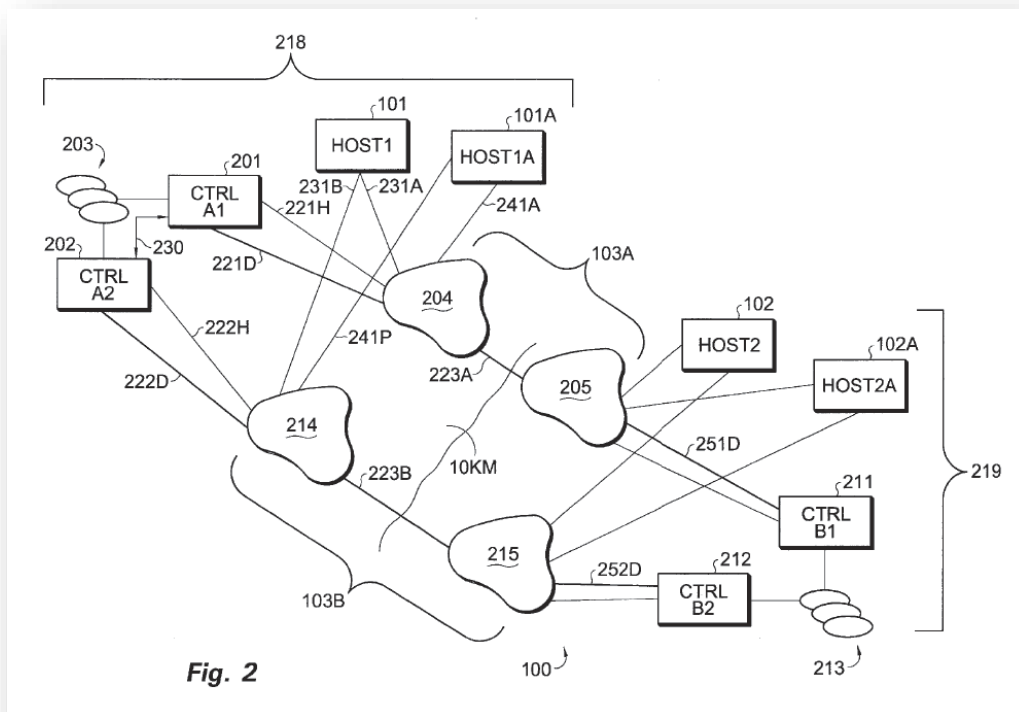
Operation can be synchronous or asynchronous, so Lubbers discloses logging and merging to track and preserve the order of write operations. Ex-1005, [0059]-[0060]. The sites may maintain logs of recent transactions (GSN, VBA, LUN, Block Count) by copying input/output (I/O) operations from the host computer into the log unit’s cache buffers. Ex-1005, [0076]. Lubbers discloses that a processor of the controller associated with the data log may monitor and maintain a ratio of logging writes to merge writes. Ex-1005, [0080]. Those merges write I/Os are transmitted to a destination storage system where they are merged into the target LUN on the destination storage system. Ex-1005, [0080].

B. Sicola (Ex-1006)

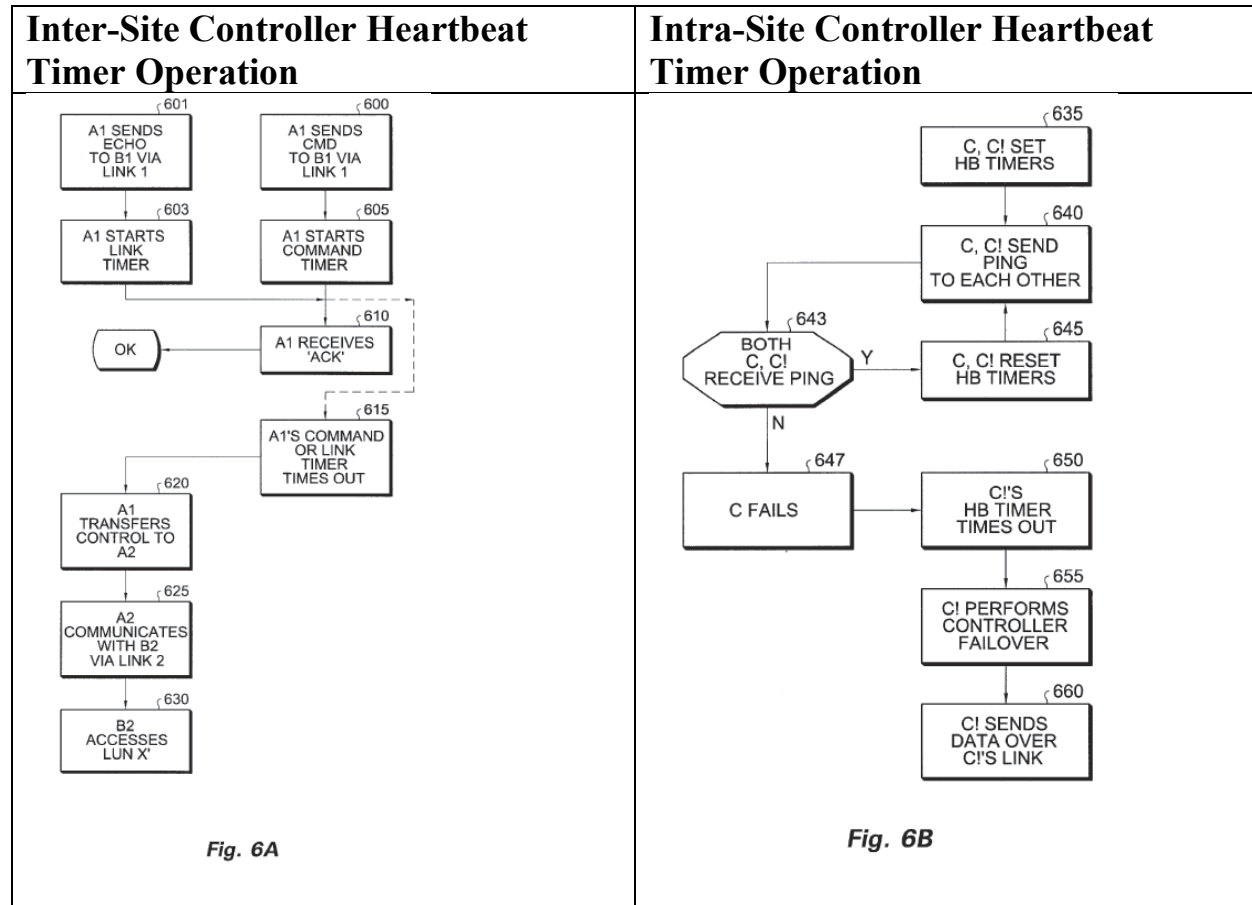
The application underlying U.S. Patent No. 6,601,187 (“Sicola”) was filed on March 31, 2000, and published upon issuance on July 29, 2003, and is therefore prior art under at least 35 U.S.C. § 102(e).

Lubbers incorporates Sicola by reference. Ex-1005, [0008] (stating “Compaq Corporation introduced an array controller referred to as the HSG80, that implemented Data Replication Management features, as described in U.S. patent application Ser. No. 09/539,745 assigned to the assignee of the present application and incorporated herein by reference, that implemented an architecture utilizing redundant storage controllers”).

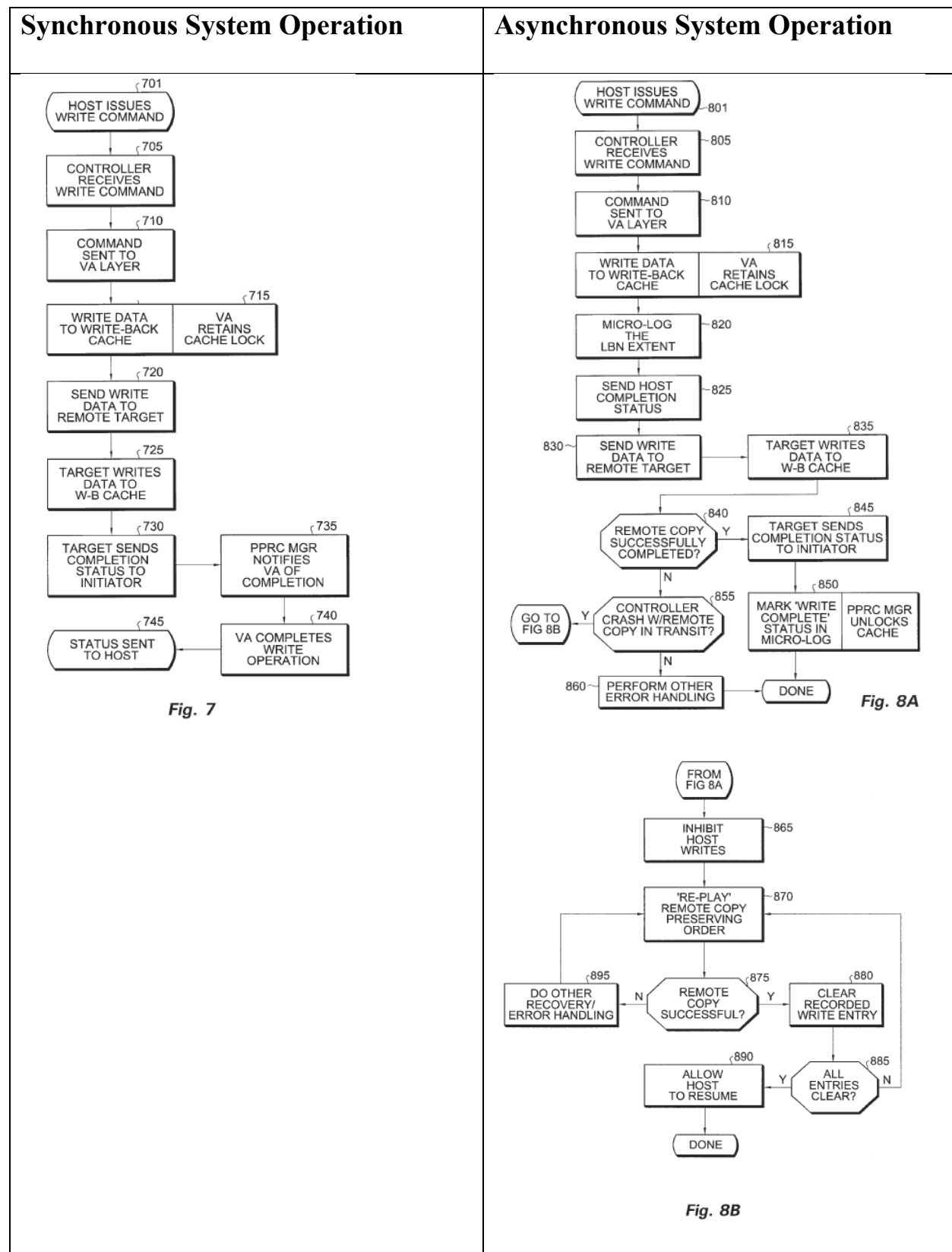
Sicola recognizes the importance of storage controllers and desirability in minimizing interrupted service due to controller malfunctions. Ex-1006, 1:37-43. Sicola discloses a system 100 that “employs a switched, dual-fabric, Fibre Channel configuration to provide a disaster tolerant storage system.” Ex-1006, 6:1-3. The Sicola architecture is similar to Lubbers (shown in FIG. 2):



Ex-1006, 6:64-66, 7:1-5, 8:23-25; FIG. 2. The operation of two array controllers, including heartbeat timers, are shown in FIG. 6A (inter-site controllers) and 6B (intra-site controllers):



Ex-1006, 9:35-37; FIGS 6A, 6B. Sicola describes synchronous system operation (FIG. 7) and asynchronous system operation (FIGS. 8A, 8B) involving a micro-logging and micro-merge process in which write commands can be exchanged and mirrored between controller-specific write back caches for both controllers in a redundant controller pair (as shown below):

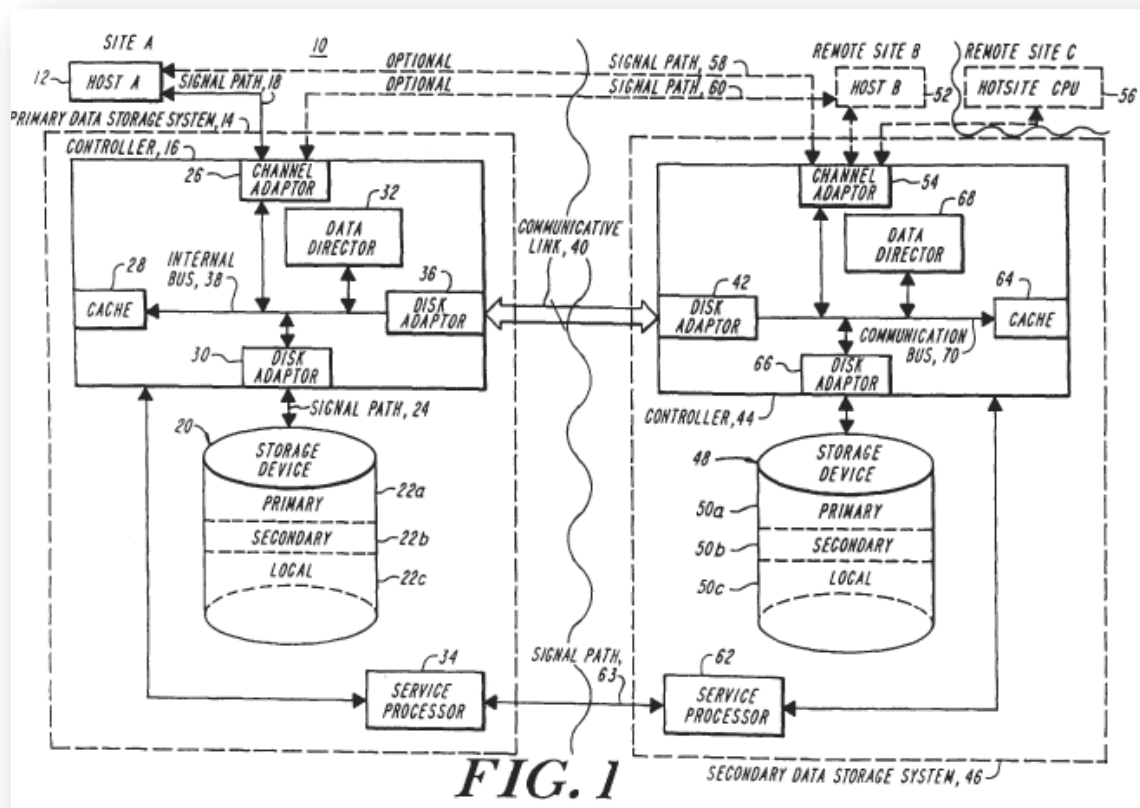


Ex-1006, 12:25-32, 12:39-43, 44-59, 13:6-13; FIGS. 7-8B.

C. Ofek (Ex-1007)

U.S. Pat. No. 5,901,327 ("Ofek") issued on May 4, 1999 and is prior art under at least 35 U.S.C. § 102(b).

Ofek discloses a data storage system 10 with a similar overall architecture to Lubbers and Sicola (shown in FIG. 1):



Ex-1007, 4:35-38; FIG. 1. Ofek discloses bundling multiple host commands (e.g., write requests) is desirable to more effectively using the capacity of the data link to transmit data between sites. Ex-1007, 34:66-35:6, 4:35-38.

A “link adapter 236 services the link queue 504 by transferring the write data across the link 240” (shown in FIG. 18):

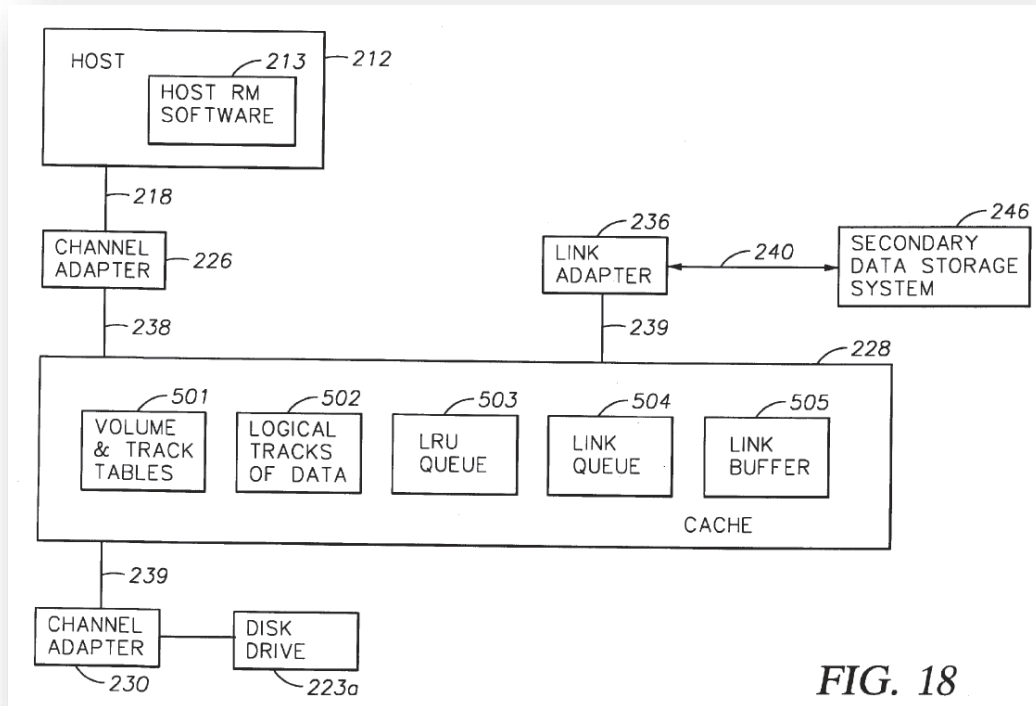


FIG. 18

Ex-1007, 35:52-54; FIG. 18. Ofek discloses using a one-byte password to confirm the validity of the entry, among other data integrity and verification mechanisms. Ex-1007, 35:60-36:2.

