

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE LLC,
Petitioner,

v.

SECURITY FIRST INNOVATIONS, LLC,
Patent Owner.

Case IPR2024-00215
Patent 10,452,854

PATENT OWNER'S RESPONSE

TABLE OF CONTENTS

	Page
I. INTRODUCTION	1
II. AS THE BOARD HAS TWICE FOUND, THE ORSINI GROUND FAILS (ALL CLAIMS, GROUND 1).	1
A. The '854 Patent Claims Fundamentally Different Technology Than That Disclosed In Orsini.	1
B. The Board Correctly Found That Petitioner Fails To Show That Orsini Anticipates The Claims.	6
1. Orsini Does Not Inherently Disclose Claim 1.	11
2. Petitioner's Other Arguments Cannot Establish Anticipation Of Claim 1.....	15
C. Petitioner Fails To Show That Orsini Renders The Claims Obvious.	18
1. Petitioner Cannot Rely On Common Sense To Supply Missing Limitation 1[c].	19
2. Petitioner Fails To Show A POSITA Would Have Been Motivated To Modify Orsini to Add Limitation 1[c].	21
III. THE FOSTER-HAYHURST GROUND FAILS (ALL CLAIMS, GROUND 2).	32
A. Petitioner Fails To Demonstrate That Hayhurst Is Analogous Art.	32
1. Hayhurst And The '854 Patent Do Not Share The Same Field Of Endeavor.	33
2. Hayhurst Is Not Reasonably Pertinent To The Problem The Inventors Of The '854 Patent Were Addressing.....	38

B.	Petitioner Fails To Demonstrate That A POSITA Would Be Motivated To Combine Foster And Hayhurst (All Claims, Ground 2).....	41
1.	Foster Provides Synchronized Transmission To Avoid Delays.....	42
2.	Hayhurst’s System Does Not Allow For Real-Time Streaming Because Its System Requires Delays.	47
3.	Petitioner’s Combination Involves Storing And Distributing All Of Foster’s Content Units Per Hayhurst’s Methodology Which Precludes Real-Time Streaming.....	49
4.	A POSITA Would Not Be Motivated To Combine Foster And Hayhurst As Proposed.....	51
a.	Adding Hayhurst’s Delivery Method To Foster’s System Exacerbates The Underlying Problem Of Delay Foster Seeks To Solve.	52
b.	Hayhurst’s Delivery Method Is Incompatible With Foster’s System.....	55
c.	Hayhurst’s Methods Make Foster’s System Less Secure.	58
d.	Petitioner Fails To Prove That The Alleged Benefits Outweigh The Drawbacks Of The Proposed Combination.	61
C.	Petitioner Fails To Demonstrate That Foster-Hayhurst Teaches “Generating A Plurality Of Data Chunks Based On The Data Set” By “Distributing The Data Set Into A Plurality Of Shares” (All Claims, Ground 2).	63
IV.	CONCLUSION	71

TABLE OF AUTHORITIES

Page(s)

COURT DECISIONS

<i>Airbus S.A.S. v. Firepass Corp.</i> , 941 F.3d 1374 (Fed. Cir. 2019)	34
<i>Arctic Cat Inc. v. Bombardier Rec. Prods.</i> , 876 F.3d 1350 (Fed. Cir. 2017)	25
<i>Arctic Cat Inc. v. Polaris Indus.</i> , 795 Fed. App'x. 827 (Fed. Cir. 2019)	25, 51
<i>Arendi S.A.R.L. v. Apple, Inc.</i> , 832 F.3d 1355 (Fed. Cir. 2016)	20
<i>Ariosa Diagnostics v. Verinata Health, Inc.</i> , 805 F.3d 1359 (Fed. Cir. 2015)	29
<i>BlephEx, LLC v. Myco Indus.</i> , 24 F.4th 1391 (Fed. Cir. 2022)	16
<i>D Three Enters., LLC v. SunModo Corp.</i> , 890 F.3d 1042 (Fed. Cir. 2018)	12
<i>DSS Tech. Mgm't v. Apple Inc.</i> , 885 F.3d 