VII. ALL CLAIMS ARE UNPATENTABLE

The Challenged Claims are unpatentable based on two grounds. Ground 1 establishes that claims 1-8 would have been obvious over Lubbers and Sicola. Ground 2 establishes that claims 9-18 would have been obvious over Lubbers, Sicola and Ofek.

As explained below, the scope and content of Lubbers alone or in combination with Sicola and Ofek includes every element of claims 1-18 without any meaningful difference between the teachings of those references and the elements of the asserted claims. After considering the teachings in the prior art from the perspective of a POSITA, it would have been obvious to combine the teachings of those known elements in the way they are recited in claims 1-18 as of June 6, 2003. Ex-1003, ¶ 52. Patent Owners have identified no secondary considerations purporting to show claims 1-18 would not have been obvious as of June 6, 2003.

Lubbers, Sicola, and Ofek are analogous art to the '182 Patent for at least two reasons. First, Lubbers, Sicola, and Ofek are directed toward the same field of endeavor of the '182 Patent—techniques for redundant data storage. Ex-1003, ¶¶ 90, 180; Ex-1001, 1:29-31; Ex-1005, [0001], [0007]; Ex-1006, 1:8-14; Ex-1007, 1:8-11.

Second, POSITAs knew that communication delays and component failures were a common data storage problem as of June 6, 2003. For example, Lubbers, Sicola, and Ofek each are reasonably pertinent to that problem and provide techniques for redundant data storage to overcome that problem. ¶¶ 91, 181; Ex-1005, [0038]; Ex-1006, 3:63-4:6; Ex-1007, 3:3-22. The '182 Patent confirms that communication delays and component failures were a problem in data storage as of June 6, 2003. Ex-1001, 1:45-51.

A. Ground 1: Claims 1-8 Would Have Been Obvious Over Lubbers and Sicola

Lubbers taught or suggested every element of claims 1-8. Even had Lubbers alone not expressly taught every element of claims 1-8, Sicola (incorporated by reference in Lubbers), provides clarifying details confirming Lubbers would have rendered obvious every element of claims 1-8. Ex-1003, ¶ 81. Since Lubbers incorporates Sicola by reference, the Board can properly consider it as part of the disclosure of Lubbers. *Arbutus Biopharma Corp. v. ModernaTX, Inc.*, 65 F.4th 656, 662-63 (Fed. Cir. 2023).

Regardless, it would have been obvious for a POSITA to combine the teachings of Lubbers and Sicola. Ex-1003, ¶ 82.

A POSITA reading Lubbers would have been motivated to look to Sicola for additional details for operation of the controllers described in Lubbers because Lubbers specifically references the controllers of Sicola, and Sicola provides additional details of an architecture that would be of interest to a POSITA. Additionally, Lubbers and Sicola were originally assigned to the same company and have two common inventors: Stephen Sicola and Susan Elkington. The subject matter of Lubbers also significantly overlaps with Sicola, as demonstrated by the similar fields of the invention, similar architectures described and shown in the figures, and Lubbers expressly referring to Sicola for its array controllers. Thus, it would have been obvious for a POSITA looking at the teachings of Lubbers to study

the teachings of Sicola beyond the clarifying details on the controllers. As explained below, the combined teachings of Lubbers and Sicola includes the architecture and functionality of Lubbers with additional implementation details of Sicola. Ex-1003, ¶ 83.

It would have been obvious to use clarifying details from Sicola for the operation and communication between controllers of Lubbers. First, Lubbers teaches how redundant controllers can update one another (*“first redundancy appliance updates the second redundancy appliance with information about data in storage”*). For example, Lubbers explains that if a “controller does not have the most current data, then the controller goes offline to the host to receive a full copy from the controller that assumes the role of the source controller.” Ex-1005, [0073]. Sicola further explains how redundant controllers update one another in a multi-site storage system. For example, Sicola explains that asynchronous system operation involves micro-logging and micro-merge processes in which write commands can be exchanged and written to controller-specific write back caches. Ex-1006, 12:16-30. Sicola also describes that outstanding writes logged to initiator controller A1’s write-back cache, are mirrored in partner controller A2’s (mirrored) write-back cache. Ex-1003, ¶ 84; Ex-1006, 13:2-13.

Second, Lubbers teaches that controllers can send acknowledgements to one another in batches after receiving write requests (*“second redundancy appliance*

sending acknowledgments to the first redundancy appliance in batches in response to receiving write records”). Ex-1005, [0068] (explaining each controller periodically sends “heartbeats,” or pings to its peer controllers, and the peer controllers respond to the ping with a message including information useful in managing data transfer processes between peer controllers). Sicola further explains controllers communicating and sending acknowledgments in this manner. Ex-1006, 10:22-26 (explaining intra-site controller heartbeat timer operation), 10:45-53 (explaining that during normal operation, companion controllers periodically send pings to each other via Dual Asynchronous Receiver/Transmitters, and the pings are received unless either controller has timed out); Ex-1003, ¶ 85.

Based on these teachings, a POSITA would have been motivated to enhance the teachings of Lubbers to further improve upon the stated goals of maintaining integrity and availability of data, minimizing latency and improving data transmission path performance, and minimizing controller load. A POSITA would have found it obvious to look to the teachings of Sicola for these improvements because Sicola also recognizes many of the same objectives. Ex-1003, ¶ 86; Ex-1006, 9:4-6, 11:35-40, 8:3-7.

A POSITA would also have been motivated to look to the teachings of Sicola to further improve the teachings of Lubbers, because Lubbers teaches that system operation can change based on certain criteria and tradeoffs depending on the desired

objectives. Ex-1003, ¶ 87; Ex-1005, [0042] (explaining multi-pathing software allows for failover between storage paths as well and static load balancing of storage volumes (logical units of storage capacity) between paths of controllers), Ex-1005, [0042] (describing how desired controller paths and storage are selected from criteria including path availability, path performance, path load, controller load, and the like).

A POSITA would have been motivated to combine teachings of Lubbers and Sicola because both references provide multiple levels of redundancy to solve the same data storage problem. Ex-1003, ¶ 88; Ex-1005, [0038] (explaining disaster tolerant configuration with hardware redundancy for multiple levels of fault tolerance to keep the primary and alternate sites running at almost non-stop metrics); Ex-1006, 3:63-4:6 (explaining the invention solves prior art problems and advances the field by providing a system that includes a completely redundant configuration with dual Fibre Channel fabric links interconnecting the components).

Further, the benefits of data redundancy and distributed databases were well-known as of June 6, 2003, when the application underlying the '182 Patent was filed. Ex-1008, 703-727. For example, data redundancy (1) “can improve availability remarkably as the entire system can continue to operate as long as at least one site is up;” and (2) “improves performance of retrieval for global queries, because the result

of such a query can be obtained locally from any one site.” Ex-1008, 711; Ex-1003, ¶ 89.

A POSITA would have reasonably expected to succeed in incorporating the teachings of Sicola with the teachings of Lubbers because Sicola merely provides additional details for similar systems and components implementing similar techniques already taught by Lubbers. Ex-1003, ¶ 92; Ex-1005, [0068], [0076]; Ex-1006, 12:16–30, 13:6–13

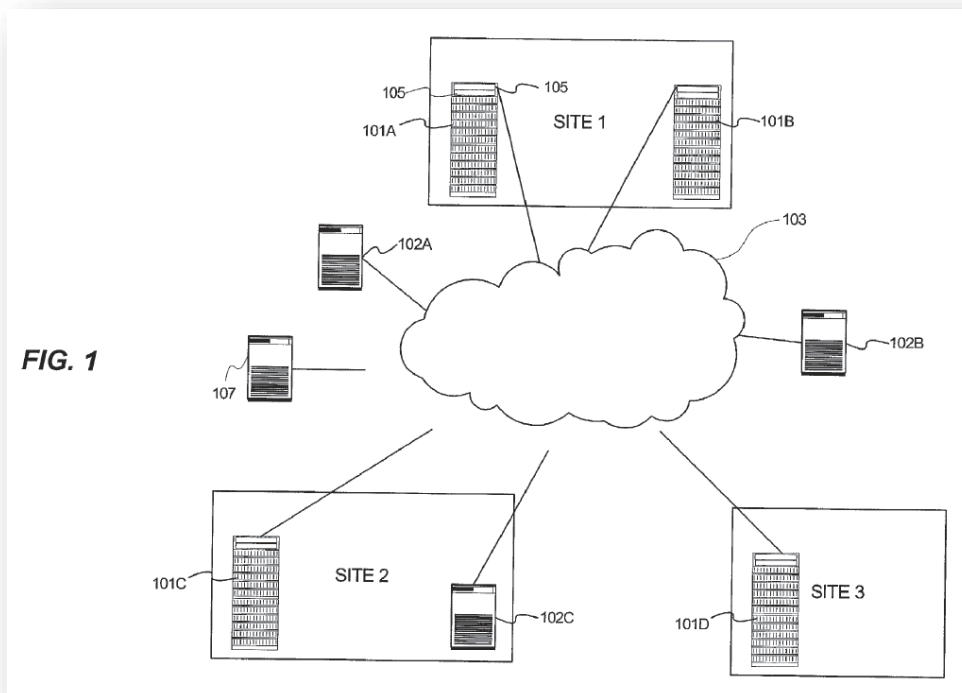
Finally, Lubbers and Sicola are at the same level of generality as the ’182 Patent such that a POSITA would have been able to apply that same level of skill in the art in understanding Lubbers and the ways in which the teachings could be implemented. Thus, a POSITA would have reasonably expected to successfully combine teachings of Lubbers and Sicola. Ex-1003, ¶ 93

As one example, A POSITA would have reasonably expected to successfully have redundant controllers for shadowing one another because Lubbers already contemplates the importance of hardware redundancy for multiple levels of fault tolerance to provide seamless operation and maintain performance when a component fails. Ex-1005, [0038]; Ex-1003, ¶ 94.

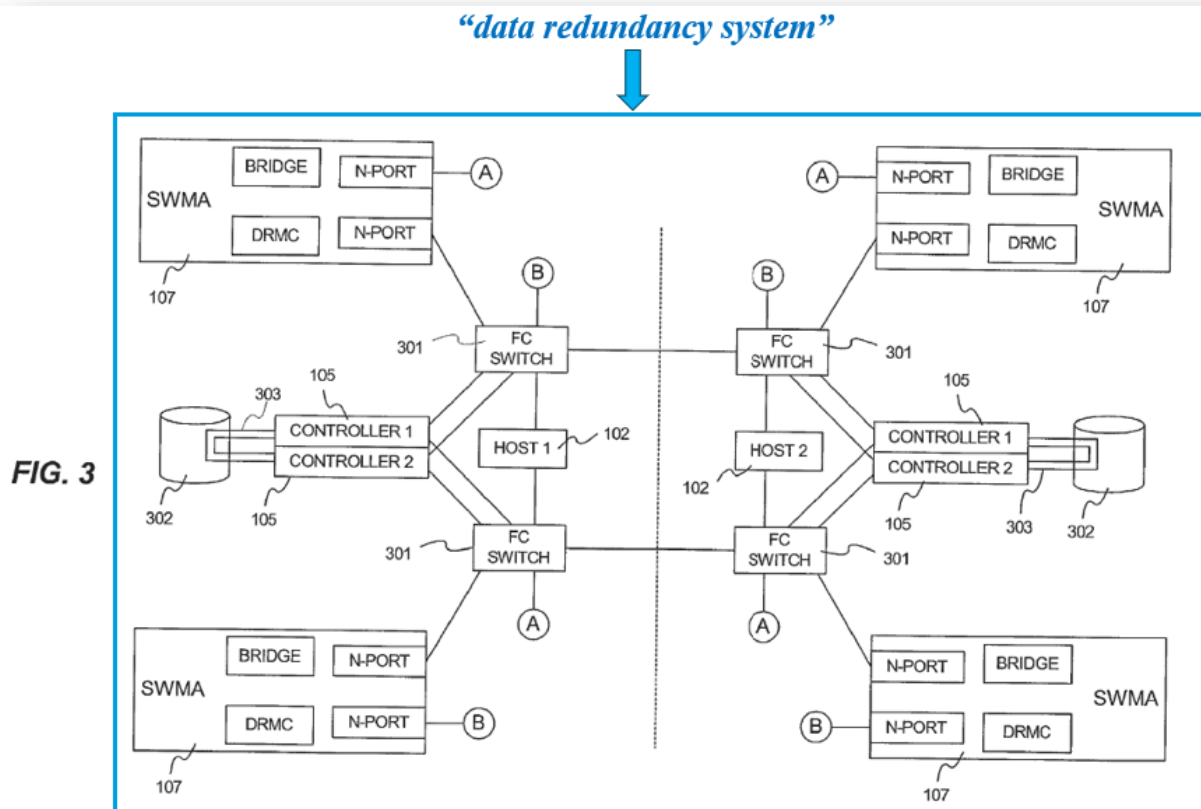
1. Claim 1

a. [1preamble]: “A data redundancy system, comprising:”

Lubbers teaches a multi-site storage area network (SAN) that is a “*data redundancy system*.” Ex-1005, [0001], [0010] (explaining the SAN is a computer-based information storage system that provides data redundancy functions, including “multiple replication, load balancing, and disaster tolerance....”). The SAN (“*data redundancy system*”) includes multiple sites, such as Site 1, Site 2, and Site 3 shown in the Figure 1 embodiment:



Ex-1005, [0001], [0010], [0028], FIG. 1; Ex-1003, ¶ 96. Figure 3 shows a disaster-tolerant embodiment of the SAN (“*data redundancy system*”) with redundant components:



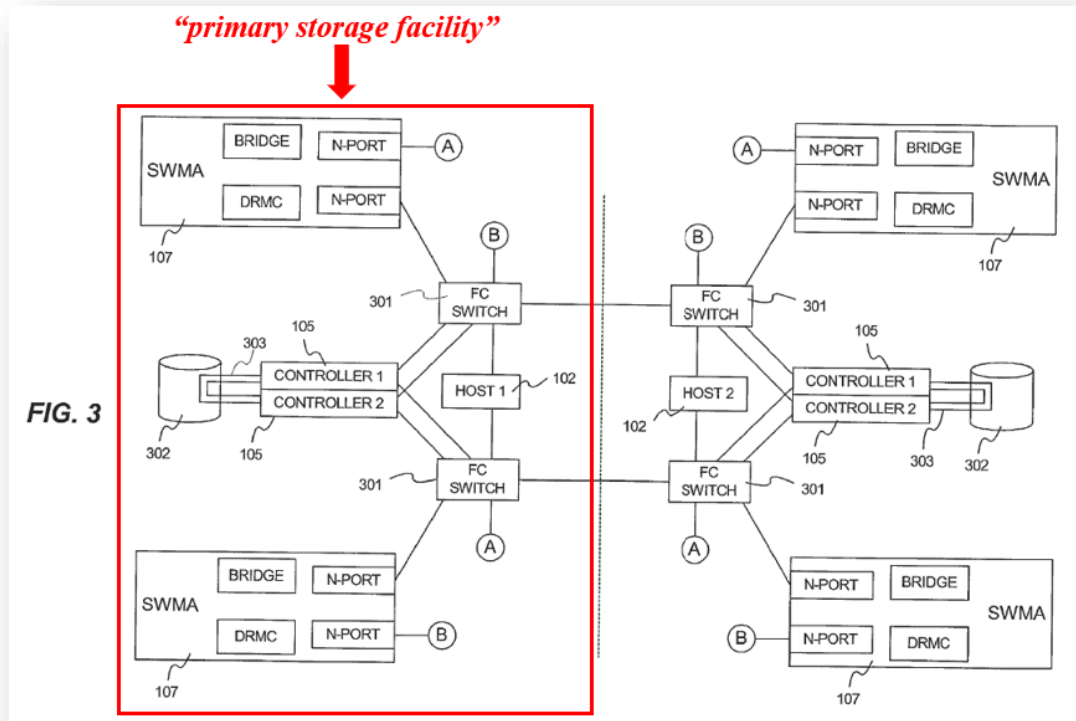
Ex-1005, [0038], FIG. 3 (annotated); Ex-1003, ¶ 97.

Accordingly, Lubbers taught or suggested this preamble to the extent it is limiting. Ex-1003, ¶¶ 95-98.

b. [1a]: “a primary storage facility for storing a primary copy of data, the primary storage facility including;”

The multi-site storage area network (SAN) (“data redundancy system”), includes multiple sites with storage cell(s) 101, and a storage area network (SAN) management appliance (SMA) 107 located at each site to operate storage cell(s) 101. Ex-1005, [0028] (describing a SAN with three sites physically separated from one

another, and each site including one or more storage cells 101), [0034] (describing SMAs 107 are typically located at each site to manage operation of local storage cells 101). And Lubbers teaches additional embodiments of multiple sites (including “primary storage facilities”), shown below:



Ex-1005, [0037]-[0040], FIG. 3 (annotated); Ex-1003, ¶ 99.

The primary site of Lubbers, including the controllers 105, storage management appliances 107, physical storage device 302, fibre channel arbitrated loop (FCAL) connections 303, switches 301, and host 102, is a “primary storage facility” because it is a “local” copy of the data. Ex-1003, ¶ 100; Ex-1001, 1:33-38 (referring to one copy of data as a ‘primary’ or ‘local’ copy, and a redundant copy

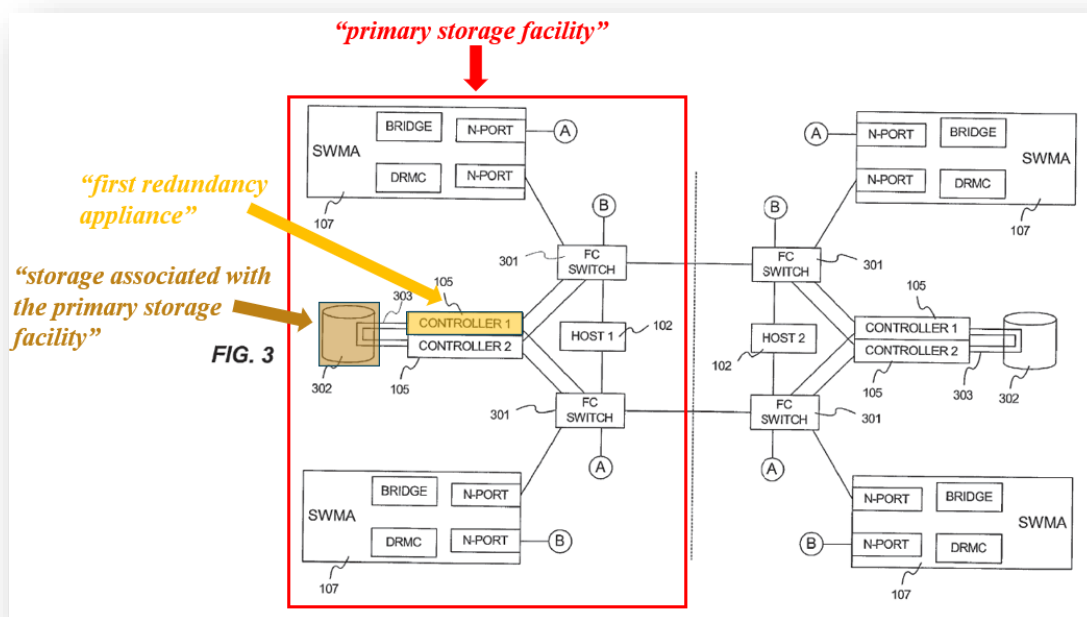
of the data as a ‘secondary’ or ‘slave’ copy); Ex-1005, [0037] (explaining storage management appliances 107 include bridge and data replication management consoles (DRMC) that manage controllers 105), [0040] (describing physical storage devices 302 coupled to controllers 105 by FCAL connections 303), [0041] (describing dual fabric implemented by switches 301, dual hosts 102, and dual storage topology).

The “*primary storage facility*” is “*for storing a primary copy of data* because it includes physical storage devices 302 coupled to controllers 105 by an FCAL connection allowing “each of the tens or hundreds of individual hard disks to be accessed not only at high speed, but by either controller in a storage cell.” Ex-1003, ¶ 101; Ex-1005, [0040]; *see also* Ex-1005, [0030] (explaining each site includes storage cells with hundreds of gigabytes to terabytes of physical storage capacity accessible by controllers 105), [0010] (explaining that SAN systems enable storing multiple copies of data at various physical locations throughout the system), [0020] (describing storage implemented at various sites is accessed by a host computer 102 that requires data storage), [0036] (explaining storage cell 101 can simultaneously be primary storage for a host and secondary storage for replicas of data from another storage cell), [0038] (describing how data processing can continue as an alternate site if a failure occurs at the primary site with little or no downtime).

Accordingly, Lubbers taught or suggested this element. Ex-1003, ¶¶ 99-102.

- c. *[1b]: a first redundancy appliance operable in a role for receiving a sequence of write requests and for storing data for the sequence of write requests in storage associated with the primary storage facility; and*

Lubbers teaches a first controller 105 (“*first redundancy appliance*”) at the primary site (“*primary storage facility*”) and physical storage device 302 (“*storage associated with the primary storage facility*”), shown below:



Ex-1005, FIG. 3 (annotated); Ex-1003, ¶ 104.

The '182 Patent uses the phrase “*a sequence of write requests*” to refer to sequential-in-time write requests (Ex-1001, 19:7-9), which the combined teachings of Lubbers and Sicola would have rendered obvious. Ex-1003, ¶ 105.

The first controller 105 is “*a first redundancy appliance*” because the first controller 105 is “*operable in a role for receiving a sequence of write requests.*” The

controllers 105 of Lubbers “are configured to handle both operational data transactions and data replication management (DRM) traffic.” Ex-1005, [0044]. After receiving sequential-in-time write requests from the host, the controllers 105 can pack “the log data from as many host writes as possible into a single buffer.” Ex-1005, [0083]; *see also* Ex-1005, [0011] (explaining “I/O commands and the data associated with those commands are written to the data log”), [0030] (explaining “controllers 105 virtualize the physical storage capacity such that it is configurable into logical units (LUNs) of storage capacity”), [0032] (explaining “[s]torage controllers 105 manage tasks of allocating storage capacity to specified LUNs, monitoring and maintaining integrity of LUNs, moving data between physical storage devices, and other functions that maintain integrity and availability” of data). Ex-1003, ¶ 106.

Sicola provides additional details of how a controller (*“first redundancy appliance”*) is *“in a role for receiving a sequence of write requests.”* Ex-1006, 11:45-47 (explaining, with reference to Figure 7, that “host computer 101 issues a write command to local controller A1 (201), which receives the command at host port 109 over path 221*h*”), 12:17-20 (same for Figure 8A). A POSITA would have been motivated to combine Lubbers and Sicola as discussed above. § VII.A; Ex-1003, ¶¶ 103, 107.

The physical storage devices 302 (e.g., hard disks) at the primary site of Lubbers are *“storage associated with the primary storage facility.”* Ex-1003, ¶ 108; Ex-1005, [0040] (explaining each physical storage device 302 can take the form of “tens or hundreds of individual hard disk drives”), [0030]-[0031] (explaining how “all data objects are identified with a universally unique identification (UUID)” stored persistently in metadata structures within physical storage, which can “be mapped and presented by any storage controller 105.”), [0032], [0005], [0011].

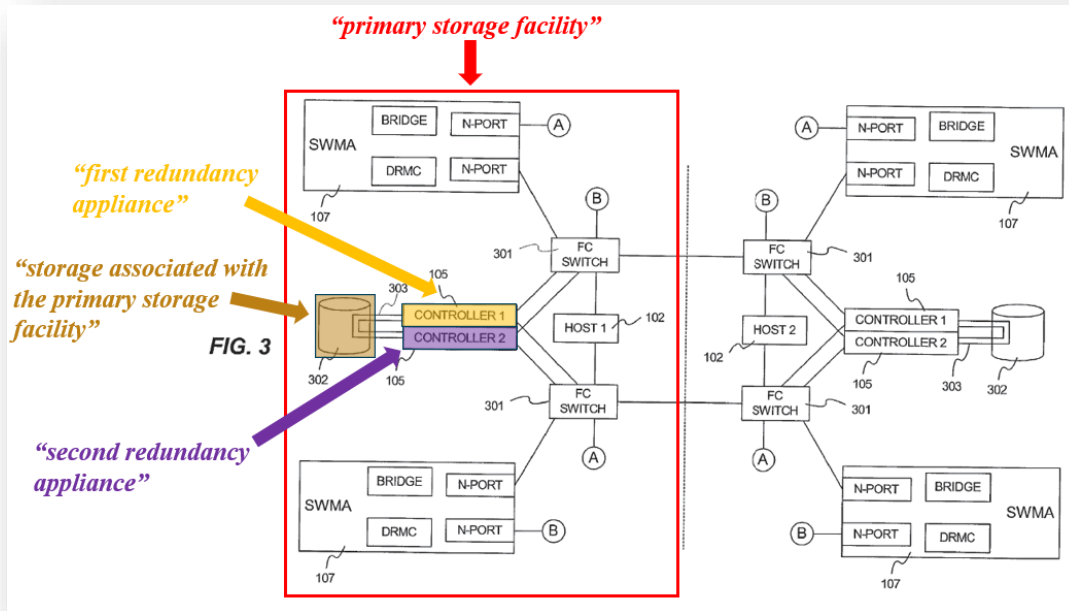
The first controller (*“a first redundancy appliance”*) is coupled to physical storage device 302 (*“storage associated with the primary storage facility”*) *“for storing data for the sequence of write requests.”* Ex-1003, ¶ 109; Ex-1005, [0040] (explaining “[p]hysical storage devices 302 are coupled to controllers 105 by a high-speed redundant connection such as fibre channel arbitrated loop (FCAL) connections 303”), [0045], [0054], [0076] (“During the data logging process, I/O operations from the host computer are copied into the log unit’s cache buffers”).

Accordingly, Lubbers and Sicola would have rendered obvious this element. Ex-1003, ¶¶ 103-110.

- d. *[1c]: “a second redundancy appliance for shadowing the first redundancy appliance wherein the second redundancy appliance assumes the role of the first redundancy appliance;”*

Lubbers teaches a second controller (controller 2) (*“second redundancy appliance”*) at the primary site (*“primary storage facility”*), shown below, which

uses the same reference numerals as other controllers 105 (implying the same structure and function).



Ex-1005, [0040], FIG. 3 (annotated); Ex-1003, ¶ 112.

The second controller (controller 2) is “a *second redundancy appliance*.” Ex-1003, ¶ 113; Ex-1005, [0040], [0045] (describing how “each controller 105 is identified by a storage cell unique identification number (UUID)”), [0044] (explaining how “controllers 105 are configured to handle both operational data transactions and data replication management (DRM) traffic”), [0083], [0030].

The ’182 Patent uses “*shadowing*” to refer to a redundancy appliance having “the same data and state information” as another redundancy appliance for when a fault occurs (Ex-1001, 21:42-44), and the “shadowing redundancy appliance 202”

takes over for the failed redundancy appliance (Ex-1001, 21:44-47). The combined teachings of Lubbers and Sicola would have rendered obvious “*shadowing*” as that term is used in the ’182 Patent. Ex-1003, ¶ 114.

The second controller of Lubbers (“*second redundancy appliance*”) is “*for shadowing*” and can “*assume the role of the first redundancy appliance.*” Ex-1003, ¶ 115; Ex-1005, [0031] (explaining “[a]ll objects in the system are identified with a universally unique identification (UUID),” which are “stored persistently in metadata structures within the physical storage which enables all or part of a storage cell, and all the LUNs implemented therein, to be mapped and presented by any storage controller”), [0038] (describing a “goal of a disaster tolerant configuration is to provide hardware redundancy for multiple levels of fault tolerance,” allowing for “failover” where a redundant component serves the function of a failed component at a site).

Sicola further describes how a “*secondary redundancy appliance*” is “*for shadowing*” and can “*assume the role of the first redundancy appliance.*” Ex-1003, ¶ 116; Ex-1006, 12:16-30, 13:6-13 (explaining a “micro-merge” process during asynchronous operation where “outstanding writes were logged to the initiator controller A1’s write-back cache, which is also mirrored in partner controller A2’s (mirrored) write-back cache, so that the cache data is available to controller A2 if controller A1 fails.”).

Even had Lubbers not expressly taught the second redundancy appliance “for shadowing” the first redundancy appliance, a POSITA would have found it obvious to use the second controller (“*second redundancy appliance*”) for shadowing the first controller (“*first redundancy appliance*”). Ex-1008, 711. A POSITA would have reasonably expected to successfully have the second controller (“*second redundancy appliance*”) shadow the first controller (“*first redundancy appliance*”) because the controllers and corresponding write-back caches of Lubbers and Sicola would merely be used for their intended functions to achieve predictable results. Ex-1008, 711. A POSITA would have been motivated to have the second controller shadow the first controller to allow for rapid recovery of user data from a disaster or significant error event at a data processing facility, as taught by Lubbers, and as discussed above. § VII.A; Ex-1003, ¶ 117.

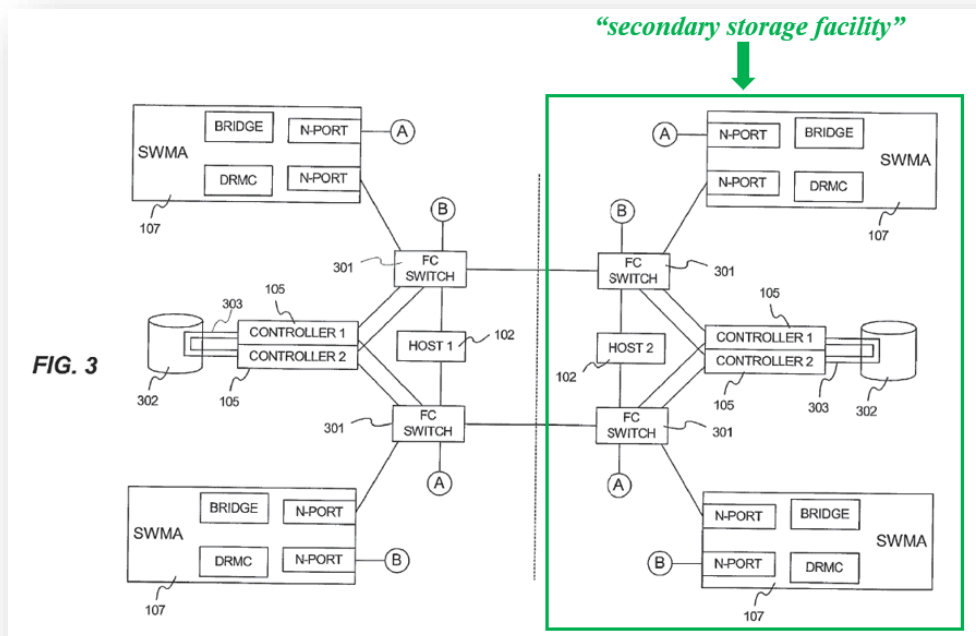
Accordingly, Lubbers and Sicola would have rendered obvious this element. Ex-1003, ¶¶ 111-118.

- e. ***[1d]: “a secondary storage facility for storing data that is redundant of the primary copy of the data, the secondary storage facility discrete from the primary storage facility and including:”***

Lubbers teaches that the multi-site storage area network can include any number of sites with the same, similar, or entirely different configurations. Ex-1003, ¶ 119.

The alternate (secondary) site of Lubbers, including controllers 105, storage management appliances 107 (SMAs A and B), physical storage device 302, fibre channel arbitrated loop (FCAL) connections 303, switches 301 (switches A and B), and host 102 (host 1), at the alternate (secondary), is a “*secondary storage facility*” of Lubbers. Ex-1003, ¶ 120; Ex-1005, [0037], [0040], [0041].

The “*secondary storage facility*” of Lubbers is shown below:



Ex-1005, FIG. 3 (annotated); Ex-1003, ¶ 121.

The “*secondary storage facility*” of Lubbers is “for storing data that is redundant of the primary copy of data” as discussed in element [1a]. Ex-1003, ¶ 122; Ex-1005, [0010], [0020], [0030], [0036], [0038], [0040].

The '182 Patent explains that “*discrete*” storage facilities refers to storage facilities physically separate from one another. Ex-1001, 4:42-58, 1:32-48. The teachings of Lubbers would have rendered obvious “*discrete*” storage facilities as that term is used in the '182 Patent. Ex-1003, ¶ 123.

Lubbers teaches “*the secondary storage facility discrete from the primary storage facility.*” Ex-1003, ¶ 124; Ex-1005, [0028] (explaining the “various sites may be physically ... separated by an arbitrary distance”), [0007] (describing a “safe physical distance between the original data and the data to [be] backed up” and that “[d]ata warehousing, ‘continuous computing’, and enterprise applications all benefit from remote copy capabilities”), [0026] (“the present invention may be implemented in a storage system that provides a peer-to-peer remote copy (backup) function”).

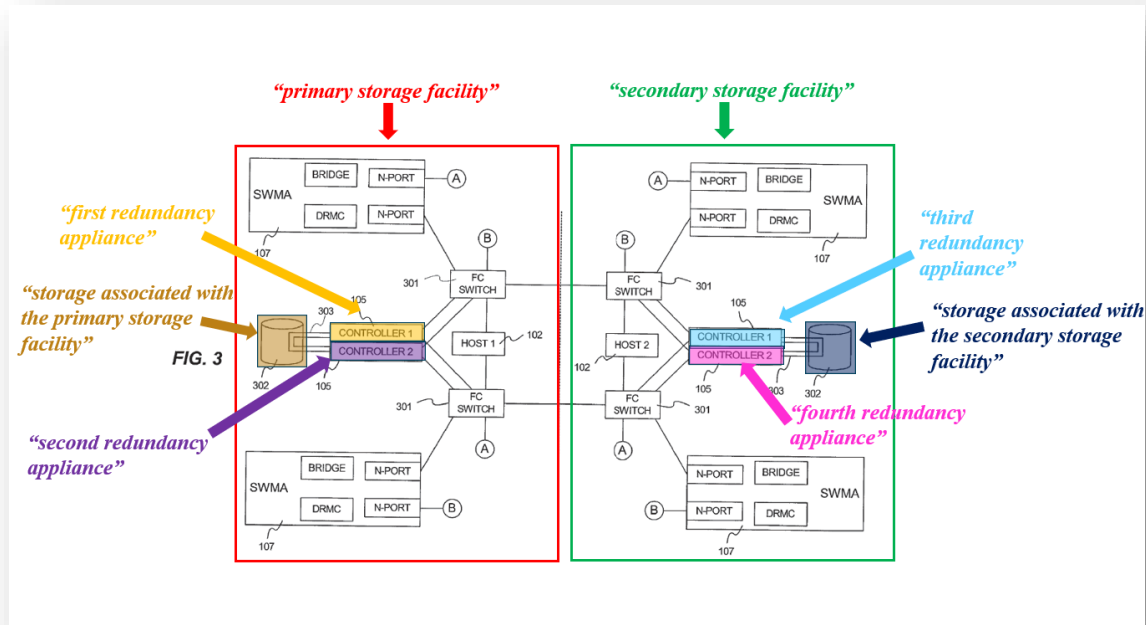
Accordingly, Lubbers taught or suggested this element. Ex-1003, ¶¶ 119-125.

- f. ***[1e]: “a third redundancy appliance operable in a role for storing redundant data for the sequence of write requests in storage associated with the secondary facility; and”***

Lubbers teaches a third controller 105 at an alternate site (e.g., another instance of the primary site). Ex-1005, [0037]. The third controller 105 is “*operable in a role for receiving a sequence of write requests*” as discussed for element [1b]. Ex-1003, ¶ 126

Annotated Figure 3 of Lubbers shows the “*primary storage facility*” that includes a first controller (“*first redundancy appliance*”) and a second controller

(*“second redundancy appliance”*), and the *“secondary storage facility”* includes a third controller (*“third redundancy appliance”*) and fourth controller (*“fourth redundancy appliance”*):



Ex-1005, FIG. 3 (annotated); Ex-1003, ¶ 127.

The physical storage devices 302 (e.g., multiple individual hard disks) at the alternate site of Lubbers is *“storage associated with the secondary storage facility”* as discussed for element [1b]. Ex-1003, ¶ 128.

Additionally, the third controller (*“third redundancy appliance”*) is coupled to the physical storage device 302 (*“storage associated with the secondary storage facility”*) *“for storing redundant data for the sequence of write requests”* as discussed in element [1b]. Further, as discussed for element [1c], the controllers and other

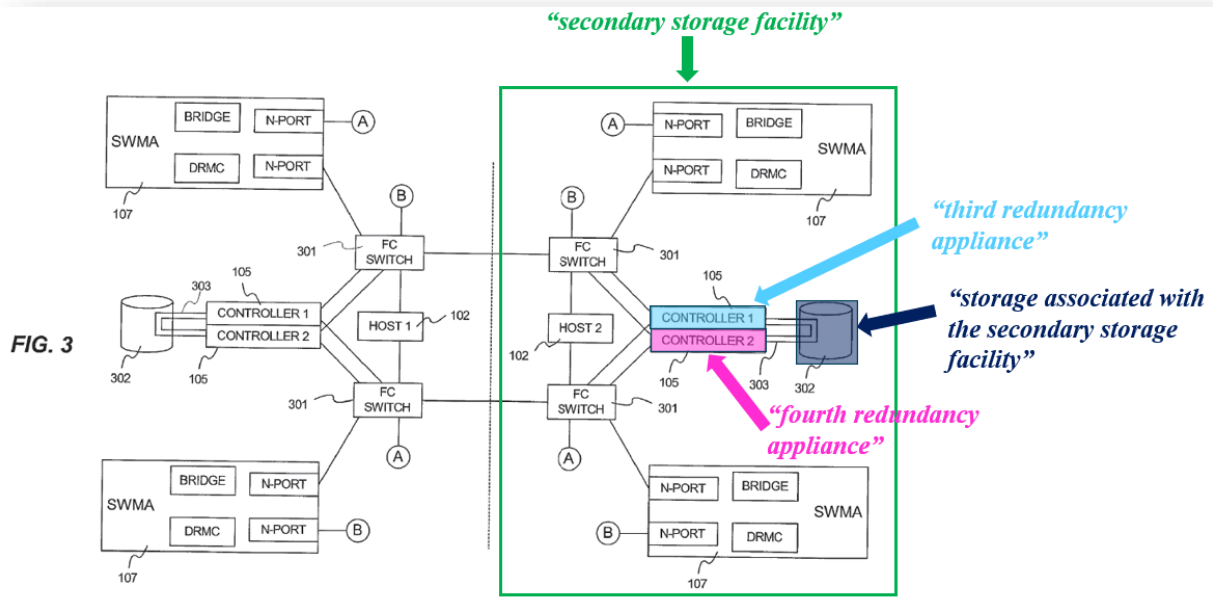
components use the same reference numerals, implying the same structure and functions. Ex-1003, ¶ 129.

Accordingly, Lubbers taught or suggested this element. Ex-1003, ¶¶ 126-130.

- g. *[1f]: “a fourth redundancy appliance for shadowing the third redundancy appliance wherein the fourth redundancy appliance assumes the role of the third redundancy appliance in the event of a fault at the third redundancy appliance.”*

Lubbers teaches a fourth controller 105 at the alternate site (“*secondary storage facility*”) is coupled to physical storage device 302 by an FCAL connection 303. Ex-1005, [0037]. The fourth controller is “*a fourth redundancy appliance*” as discussed for element [1c]. Ex-1003, ¶ 131.

The fourth controller 105 (“*fourth redundancy appliance*”) described in Lubbers is shown below, which uses the same reference numerals as other controllers 105 (implying the same structure and function):



Ex-1005, [0040], FIG. 3 (annotated); Ex-1003, ¶ 132.

The fourth controller of Lubbers (“*fourth redundancy appliance*”) is “*for shadowing*” and can “*assume the role of the third redundancy appliance in the event of a fault at the third redundancy appliance*” as discussed in element [1c]. The third and fourth redundancy appliances taught by Lubbers operate in the same “*shadowing*” arrangement described for the first and second redundancy appliances.

Ex-1003, ¶ 133.

Accordingly, Lubbers taught or suggested this element. Ex-1003, ¶ 131-134.

2. Claim 2:

- a. ***“The data redundancy system according to claim 1, wherein the secondary storage facility assumes the role of the primary storage facility in response to a failover event,”***

Lubbers teaches the alternate site (“*secondary storage facility*”) assumes the role of the primary site (“*primary storage facility*”) to provide system recovery after a disaster event or significant error (“*in response to a failover event*”). Ex-1003, ¶¶ 135-136; Ex-1005, [0007] (explaining it “is desirable to provide the ability for rapid recovery of user data from a disaster or significant error event at a data processing facility,” termed “disaster tolerance,” which allows “providing for replicated data and redundant storage to support recovery after the event”), [0038] (explaining a disaster tolerant configuration allows data processing to resume at an alternate site if a failure happens at the primary site).

- b. ***“the failover event including a communication traffic condition having a greater portion of request traffic originating closer to the secondary storage facility than to the primary storage facility.”***

Lubbers would have rendered obvious this element for several reasons. Ex-1003, ¶¶ 137-139. First, as discussed for element [1d], Lubbers teaches “*the secondary storage facility* discrete from the *primary storage facility*.” Second, Lubbers explains the benefits of load balancing and added complexities of increasing transmission distances. Ex-1005, [0006] (describing how prior storage systems “provided minimal load balancing benefit that would enable data access requests to

be directed intelligently to replicas such that resources were used more efficiently”), [0041] (explaining that longer fibre channels increases “both the latency and the variability in latency”), [0042] (describing that “[h]osts 102 preferably run multi-pathing software that dynamically allows failover between storage paths as well as static load balancing of storage volumes (LUNs) between the path of controllers 102,” which “enables a host 102 to identify multiple connection options that lead to desired storage, and select amongst the available paths based on selected criteria” such as “path availability, path performance, path load, controller load, and the like.”), [0064] (explaining “the present invention allows great flexibility in implementing a particular system to emphasize any balance of, for example, low latency, high availability, disaster tolerance, and cost”).

Thus, Lubbers teaches criteria like latency, path performance, and path load in balancing loads between storage volumes at different sites. Based on these criteria, a POSITA would have found it obvious to use a failover condition where the primary site and alternate site change roles in response to the secondary site being located closer to a host computer 102 generating write and read requests. A POSITA would have been motivated to consider this type of failover condition to minimize latency and improve data transmission path performance, as taught by Lubbers. Ex-1003, ¶ 140.

A POSITA would have reasonably expected to successfully use this type of failover condition for the multi-site storage area network (SAN) of Lubbers because Lubbers already contemplates flexible assignment of sites as primary and secondary (Ex-1005, [0036]), site failover situations like a communication link failure between controllers and where storage cells are “inaccessible” to sites (Ex-1005, [0068]), and teaches site failover can occur automatically (Ex-1005, [0022]). Thus, using this type of failover condition for the SAN of Lubbers would have been a routine matter well within a POSITA’s ability to achieve predictable results. Ex-1003, ¶ 141.

Accordingly, Lubbers and Sicola would have rendered obvious claim 2. Ex-1003, ¶¶ 135-142.

3. Claim 3: *The data redundancy system according to claim 1, wherein the second redundancy appliance assumes the role of the first redundancy appliance in response to a failover event selected from the group consisting of a manually initiated event and an automatically initiated event.*

As discussed for element [1c], Lubbers teaches “*wherein the second redundancy appliance assumes the role of the first redundancy appliance.*” Ex-1003, ¶ 144.

Lubbers teaches that the second controller (“*second redundancy appliance*”) assumes the role of the first controller (“*first redundancy appliance*”) automatically upon failover (“*in response to a failover event selected from the group consisting of a manually initiated event and automatically initiated event*”). Ex-1003, ¶ 145; Ex-

1005, [0046] (explaining when “a controller fails, the other controller presents the LUNs that were being presented on the failed controller,” and can “continue data replication operations without loss of data or requiring manual process.”).

Accordingly, Lubbers and Sicola would have rendered obvious claim 3. Ex-1003, ¶ 143-146.

4. Claim 4:

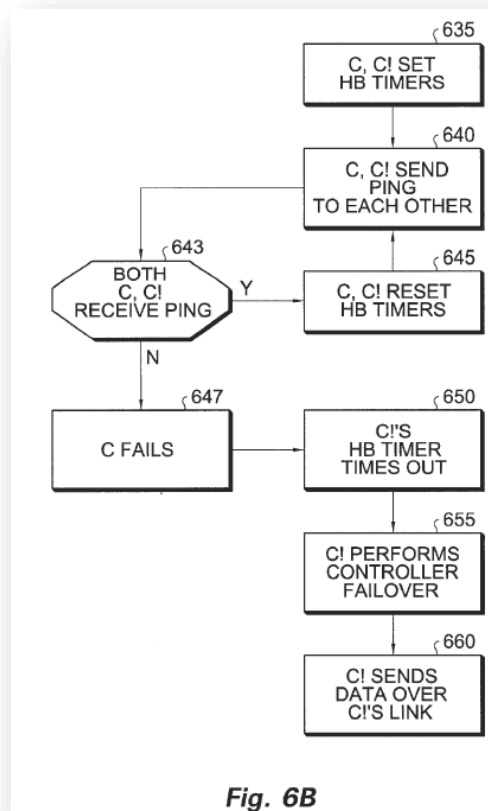
- a. *“The data redundancy system according to claim 1, wherein the first redundancy appliance updates the second redundancy appliance with information about the data in storage by forwarding a write record to the second redundancy appliance for each write request,”*

Lubbers and Sicola would have rendered obvious this element. Ex-1003, ¶¶ 147-150. First, Lubbers teaches that the first controller (*“first redundancy appliance”*) updates the second controller (*“second redundancy appliance”*) *“with information about the data in storage”* by moving data between physical storage devices. Ex-1005, [0032], [0073] (describing that if a “controller does not have the most current data, then the controller goes offline to the host to receive a full copy from the controller that assumes the role of the source controller (step 830)”). Second, Sicola teaches that the *“updates”* occur *“by forwarding a write record to the second redundancy appliance for each write request”* as discussed above in element [1c]. Ex-1006, 12:16-30, 13:6-13.

b. “the second redundancy appliance sending acknowledgements to the first redundancy appliance in batches in response to receiving write records.”

Lubbers and Sicola also would have rendered obvious this element. Ex-1003, ¶ 151; Ex-1005, [0068] (explaining each “controller periodically sends ‘heartbeats,’ to and receives ‘heartbeats’ from peer controllers,” which are pings “with a message including information useful in managing data transfer processes between peer controllers”), [0069] (describing “a full-duplex path between one controller and another is referred to as a ‘tunnel’”), [0072] (explaining a “controller determines whether the data under the supervision of the peer controllers is consistent ... by comparing GSNs of the various controllers”), claim 2 (explaining a “step of sending the host computer a write completion status message prior to sending the data to the destination storage system), claim 11 (same).

Figure 6B of Sicola shows and describes that these pings and responsive information are “*acknowledgements ... “in response to receiving write records”*”:



Ex-1006, FIG 6B, 10:22-26 (explaining intra-site controller heartbeat timer operation), 10:45-53 (describing that during normal operation, at step 640, companion controllers “periodically send pings to each other via DUARTs (Dual Asynchronous Receiver/Transmitters),” and the pings are received unless either controller has timed out); Ex-1003, ¶ 152.

Thus, Lubbers teaches component redundancies for multiple levels of fault tolerance, allowing failover to a redundant component at the same site for continued operation, storing write commands in controller write-back caches, and using criteria to balance controller load. Ex-1003, ¶ 153. Based on these teachings, a POSITA

would have found it obvious to use the first controller (“*the first redundancy appliance*”) to update a second (redundant) controller (“*the second redundancy appliance*”) with information about the data in physical storage device 302 (“*storage*”) by forwarding a write request or log (“*write record*”) to the second controller (“*second redundancy appliance*”) for each write request, where the second controller (“*second redundancy appliance*”) sends pings (“*acknowledgements*”) to the first controller (“*first redundancy appliance*”) in batches in response to receiving write records.” Ex-1003, ¶ 154. Further, a POSITA would have been motivated to consider forwarding write records and sending return acknowledgements in this manner to maintain integrity and availability of data, while also minimizing controller load, as taught by Lubbers. Ex-1003, ¶ 155.

Accordingly, Lubbers and Sicola would have rendered obvious claim 4. Ex-1003, ¶¶ 147-156.

5. Claim 5: *The data redundancy system according to claim 4, wherein the first redundancy appliance sends an acknowledgement to an originator of the write request in response to receiving the acknowledgement from the second redundancy appliance.*

Lubbers and Sicola would have rendered obvious “*wherein the first redundancy appliance sends an acknowledgement to an originator of the write request in response to receiving the acknowledgement from the second redundancy appliance*” by teaching controllers communicating through and responding to pings

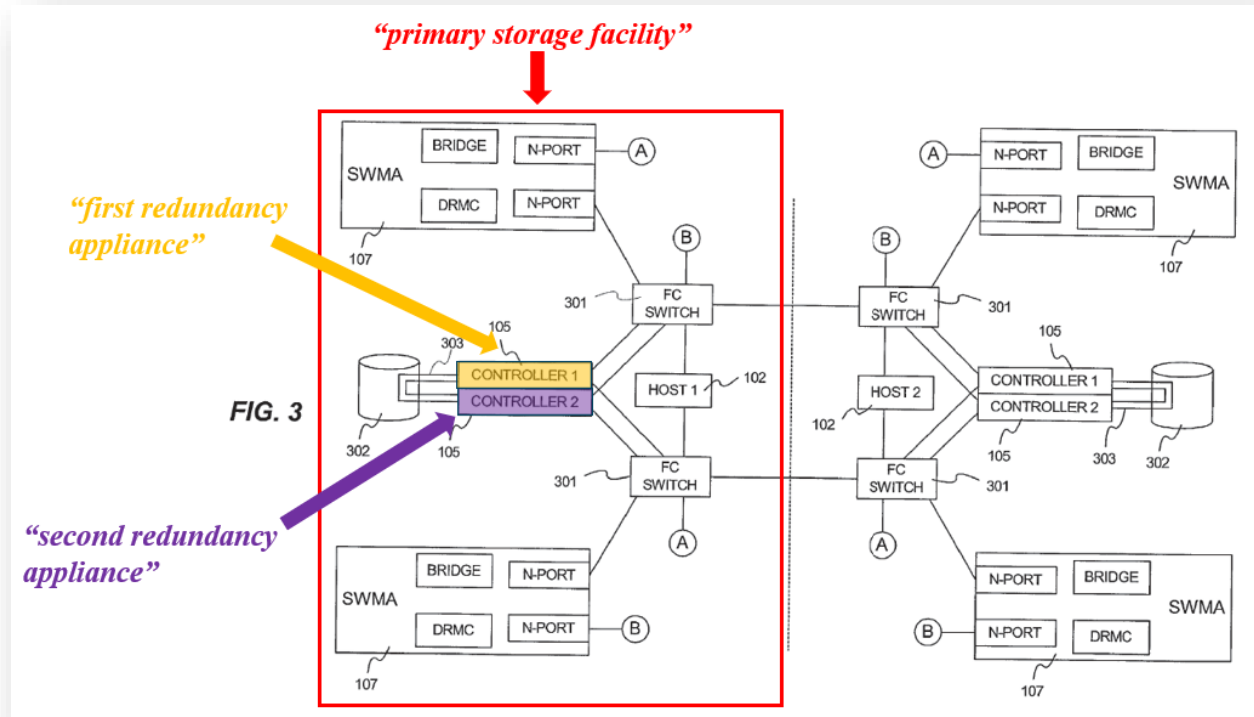
and sending write completion status messages to the host computer. Ex-1003, ¶¶ 157-158; Ex-1005, claim 2 (explaining a “step of sending the host computer a write completion status message prior to sending the data to the destination storage system”), claim 11 (same), [0068] (explaining each “controller periodically sends ‘heartbeats,’ to and receives ‘heartbeats’ from peer controllers,” which are pings “with a message including information useful in managing data transfer processes between peer controllers”). Sicola teaches that when the host write request is complete, “the host is given the completion status.” Ex-1005, 12:44-46. Given the teachings of Lubbers and Sicola, a POSITA would have recognized the pings could contain the write completion status information to synchronize the controllers (for redundancy) before relaying the write completion status to the host. Ex-1003, ¶ 159.

Accordingly, Lubbers and Sicola would have rendered obvious claim 5. Ex-1003, ¶¶ 157-160.

6. Claim 6: *The data redundancy system according to claim 1, wherein the first redundancy appliance is disjoint from the second redundancy appliance.*

The ’182 Patent refers to a redundancy appliance as “*disjoint*” from another redundancy appliance where “it preferably does not share common hardware or software elements, or common failure modes” (Ex-1001, 21:23-26). The combined teachings of Lubbers and Sicola would have rendered obvious “*disjoint*” redundancy appliances as that term is used in the ’182 Patent. Ex-1003, ¶¶ 161-162.

The first controller (“*first redundancy appliance*”) is “*disjoint*” from the second controller (“*second redundancy appliance*”) of Lubbers because they do not share common hardware or software elements:



Ex-1003, ¶ 163; Ex-1005, FIG. 3 (annotated), [0037] (emphasizing “redundancy in physical components”), [0038] (describing a disaster tolerant configuration “to provide hardware redundancy for multiple levels of fault tolerance ... to keep the primary and alternate sites running, almost at ‘non-stop’ metrics”), [0026] (explaining controller-based replication “between two remotely separated pairs of array controllers”), [0039]-[0040] (describing how each storage device 302 is accessibly by either controller).

Accordingly, Lubbers and Sicola would have rendered obvious claim 6. Ex-1003, ¶¶ 161-164.

7. **Claim 7: *The data redundancy system according to claim 1, wherein the second redundancy appliance maintains a shadow copy of data stored by the first redundancy appliance wherein the second redundancy appliance accesses the shadow copy of the data when the second redundancy appliance assumes the role of the first redundancy appliance.***

The second controller (“*second redundancy appliance*”) is “for shadowing,” assuming the role of, and independently performing the same functions as, the first controller (“*first redundancy appliance*”) as discussed in element [1c]. Ex-1003, ¶ 166.

Lubbers teaches that the second controller (“*second redundancy appliance*”) “maintains a shadow copy of data stored by the first redundancy appliance” and “accesses the shadow copy of the data” when the second controller assumes the role of the first controller. Operating independently includes the second controller (“*second redundancy appliance*”) using the “shadow copy” of data from the first controller (“*first redundancy appliance*”), particularly during a failover event where role assumption becomes necessary. Further, Sicola teaches that both controllers in a redundant controller pair can store their own copy of data in their own separate write-back cache as discussed above in element [1c]. Ex-1003, ¶ 167; Ex-1006, 12:16-30, 13:6-13.

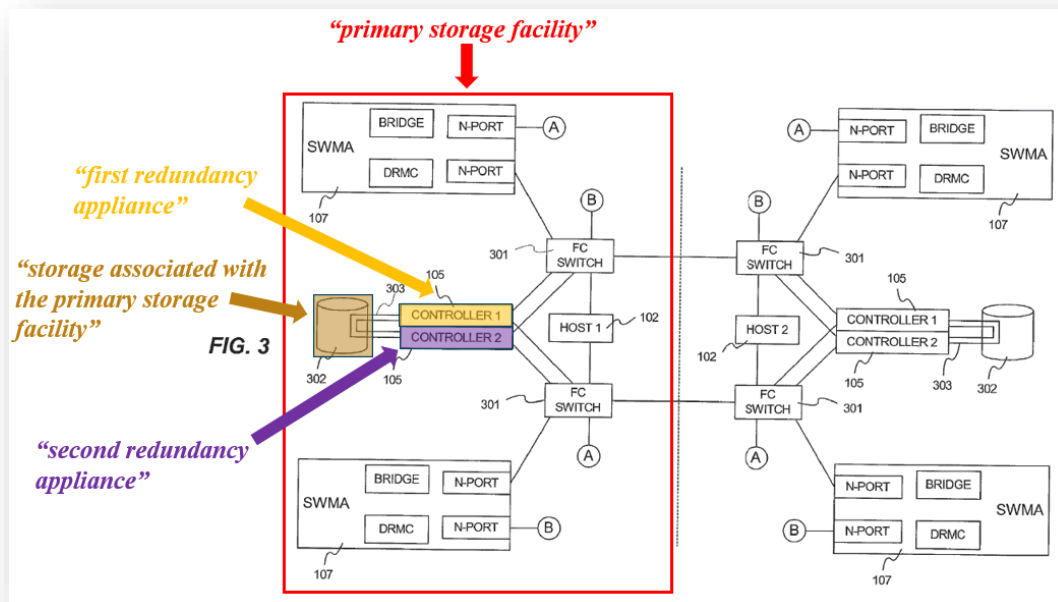
Thus, Lubbers teaches component redundancies for multiple levels of fault tolerance, allowing failover to a redundant component at the same site for continued operation, storing write commands in controller write-back caches, and using criteria to balance controller load. A POSITA would have found it obvious to use a second (redundant) controller to maintain a copy of the write-back cache data stored by the first controller (“*wherein the second redundancy appliance maintains a shadow copy of data stored by the first redundancy appliance*”) and to use the second controller to access that write-back cache data when the second controller assumes the role of the first controller (e.g., due to failover) (“*wherein the second redundancy appliance accesses the shadow copy of the data when the second redundancy appliance assumes the role of the first redundancy appliance*”). Ex-1008, 711. A POSITA would have been motivated to use a second controller to maintain a copy of the write-back cache data in this manner to facilitate asynchronous operation and maintain data integrity when the first controller fails in the window where the host write status has already returned to the host, but remote operation has not yet completed, as taught by Sicola. Ex-1003, ¶ 168; Ex-1008, 711.

Accordingly, Lubbers and Sicola would have rendered obvious claim 7. Ex-1003, ¶¶ 165-169.

8. **Claim 8:** *The data redundancy system according to claim 1, wherein the second redundancy appliance accesses the data stored in storage by the first redundancy appliance when the second redundancy appliance acts as the first redundancy appliance in response to a failover event.*

The second controller (“*second redundancy appliance*”) of Lubbers assumes the role of, and can independently perform the same functions as, the first controller (“*first redundancy appliance*”) in the event of a first controller failure (“*in response to a failover event*”), as discussed in element [1c]. Ex-1003, ¶ 170.

Lubbers also teaches the first controller (“*first redundancy appliance*”) and second controller (“*second redundancy appliance*”) can each store data in physical storage devices 302 (“*storage associated with the primary storage facility*”) as discussed in elements [1b] and [1c], shown below:



Ex-1005, FIG. 3 (annotated). Lubbers teaches that the second controller (“*second redundancy appliance*”) can access data in the same physical storage device 302 (“*storage associated with the primary storage facility*”) written to by the first controller (“*first redundancy appliance*”), particularly when the controllers swap roles. Ex-1003, ¶ 172; Ex-1005, [0040] (describing access to the individual hard disk drives “by either controller 105”), [0038], [0046] (describing that “[w]hen a controller fails, the other controller presents the LUNs that were being presented on the failed controller”), [0020].

Accordingly, Lubbers and Sicola would have rendered obvious claim 8. Ex-1003, ¶¶ 170-173.

B. Ground 2: Claims 9-18 are Obvious Over Lubbers, Sicola and Ofek

As discussed above, Lubbers and Sicola would have rendered obvious every element of claims 1-8. Lubbers, Sicola, and Ofek would have rendered obvious every element of claims 9-18, as explained below. Ex-1003, ¶ 174.

It would have been obvious to combine Ofek’s teachings to bundle write data for transmission to the secondary data storage system with the teachings of Lubbers and Sicola (“*redundant data to the secondary storage facility as send batches*”). Lubbers teaches “a data logging mechanism (e.g., a write history log unit) for storing commands and associated data for I/O transactions that occur when the remote

backup storage device is unavailable.” Ex-1005, [0026] (explaining that when “the destination site becomes available, the system performs an in-order merging of the log unit data with the data on the previously unavailable backup device to quickly return the source and destination sites to the same data state after link restoration or destination site restoration”), [0060] (describing “I/O commands and the data associated with those commands are written to the data log, and after destination becomes available the information in the data log is merged into the destination volume to conform the data in the destination volume to the data in the source volume”), [0075] (describing how the “source controller may begin logging”), [0076] (explaining that during “the data logging process, I/O operations from the host computer are copied into the log unit’s cache buffers”). In other words, Lubbers already teaches storing multiple transactions (“*batches*”) of transactions in a log, which is then sent and merged with the data in the alternate site. Ex-1003, ¶ 175.

Similarly, Ofek teaches that “[i]n order to more effectively use the capacity of the data link, the data storage system containing the primary (R1) volume bundles the write data for all write commands in the chain into a single write command for transmission over the data link to the secondary data storage system containing the secondary (R2) volume.” Ex-1003, ¶ 176; Ex-1007; *see also* Ex-1007, 34:66-35:6 (it is very desirable to ‘bundle’ the write data for all write commands in the channel

command word chain into a single write command transmitted over a link to the secondary storage system 246.”).

A POSITA would have been motivated to combine the teachings of Lubbers and Sicola and analogous write bundling taught by Ofek for several reasons. First, Lubbers and Ofek teach effectively using bandwidth and minimizing latency. Ex-1003, ¶ 177; Ex-1005, [0041] (describing how “both latency and the variability in latency increases with more complex fabrics”), [0042] (explaining load balancing based on criteria such as “path availability, path performance, path load, controller load, and the like”), [0042]; Ex-1007, Abstract (explaining that “to more effectively use the capacity of the data link, the data storage system containing the primary (R1) volume bundles the write data for all write commands in the chain into a single write command for transmission over the data link to the secondary data storage system containing the secondary (R2) volume”).

Second, both Lubbers and Ofek teach a goal of maintaining data integrity. Ex-1003, ¶ 178; Ex-1005, [0032] (explaining “[s]torage controllers 105 manage the tasks of allocating physical capacity to specified LUNs, moving data between physical storage devices, and other functions that maintain integrity and availability of the data stored therein”); Ex-1007, 3:10-18 (explaining the data storage system “achieves nearly 100 percent data integrity by assuring that all data is copied to a geographically remote site, and in those cases when a back-up copy is not made due

to an error of any sort, an indication is stored that the data has not been copied, but instead must be updated at a future time.”), 35:60-36:2 (describing that “the entry in the link queue 504 includes a one-byte password for confirming the validity of the entry”), 37:17-24 (explaining “[e]ach link adapter scans the link queue 504 in an iterative loop, looking for unlocked entries to service, beginning at the head of the queue,” and then “locks the next entry to service, checks the password to determine if the entry is valid, and if so, gets the buffer pointer from the entry, reads the buffer, and builds a job to be executed for transferring data from cache across the link in a direct memory access (DMA) operation”). A POSITA would have been motivated to combine the teachings of Ofek and Lubbers and Sicola to include data integrity assurances like object checking and verification for maintaining accurate data in a backup system. Ex-1003, ¶ 178.

A POSITA would recognize that bundling writes into a single write transaction for transmission, taught by Ofek, would also use a cache memory with a buffer rather than a transaction log, which Lubbers and Sicola already teach. Ex-1003, ¶ 182; Ex-1005, [0032] (explaining “how hosts 102 access physical storage capacity by addressing read and write operations to specified LUNs” (logical units) using the worldwide LUN ID (WWLID)).

There are multiple reasons why a POSITA would have been motivated to combine the teachings of Lubbers and Sicola with Ofek. For example, a POSITA

would have been motivated to combine batching and sending write commands, as taught by Ofek, with the teachings of Lubbers and Sicola to minimize the effects of latency and improve data transmission path performance for backup data systems. Ex-1003, ¶ 182. A POSITA would have reasonably expected to successfully have a first controller batch and send write commands in this manner because Lubbers and Sicola already contemplate including multiple write requests in logs before transmitting them to be merged into the alternate site, (Ex-1005, [0060], Ex-1006, 10:22-53), and Ofek contemplates using chains of channel commands bundling multiple write commands (Ex-1007, Abstract, 34:66–35:6). Ex-1003, ¶ 182. Thus, sending multiple write commands in a batch to an alternate site would have been a routine matter well within a POSITA’s ability to achieve predictable results. Ex-1003, ¶ 182.

It also would have been obvious to bundle write data for write commands (“*batches*”) based on quantity of write commands, taught by Lubbers and Sicola, or based on time elapsed between batches, taught by Ofek to achieve the goals of minimizing latency and improving data transmission and path performance. Ex-1003, ¶ 183. For at least these same reasons, a POSITA would have found it obvious to adaptively size bundles of write data for write commands (“*batches*”) based on availability of communication bandwidth. Ex-1003, ¶ 183.

A POSITA would have been motivated to combine teachings for adaptively assessing batch size to balance the effects of latency, high availability, disaster tolerance, data transmission path performance, and system performance, speed, and cost. Ex-1003, ¶ 184. A POSITA would have reasonably expected to successfully have a first controller determine the size of send batches in this manner because Lubbers and Sicola already contemplate different modes of operation to focus on certain of the competing considerations, and Ofek likewise includes a user-configurable skew parameter. Ex-1003, ¶ 184; Ex-1007, 17:39-41.

A POSITA would have found it obvious to replace a first write transaction in a bundle of write data (“*send batch*”) at a primary site (“*primary storage facility*”) by overwriting it with a second write transaction in the bundle of write data. Lubbers teaches ordering and sequencing write commands such that a write transaction can be overwritten. Ex-1003, ¶ 185; Ex-1005, [0060] (describing an ordering algorithm that “uses a ‘group sequence number’ and the remote groups 705 ensure that the data is written in order sequence”), [0072] (explaining “the controller determines whether the data under the supervision of the peer controllers is consistent” by “comparing GSNs of the various controllers” and that if “the GSNs differ, then the data states of various controllers differ”). Further, Lubbers describes a circumstance where overwriting is beneficial. Ex-1003, ¶ 185; Ex-1005, [0072]-[0073] (if “the controller does not have the most current data, then the controller goes offline to the host to

receive a full copy from the controller that assumes the role of the source controller.”).

Sicola provides further details on overwriting write transactions in a batch in a multi-site storage system. Ex-1003, ¶ 186; Ex-1006, 12:16-30 (describing asynchronous operation involving a “micro-logging” process where the controller “writes the data into its write-back cache”), 13:6-13 (explaining a “micro-merge” process during asynchronous operation where “outstanding writes were logged to the initiator controller A1’s write-back cache, which is also mirrored in partner controller A2’s (mirrored) write-back cache, so that the cache data is available to controller A2 if controller A1 fails.”).

Ofek teaches a more specific overwriting procedure. Ex-1007, 39:28-31 (the channel adapter first checks whether the track is on disk, which is then copied from disk to the cache slot in cache); Ex-1007, 39:41-43 (if the track does reside in cache, then an “overwrite cache option” for the volume can be executed); Ex-1007, 39:43-52 (“if the cache contains a remote write pending record that has not yet been transmitted by a link adapter to the data storage system having the secondary volume, then a new version will overwrite this write pending in cache.”). Further, Ofek teaches that “[i]n the adaptive copy modes, however, the overwrite cache option substantially increases the performance of remote mirroring of data, by

reducing the number of remote pending records that are maintained in cache or on disk.” Ex-1003, ¶ 187; Ex-1007, 39:52-56.

A POSITA would have been motivated to combine the teachings of Lubbers, Sicola, and Ofek to enable overwriting write transactions in a cache before or after transmitting and storing the transactions to improve the performance of remote mirroring data, by reducing the number of remote pending records that are maintained in cache or on disk. Ex-1003, ¶ 188.

A POSITA would have reasonably expected to successfully overwrite write transactions within a send batch or in combinations of send batches at either the primary site or secondary sites because Lubbers and Sicola already contemplate replacing controller data with the most current data, Ex-1005, [0073]; Ex-1006, 12:16–30, 13:6–13, and Ofek contemplates overwriting transactions within bundles of write transactions in a cache. Ex-1007, 37:64–67, 39:22–56. Ex-1003, ¶ 189. Thus, overwriting write transactions within a send batch at the primary site would have been a routine matter well within a POSITA’s ability to achieve predictable results. Ex-1003, ¶ 189. Additionally, a POSITA would have reasonably expected to successfully combine receive batches and having write transactions overwrite other write transactions in a receive batch at the alternate site because Lubbers and Sicola already contemplates replacing controller data with the most current data, Ex-1005, [0073]; Ex-1006, 12:16–30, 13:6–13, and Ofek contemplates overwriting

transactions within bundles of write transactions in a cache, Ex-1007, 35:13–24, 36:13–16, 40:21–28. Ex-1003, ¶ 189.

As explained above, the same technique for bundling writes to create “*batches*” and overwriting transactions would apply equally to batches being sent (“*send batches*”) from the primary site (“*primary storage facility*”) and being received (“*receive batches*”) at the alternate site (“*secondary storage facility*”). Ex-1003, ¶ 190; *see e.g.*, Ex-1005, [0026]; Ex-1007, 35:13–16, 37:33–37:35.

A POSITA would have reasonably expected to successfully have the third controller (“*third redundancy appliance*”) determine data is safe to apply before writing it to storage at the alternate site (“*secondary storage facility*”) because Lubbers already teaches the importance of data reliability and integrity, and using LUNs, UUIDs, Ex-1005, [0032], and Ofek further teaches using a one-byte password (among other things) to confirm the validity of write requests before writing them to storage, Ex-1007, 35:60–36:2. Ex-1003, ¶ 191. Thus, determining write requests in a receive batch are safe to apply before writing those requests to storage at the alternate site (“*secondary storage facility*”) would have been a routine matter well within a POSITA’s ability to achieve predictable results. Ex-1003, ¶ 191.

1. Claim 9: *The data redundancy system according to claim 1, wherein the first redundancy appliance forwards redundant data to the secondary storage facility as send batches.*

Lubbers, Sicola, and Ofek would have rendered obvious claim 9. A POSITA would have been motivated to combine Lubbers, Sicola, and Ofek as discussed above. §VII.B; Ex-1003, ¶ 192.

Lubbers teaches the first controller (“*first redundancy appliance*”) accesses and transfers a replica of the data (“*redundant data*”) to the alternate site (“*secondary storage facility*”) as discussed in element [1d]. Ex-1003, ¶ 193.

It would have been obvious to combine Ofek’s teachings for bundling write commands and forwarding “*redundant data to the secondary storage facility as send batches*” with the teachings of Lubbers and Sicola. Ex-1003, ¶ 194. Lubbers already teaches batching write requests in the form of transaction logs. Ex-1005, [0060] (explaining “the log is sized to store all write transactions until the transaction is committed to each member of a copy set” and that “I/O commands and the data associated with those commands are written to the data log, and after destination becomes available the information in the data log is merged into the destination volume to conform the data in the destination volume to the data in the source volume”).

A POSITA would recognize that bundling write requests for transmission would similarly involve using a cache memory, but would merely involve gathering

the data in a buffer rather than a transaction log, which Lubbers already teaches. Ex-1003, ¶ 196. Like Lubbers, Ofek teaches “*send batches*.” Ex-1007, Abstract (explaining that “to more effectively use the capacity of the data link, the data storage system containing the primary (R1) volume bundles the write data for all write commands in the chain into a single write command for transmission over the data link to the secondary data storage system containing the secondary (R2) volume”), 34:66-35:6 (describing it is “desirable to ‘bundle’ the write data for all write commands in the channel command word chain into a single write command transmitted over a link to the secondary storage system”).

A POSITA would have been motivated to combine the teachings of Lubbers and Ofek to balance considerations like latency, path performance, and load capacity, taught by Lubbers. Ex-1003, ¶ 198. A POSITA would have reasonably expected to successfully combine the teachings of Lubbers and Ofek in this manner because Lubbers already teaches sending batched transactions via transaction logs. Ex-1003, ¶ 199.

Accordingly, the combined teachings of Lubbers, Sicola, and Ofek would have rendered obvious claim 9. Ex-1003, ¶¶ 192-200.

2. **Claim 10:** *The data redundancy system according to claim 9, wherein a send batch size is determined according to either a time interval for collecting transactions into the send batch or a predetermined quantity of transactions to be included in the send batch.*

Lubbers, Sicola, and Ofek would have rendered obvious claim 10. A POSITA would have been motivated to combine Lubbers, Sicola, and Ofek as discussed above. §VII.B; Ex-1003, ¶ 201.

Lubbers teaches using “a time delay” or “a predetermined quantity of transactions” to determine “send batch size.” Ex-1005, [0011] (describing that “[d]uring the merge process the system maintains a ratio of merge data writes out of the data log to data writes from a host computer into the data log within a desired range until the data log reaches a predetermined size, [0059] (describing “asynchronous ... operation mode in which a modification to one member of a copy set can be propagated to other members of the copy set after a time delay”), [0081]. Thus, based on the teachings of Lubbers, a POSITA would have reasonably expected to successfully combine the teachings of Lubbers, Sicola, and Ofek to achieve predictable results. Ex-1003, ¶¶ 202-203.

Accordingly, the combined teachings of Lubbers, Sicola, and Ofek would have rendered obvious claim 10. Ex-1003, ¶¶ 201-204.

3. Claim 11: *The data redundancy system according to claim 9, wherein a send batch size is determined adaptively based on available communication bandwidth between the primary storage facility and the secondary storage facility.*

Lubbers, Sicola, and Ofek would have rendered obvious claim 11. A POSITA would have been motivated to combine Lubbers, Sicola, and Ofek as discussed above. §VII.B; Ex-1003, ¶ 205.

It would have been obvious to combine Ofek's teachings to “*adaptively*” determine a “*send batch size*” based on “*available communication bandwidth*” with the teachings of Lubbers and Sicola. Ex-1003, ¶ 206. Lubbers already teaches different modes of operations and balancing competing criteria for system performance. Ex-1003, ¶ 207; Ex-1005, [0042] (explaining “[m]ulti-pathing software enables a host 102 to identify multiple connection options that lead to desired storage and select amongst the available paths based on selected criteria,” including “path availability, path performance, path load, controller load, and the like”), [0059] (describing “asynchronous ... operation mode in which a modification to one member of a copy set can be propagated to other members of the copy set after a time delay”). Likewise, Ofek teaches balancing similar operational requirements, and adjusting certain parameters like “*send batch size*” based on “*available communication bandwidth*” to achieve desired system performance. Ex-1003, ¶ 208; Ex-1007, 7:10-21 (explaining the data storage system can operate in a “point-in-time or asynchronous mode” to eliminate “any performance penalty if the

communication link between the primary and secondary data storage systems is too slow), 12:22-24 (explaining “operational modes are selectable at the logical volume level based on the performance, distance, and speed of recovery requirements”), 17:39-42, 18:9-13 (explaining the adaptive copy—write pending mode for specified volumes “is convenient in situations where the write activity is caused by heavy batch loads or data reorganization can severely impact performance due to the data storage systems maintaining a full synchronous state”).

A POSITA would have recognized that quantity of data that could be reliably transmitted from the primary storage facility to the secondary storage facility would likely depend on the available bandwidth of the communication link between them, particularly given the teachings of Lubbers, Sicola, and Ofek discussed above for claim 11 and in §VII.B. Ex-1003, ¶ 209. Thus, a POSITA would have reasonably expected to succeed in adaptively sizing send batches in this manner because Lubbers and Ofek already teach adjusting system operation based on balancing these competing objectives. Ex-1003, ¶ 210, Ex-1007, 17:39-41.

Accordingly, the combined teachings of Lubbers, Sicola, and Ofek would have rendered obvious claim 11. Ex-1003, ¶¶ 205-211.

4. **Claim 12:** *The data redundancy system according to claim 1, wherein a first write transaction in a send batch at the primary storage facility is replaced with a second write transaction where the second write transaction overwrites data written by the first write transaction.*

Lubbers, Sicola, and Ofek would have rendered obvious claim 12. A POSITA would have been motivated to combine Lubbers, Sicola, and Ofek as discussed above. §VII.B; Ex-1003, ¶ 212.

It would have been obvious to combine Ofek's teachings to “*overwrite*” a “*first write transaction in a send batch*” with “*a second write transaction*” with the teachings of Lubbers and Sicola. Ex-1003, ¶ 213. Lubbers contemplates a circumstance where an initial write transaction needs to be replaced with a second write transaction. Ex-1003, ¶ 214; Ex-1005, [0072]-[0073] (explaining that when “the controller does not have the most current data, then the controller goes offline to the host to receive a full copy from the controller that assumes the role of the source controller”). Sicola provides further details on overwriting write transactions in a batch in a multi-site storage system. Ex-1003, ¶ 215; Ex-1006, 13:6-15 (explaining that asynchronous system operation involves a micro-logging and micro-merge process where “‘outstanding’ writes were logged to the initiator controller A1’s write-back cache, which is also mirrored in partner controller A2’s write-back cache,” and that during failover, “the partner controller (A2) re-issues these remote copies from the micro-log.”).

Ofek teaches a channel adapter writes one or more records into a cache that can be “*overwritten*” with new writes that in operation, “the channel adapter writes one or more records into cache.” Ex-1003, ¶ 216; Ex-1007, 37:64-67, 39:22-24, 39:41-43 (explaining that if a “track does reside in cache,” then the “channel adapter is working in an ‘overwrite cache option’ for the volume,” where “a new version will overwrite this write pending record in the cache”), 39:52-56.

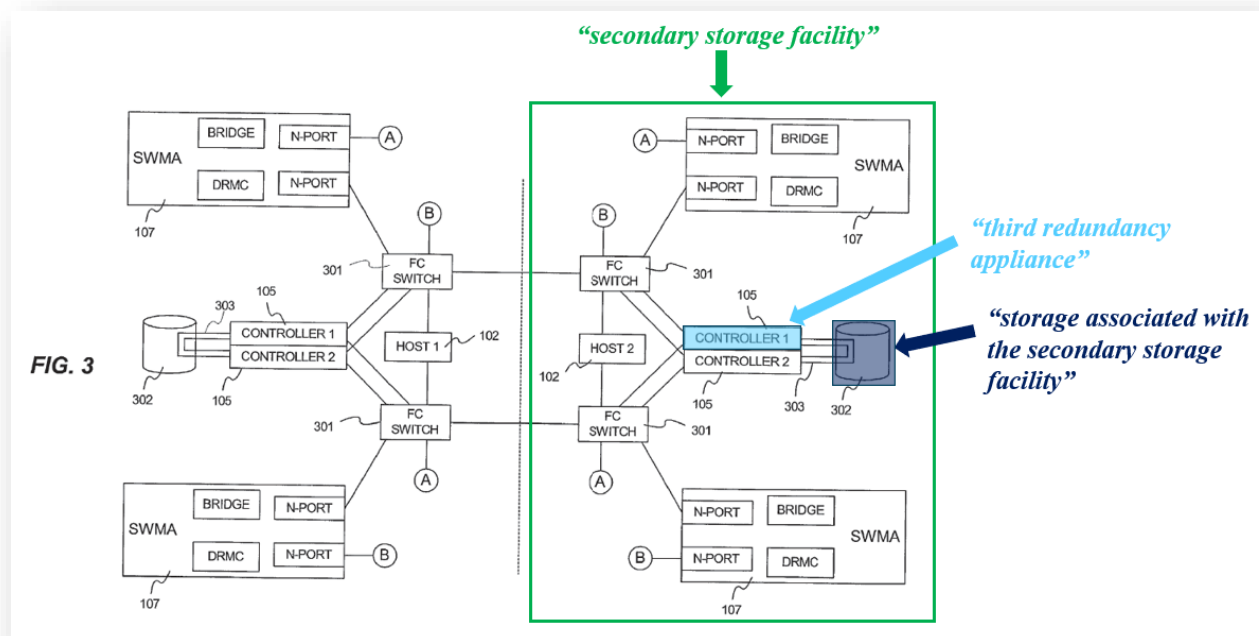
A POSITA would have been motivated to combine the teachings of Lubbers, Sicola, and Ofek to replace a first write transaction in a bundle of write data (“*send batch*”) at a primary site (“*primary storage facility*”) by overwriting it with a second write transaction in this manner to balance latency, high availability, disaster tolerance, data transmission path performance, and system performance, speed, and cost. Ex-1003, ¶ 217. A POSITA would have reasonably expected to succeed in adaptively sizing send batches in this manner because Lubbers and Ofek teach overwriting transactions stored in caches. Ex-1003, ¶ 218, Ex-1007, 17:39-41.

Accordingly, the combined teachings of Lubbers, Sicola, and Ofek would have rendered obvious claim 12. Ex-1003, ¶¶ 212-219.

5. **Claim 13:** *The data redundancy system according to claim 1, wherein the third redundancy appliance stores redundant data to the storage associated with the secondary facility according to receive batches.*

Lubbers, Sicola, and Ofek would have rendered obvious claim 13. A POSITA would have been motivated to combine Lubbers, Sicola, and Ofek as discussed above. §VII.B; Ex-1003, ¶ 220.

Lubbers, Sicola, and Ofek teach “*the third redundancy appliance stores redundant data to the storage associated with the secondary facility*” as discussed above in element [1e], and shown below:



Ex-1005, FIG. 3 (annotated); Ex-1003, ¶ 221.

Lubbers, Sicola, and Ofek also teach “*receive batches*,” and storing the write transactions “*according to receive batches*” as discussed in claim 9 that Lubbers,

Sicola, and Ofek teaches “*send batches*.” Ex-1003, ¶ 222; Ex-1007, 35:13-16, 37:33-37:35.

A POSITA would have been motivated to combine the teachings Ofek to receive and decode “*receive batches*” with the teachings of Lubbers and Sicola in this manner to balance the effects of latency, high availability, disaster tolerance, data transmission path performance, and system performance, speed, and cost. Ex-1003, ¶ 223.

A POSITA would have reasonably expected to successfully combine the teachings of Lubbers, Sicola, and Ofek such that the third controller (“*third redundancy appliance*”), at the alternate site (“*secondary storage facility*”), writes redundant data (“*stores redundant data*”) to the physical storage device 302 (“*the storage associated with the secondary facility*”) after decoding bundles of write commands in a channel command word chain (“*receive batches*”) to efficiently use resources during write processes. Ex-1003, ¶ 224.

Accordingly, the combined teachings of Lubbers, Sicola, and Ofek would have rendered obvious claim 13. Ex-1003, ¶¶ 220-225.

6. Claim 14: *The data redundancy system according to claim 13, wherein the third redundancy appliance stores a receive batch after the receive batch is determined safe to apply.*

Lubbers, Sicola, and Ofek would have rendered obvious claim 14. A POSITA would have been motivated to combine Lubbers, Sicola, and Ofek as discussed above. §VII.B; Ex-1003, ¶ 226.

It would have been obvious to combine Ofek's teachings to store “*a receive batch after the receive batch is determined safe to apply*” and the teachings of Lubbers and Sicola because of the shared goal of maintaining data integrity and availability. Ex-1003, ¶ 227; Ex-1005, [0032] (explaining “[s]torage controllers 105 manage the ... functions that maintain integrity and availability of the data stored therein”).

Ofek also teaches the importance of ensuring data integrity before storing write requests, shown below:

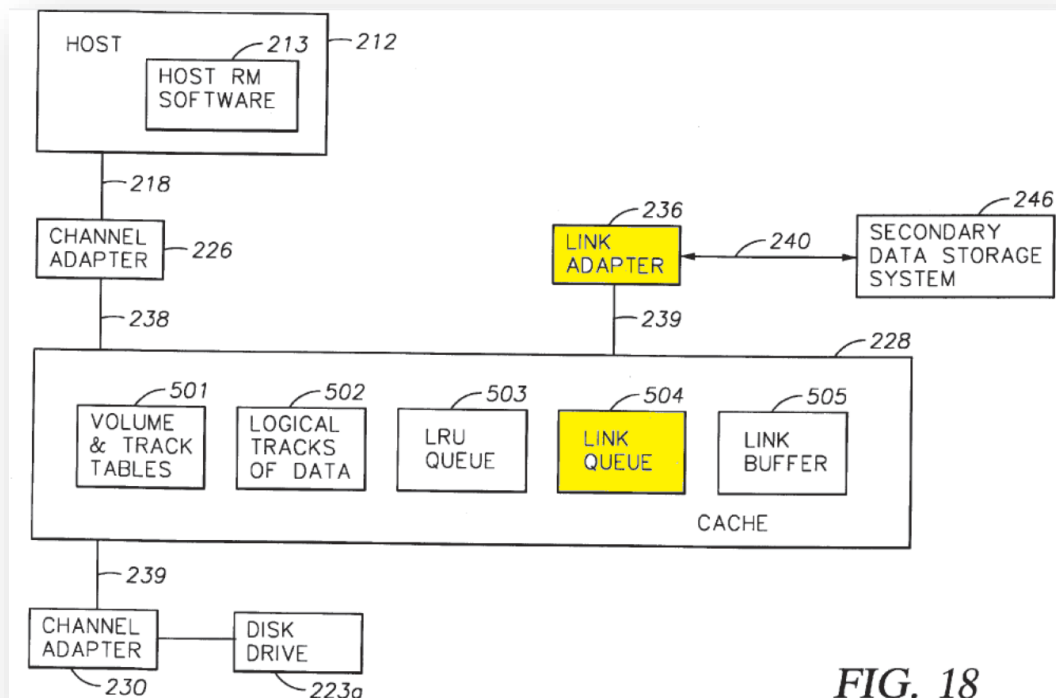


FIG. 18

Ex-1003, ¶ 228; Ex-1007, FIG. 18 (annotated), 35:18-20, 35:60-36:2 (describing “the entry in the link queue 504 includes a one-byte password for confirming the validity of the entry”), 3:10-18 (explaining the data storage system “achieves nearly 100 percent data integrity by assuring that all data is copied to a geographically remote site.”).

A POSITA would have been motivated to store “a receive batch after the receive batch is determined safe to apply” using the combined teachings of Lubbers, Sicola, and Ofek in this manner to improve data availability and reliability, which is a shared goal. Ex-1003, ¶ 229. A POSITA would have reasonably expected to successfully determine whether write commands are safe to apply to a standby

system in this manner as merely combining the system software teachings of Lubbers and Sicola with the teachings of Ofek to achieve a mutual goal. Ex-1003, ¶ 230, Ex-1007, 17:39-41.

Accordingly, the combined teachings of Lubbers, Sicola, and Ofek would have rendered obvious claim 14. Ex-1003, ¶¶ 226-231.

7. Claim 15: *The data redundancy system according to claim 14, wherein a receive batch is determined safe to apply when a data integrity test is successfully performed.*

Lubbers, Sicola, and Ofek would have rendered obvious claim 15 as discussed above in claim 14. As stated in claim 14, Ofek teaches a one-byte password for confirming the validity of the entry to verify data integrity, and how the system achieves nearly 100% data integrity by ensuring no errant data is stored in the backup system. Ex-1007, 3:10-18, 35:60-36:2. Further, a POSITA would have been motivated to combine Lubbers, Sicola, and Ofek as discussed above. §VII.B; Ex-1003, ¶ 232.

Accordingly, the combined teachings of Lubbers, Sicola, and Ofek would have rendered obvious claim 15. Ex-1003, ¶¶ 232-235.

8. **Claim 16:** *The data redundancy system according to claim 13, wherein a first write transaction in a first receive batch at the secondary storage facility is replaced with a second write transaction in a second receive batch wherein the second write transaction overwrites data written by the first write transaction and the first and second receive batches are combined.*

Lubbers, Sicola, and Ofek would have rendered obvious claim 16. A POSITA would have been motivated to combine Lubbers, Sicola, and Ofek as discussed above. §VII.B; Ex-1003, ¶ 236.

Lubbers, Sicola, and Ofek would have rendered obvious “*receive batches*” as discussed in claim 13, and a “*second write transaction overwrites data written by a first write transaction*” as discussed in claim 12. Ex-1003, ¶ 237.

It would have been obvious to combine the teachings of Ofek to “*a first write transaction in a first receive batch*” and “*a second write transaction in a second receive batch*” when “*the first and second receive batches are combined*” and the teachings of Lubbers and Sicola. Ex-1007, 36:13-16 (explaining benefit of bundling write data before transmission and decoding), 35:13-24 (describing how each write and read command “entry is placed in a link queue 504 by a channel adapter” and “transmitted over the link 240”). Ofek teaches a first write bundle (“*first receive batch*”) and a second write bundle (“*second receive batch*”) that are sequentially sent to the secondary storage volume (R2) (“*secondary storage facility*”). Additionally, Ofek teaches write bundles can be combined, and transactions can be overwritten while in cache. Ex-1007, 40:21-28 (explaining that “keeping multiple versions of

tracks in cache will tie up cache resources, so ... no more than a certain number of such old versions should be permitted in cache at any given time”) which also applies to the bundled writes received at the secondary storage volume. Ex-1007, 40:21-28. That teaching could be combined with the teachings of Lubbers and Sicola and applied to the primary site or the alternate site (*“secondary storage facility”*). Ex-1003, ¶¶ 238-240.

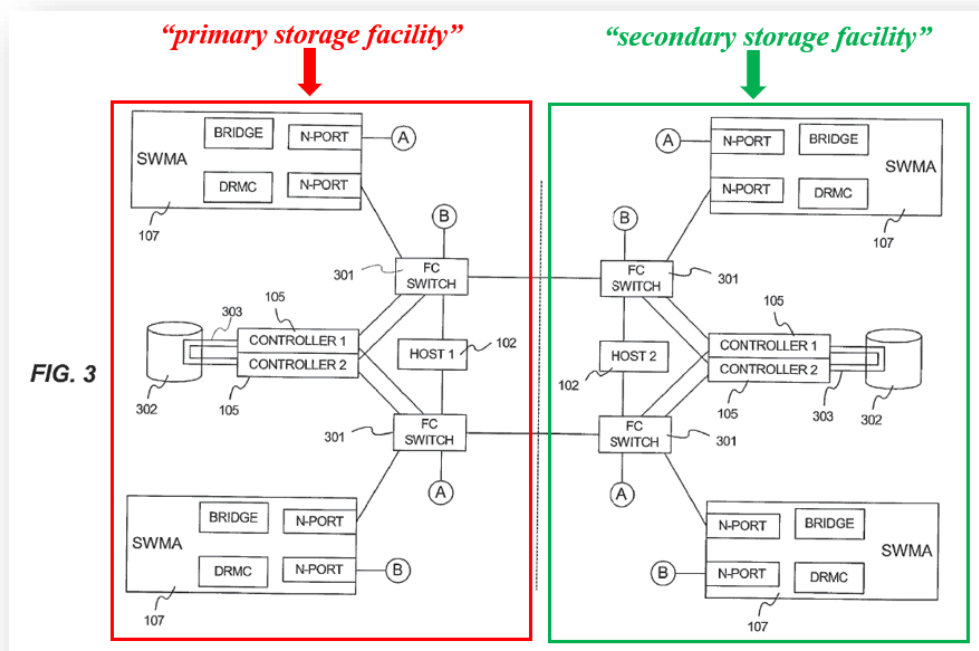
A POSITA would have been motivated to combine the teachings of Ofek to *“a first write transaction in a first receive batch”* and *“a second write transaction in a second receive batch”* where *“the first and second receive batches are combined”* with the teachings of Lubbers and Sicola in this manner to free up cache resources and improve system performance, which is a shared goal. Ex-1003, ¶ 241. A POSITA would have reasonably expected to succeed in *“overwriting”* transactions in this manner by adjusting system software taught by Lubbers and Sicola using teachings of Ofek to achieve a mutual goal. Ex-1003, ¶ 242.

Accordingly, the combined teachings of Lubbers, Sicola, and Ofek would have rendered obvious claim 16. Ex-1003, ¶¶ 236-243.

9. Claim 17

- a. *[17preamble]: “A data redundancy method for storing a primary copy of data at a primary storage facility and for storing data that is redundant of the primary copy at a secondary storage facility, the method comprising”*

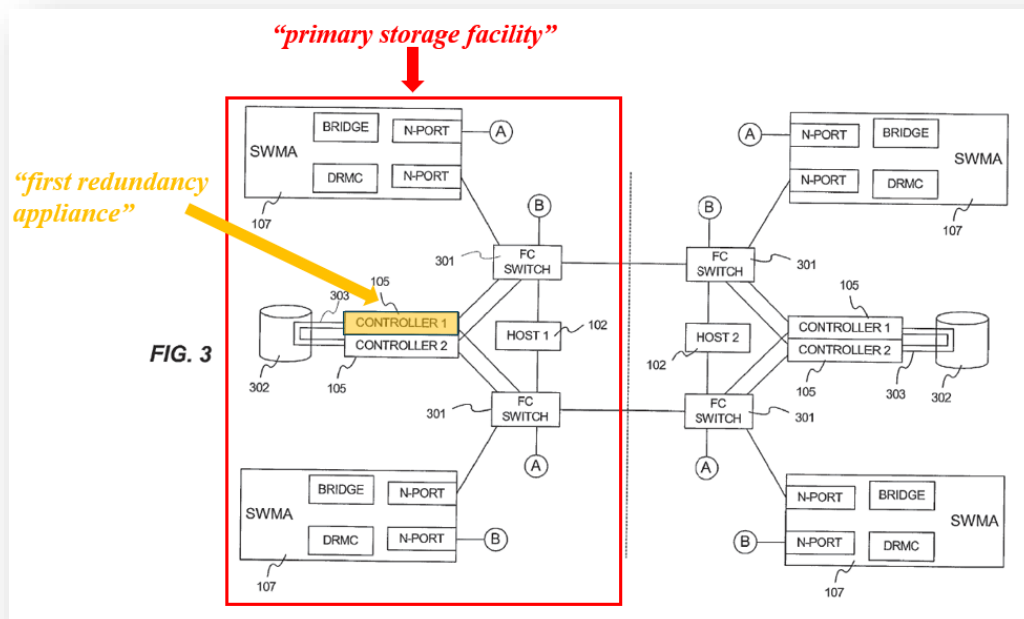
As discussed for elements [1a] and [1d], Lubbers and Sicola would have rendered obvious a multi-site storage area network that includes multiple sites where a primary site serves as the “*primary storage facility*” “for storing a primary copy of data” and an alternative site, the “*secondary storage facility*,” where the redundant data is stored:



Ex-1005, FIG. 3 (annotated); Ex-1003, ¶¶ 244-246.

b. *[17a]: “receiving a sequence of write requests at a first redundancy appliance at the primary storage facility;”*

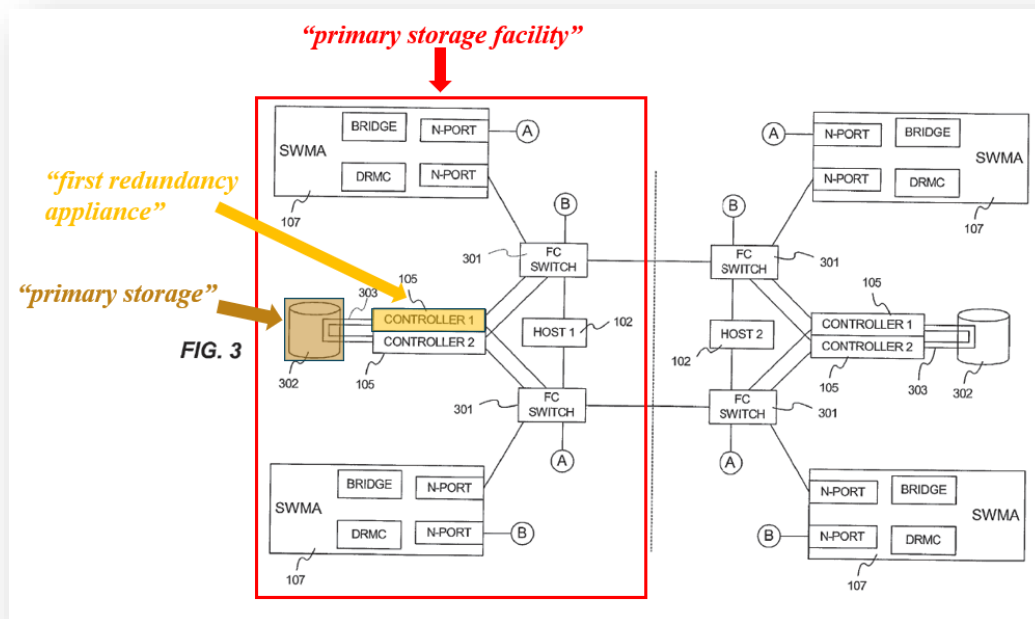
As discussed for element [1b], Lubbers and Sicola would have rendered obvious the first controller (*“first redundancy appliance”*), at the primary site (*“primary storage facility”*), handling data traffic (*“sequence of write requests”*) by packing as many host writes as possible into a single buffer is *“receiving a sequence of write requests at a first redundancy appliance”* *“receiving a sequence of write requests.”* Further, as discussed for element [1a], Lubbers and Sicola would have rendered obvious storage at a primary site (*“primary storage facility”*).



Ex-1005, FIG. 3 (annotated); Ex-1003, ¶¶ 247-248.

c. [17b]: “storing data for the sequence of write requests in primary storage at the primary storage facility;”

As discussed for element [1b], Lubbers and Sicola would have rendered obvious the first controller logs data, implements and manages logical units, and stores the same (“storing data”) in physical storage devices 302 (“primary storage”) coupled to the first controller using the FCAL 303 at the primary site (“primary storage facility”).



Ex-1005, FIG. 3 (annotated); Ex-1003, ¶¶ 249-250.

d. [17c]: “logging at least a sequence of write requests for transfer to the secondary storage facility;”

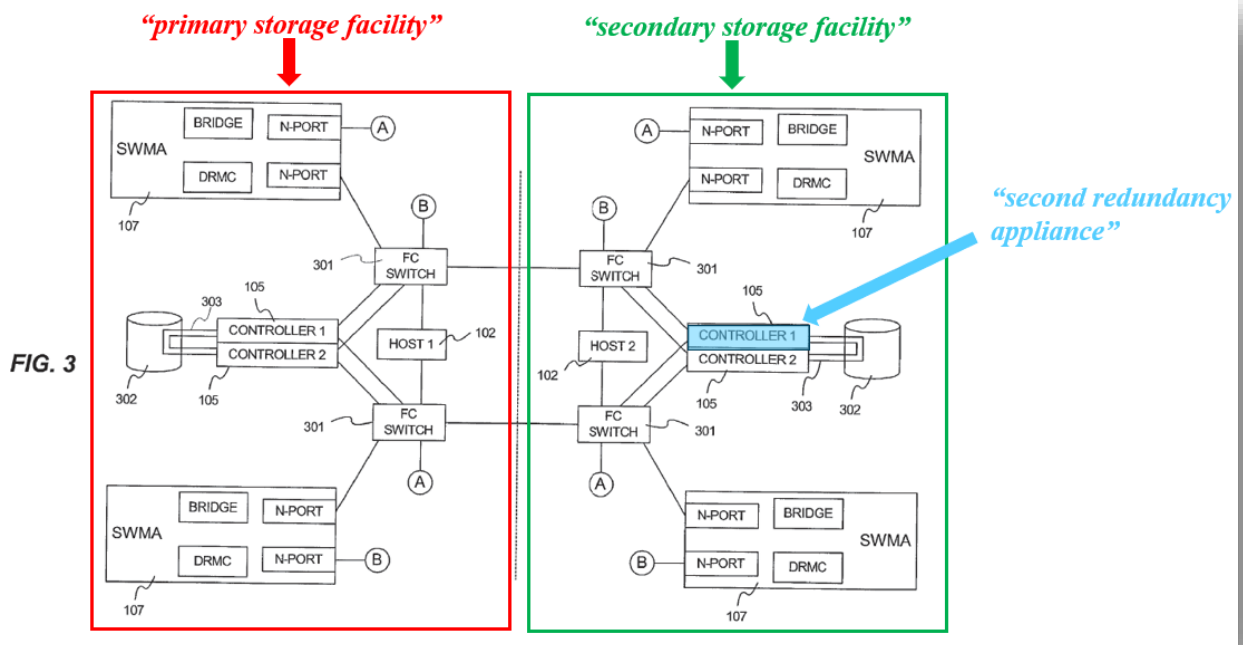
As discussed for elements [1b] and [1c], and claim 12, Lubbers, Sicola, and Ofek would have rendered obvious this element. Ex-1003, ¶ 251.

- e. [17d]: “forwarding the logged write requests as send batches to a second redundancy appliance at the secondary storage facility disjoint from the primary storage facility, the second storage facility batching the received write request as receive batches;”

As discussed for element [1d], Lubbers and Sicola would have rendered obvious “*the secondary storage facility disjoint from the primary storage facility.*”

Ex-1003, ¶ 252.

As discussed for claim 9, Lubbers, Sicola, and Ofek would have rendered obvious “forwarding write requests as send batches to *a second redundancy appliance at the secondary storage facility . . . the second storage facility batching the received write request as receive batches:*”

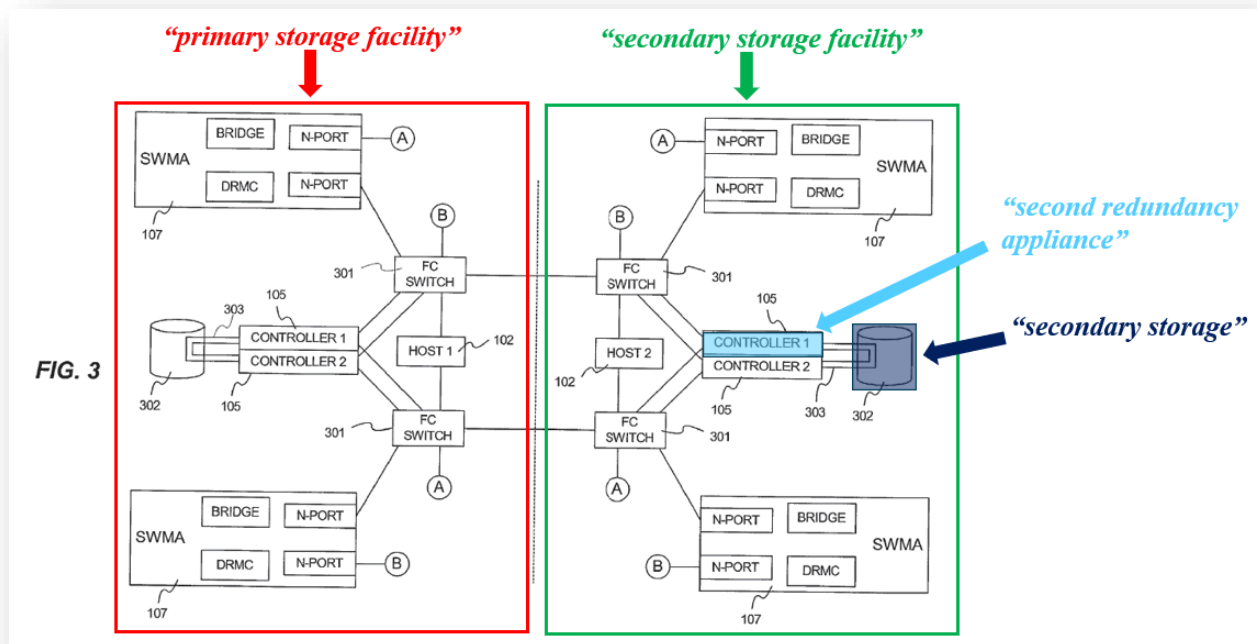


Ex-1005, FIG. 3 (annotated); Ex-1003, ¶ 253.

As discussed for element [17c], Lubbers and Sicola would have rendered obvious logged sequences of write requests, rephrased in this claim as “*logged write requests.*” Ex-1003, ¶¶ 254-255.

f. [17e]: “*storing data associated with the receive batches in secondary storage at the secondary storage facility*”

As discussed for claim 13, Lubbers, Sicola, and Ofek would have rendered obvious this element:



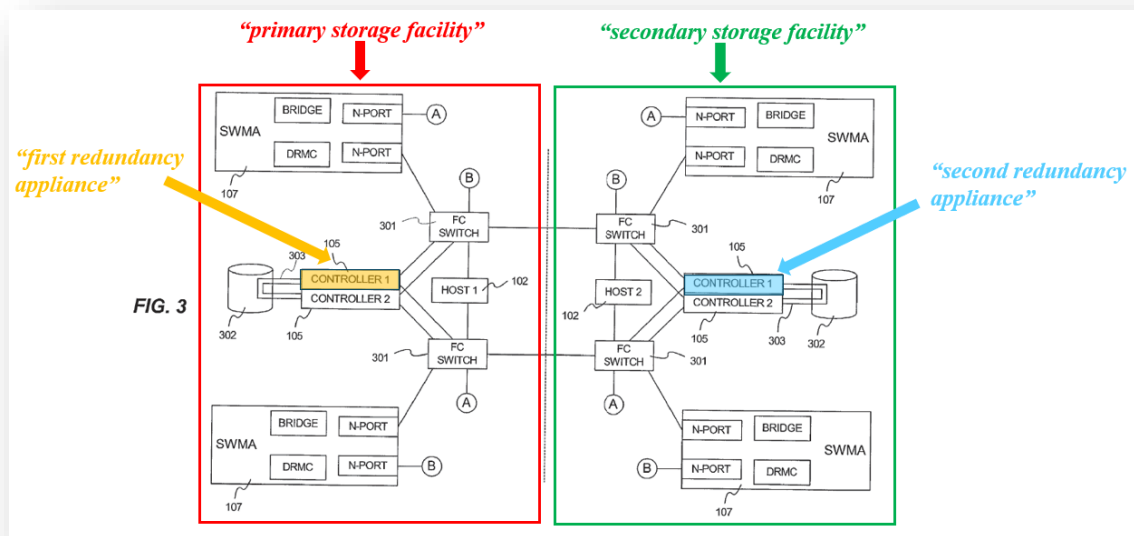
Ex-1005, FIG. 3 (annotated); Ex-1003, ¶ 256.

- g. [17f]: “replacing a first write transaction in a receive batch at the second storage facility with a second write transaction wherein the second write transaction overwrites data written by the first write transaction; and”

As discussed for claim 14, Lubbers, Sicola, and Ofek would have rendered obvious this element. Ex-1003, ¶ 257

- a. [17g]: “the second redundancy appliance acting as the first redundancy appliance in response to a failover event.”

As discussed for claims [1c], [1f] and 2, Lubbers and Sicola would have rendered obvious this element:



Ex-1005, FIG. 3 (annotated); Ex-1003, ¶ 258.

10. Claim 18

a. *[18preamble]: “A data redundancy system, comprising:”*

As discussed for element [1preamble], Lubbers taught or suggested “a data redundancy system.” Ex-1003, ¶ 260.

b. *[18a]: “a primary storage facility including at least one primary redundancy appliance for receiving a sequence of write requests and for storing data for the sequence of write requests in storage associated with the primary storage facility, the primary storage facility batching the write requests as send batches for transfer to a secondary storage facility; and”*

As discussed for elements [1a], [1b], 1[c], Lubbers and Sicola would have rendered obvious this element, except “*batching the write requests as send batches.*” Ex-1003, ¶ 261.

As discussed for claim 9, Lubbers, Sicola, and Ofek would have rendered obvious “*the primary storage facility batching the write requests as send batches for transfer to a secondary storage facility.*” Ex-1003, ¶ 262.

c. *[18b]: “the secondary storage facility disjoint from the primary storage facility including at least one secondary redundancy appliance for receiving the sequence of write requests and for storing data for the sequence of write requests in storage associated with the secondary storage facility, the second storage facility batching the received write request as receive batches, the second redundancy appliance acting as the primary redundancy appliance in response to a failover event;”*

As discussed for elements [1a]-[1f], Lubbers and Sicola would have rendered obvious this element, except “*batching received write requests.*” As discussed for

claim 13, Lubbers, Sicola, and Ofek would have rendered obvious “*the second storage facility batching the received write request as receive batches.*” Ex-1003, ¶ 263.

- d. ***[18c]: “wherein a first write transaction in a receive batch at the second storage facility is replaced with a second write transaction where the second write transaction over writes data written by the first write transaction.”***

As discussed for claim 14, Lubbers, Sicola, and Ofek would have rendered obvious this element. Ex-1003, ¶ 264.

VIII. DISCRETIONARY DENIAL IS NOT APPROPRIATE

A. Denial under § 325(d) is not warranted.

Denial under § 325(d) is not warranted because the challenges presented in this Petition are neither cumulative nor redundant to the prosecution of the ’182 Patent, as Lubbers, Sicola, and Ofek were neither cited nor considered by the Examiner during prosecution of the ’182 Patent. The references in this Petition were neither relied upon during prosecution nor are cumulative of art relied upon during prosecution. Further, the ’182 Patent has not been previously challenged in an IPR. In addition, the Examiner did not have the benefit of Dr. Almeroth’s detailed expert testimony and the other evidence of record.

Thus, the Board should not exercise discretion to deny institution.

B. The Board Should Not Deny Institution Under 35 U.S.C. § 314(a)

The merits of Petitioner’s arguments are compelling and the evidence in support is substantial. That “alone demonstrates that the PTAB should not discretionarily deny institution under *Fintiv*.” Ex-1011 at 4-5. Regardless, the six *Fintiv* factors do not justify denying institution.

Factor 1 (stay) is **neutral**. SAP has not yet moved to stay pending this IPR.

Factor 2 (proximity to trial) is **neutral or weighs against denial**. In the originally filed litigation by Patent Owner Valtrus (*Valtrus I*), the trial date is not currently set (though jury selection is set for September 22, 2025). Ex-1012. However, the Board should not rely on the current schedule in *Valtrus I* because Patent Owner Valtrus filed suit alone (without the other named Patent Owner, Key Patent Innovations) and thus did not have Article III Standing. Ex-1013; Ex-1014. Petitioner filed a Motion to Dismiss for Lack of Standing in *Valtrus I*, reconsideration of which remains pending and casts serious doubt on whether *Valtrus I* will actually proceed along the current schedule. Indeed, in response to Petitioner’s Motion to Dismiss for Lack of Standing, Patent Owners essentially concede that Valtrus alone lacked standing because Patent Owners together filed a second, duplicative lawsuit (*Valtrus II*) in an effort to rectify the Article III standing deficiencies in *Valtrus I*. Ex-1010; Ex-1015. This conduct confirms that Patent Owners themselves doubt that the trial schedule in *Valtrus I* will hold, further

supporting that the Board should not rely on that schedule at all. Additionally, the other pending litigations involving the challenged patent (*Valtrus II* and a declaratory judgment action filed by Petitioner) do not have any trial date currently set, meaning there is no reliable trial date for the Board to consider in relation to this Petition.

Even if the Board assumes the *Valtrus I* trial date applies, this factor is neutral because there will be other claims not litigated at trial that are covered by this Petition. Therefore, a final written decision will issue first at least on the claims not litigated at trial but covered by this Petition.

Factor 3 (investment in parallel proceedings) **weighs against denial**. Importantly, claims 2, 4, 6, and 13 are not at issue in the related litigations. *Samsung Bioepis Co. Ltd v. Regeneron Pharmaceuticals, Inc.*, IPR2023-00739, Paper 9 at 57 (Oct. 20, 2023) (“[B]ecause certain of the challenged claims, . . . are not expressly at issue in the district court litigation, *Fintiv* factor 3 favors institution.”) The related litigations are in its early stages of discovery, motions to transfer and dismiss have been filed, and the investment by the parties has been relatively minimal. All litigations involving the challenged patent are at an early stage, with major deadlines several months away. Ex-1012. Patent Owners may argue that the parties have invested substantial resources on procedural motions, but the resources expended have been directed toward resolving procedural missteps perpetuated by Patent

Owner Valtrus's choice to file alone (without Article III standing) and Patent Owners' later attempts to fix that with procedurally improper duplicative litigation. These procedural disputes (caused by Patent Owners' actions) should not be held against Petitioner under this factor.

Factor 4 (overlap) **weighs against denial** because Petitioner hereby stipulates that, if the IPR is instituted, it will not pursue the IPR grounds in the district court. Indeed, neither Lubbers, Sicola, nor Ofek are at issue in the District Court Litigation. Ex-1016. Thus, "[i]nstituting trial here serves overall system efficiency and integrity goals by not duplicating efforts and by resolving materially different patentability issues." *Apple, Inc. v. SEVEN Networks, LLC*, IPR2020-00156, Paper 10 at 19 (June 15, 2020); *see also Sand Revolution II, LLC v. Continental Intermodal Group-Trucking LLC*, IPR2019-01823, Paper 24 at 12 (June 16, 2020). In addition, Petitioner challenges more claims than Patent Owners will ultimately assert at trial. Patent Owner Valtrus agreed to narrow to 30 total asserted claims by April 30, 2025, and a maximum of six asserted claims per asserted patent. Ex-1012. Thus, Patent Owner will be limited to asserting up to six of the 14 asserted claims of the '182 Patent, merely a subset of the claims challenged in this Petition. *See, e.g., Apple*, No. IPR2020-00156, Paper 10, at 17, 21-22.

While Factor 5 (parties) may weigh slightly in favor of denial because the parties are the same as in district court, under the "holistic view" of the remaining

factors and whether integrity of the system and efficiency is best served, institution is favored. *Samsung Elecs. Co. Ltd. v. Dynamics Inc.*, IPR2020-00505, Paper 11 at 15 (Aug. 12, 2020).

Factor 6 (other circumstances) **weigh against denial**. Here, the merits of the Petition are particularly compelling—for example, Lubbers strikingly similar disclosure to that of the '182 Patent, is directed at the same problem and proposes the same solutions as the '182 Patent and discloses nearly identical architecture, including several nearly identical figures. The evidence presented, if unrebutted in trial, would plainly lead to a conclusion that one or more claims are unpatentable by a preponderance of the evidence.

IX. CONCLUSION

Petitioner respectfully requests that the Board institute IPR and enter a final written decision finding the Challenged Claims unpatentable.

Dated: January 14, 2025

Respectfully submitted,

/Louis L. Campbell/

Louis L. Campbell (Lead Counsel)

llcampbell@winston.com

USPTO Reg. No. 59,963

Winston & Strawn LLP

255 Shoreline Dr., Ste. 520

Redwood City, CA 94065

T: (650) 858-6500, F: (650) 858-6550

Lead Counsel for Petitioner

CERTIFICATE OF COMPLIANCE

This Petition complies with the word count limits set forth in 37 C.F.R. § 42.24(a)(1)(i), because this Petition contains 13,976 words, excluding the parts of the Petition exempted by 37 C.F.R. § 42.24(a)(1), determined using the word count provided by Microsoft Word, which was used to prepare this Petition.

Dated: January 14, 2025

Respectfully submitted,

/Louis L. Campbell/

Louis L. Campbell (Lead Counsel)

llcampbell@winston.com

USPTO Reg. No. 59,963

Winston & Strawn LLP

255 Shoreline Dr., Ste. 520

Redwood City, CA 94065

T: (650) 858-6500, F: (650) 858-6550

Lead Counsel for Petitioner

CERTIFICATE OF SERVICE

Under 37 C.F.R. §§ 42.6(e) and 42.105(a), this is to certify that on January 14, 2025, I caused to be served a true and correct copy of the foregoing “**PETITION *INTER PARTES* REVIEW OF U.S. PATENT NO. 7,152,182**” and accompanying exhibits by Federal Express priority overnight mail on the following:

Botos Churchill IP Law LLP
430 Mountain Avenue
Suite 401
New Providence, NJ 07974

With a courtesy copy of this Petition and supporting material was also served on litigation counsel for Patent Owners via email:

Matt Berkowitz
mberkowitz@reichmanjorgensen.com
Patrick R. Colsher
pcolsher@reichmanjorgensen.com
Aaron Morris
amorris@reichmanjorgensen.com
Navid Cyrus Bayar
nbayar@reichmanjorgensen.com
Reichman Jorgensen Lehman & Feldberg LLP
100 Marine Parkway
Suite 300
Redwood Shores, CA 94065

Khue Van Hoang
khoang@reichmanjorgensen.com
Reichman Jorgensen Lehman & Feldberg LLP
400 Madison Avenue
Suite 14d
New York, NY 1001

Philip Ekle

peklem@reichmanjorgensen.com
Savannah Harman Carnes
scarnes@reichmanjorgensen.com
Reichman Jorgensen Lehman & Feldberg LLP
1909 K Street, NW
Suite 800
Washington, DC 20006

/Louis L. Campbell/
Louis L. Campbell (Lead Counsel)
llcampbell@winston.com
USPTO Reg. No. 59,963
Winston & Strawn LLP
255 Shoreline Dr., Ste. 520
Redwood City, CA 94065
T: (650) 858-6500, F: (650) 858-6550

Lead Counsel for Petitioner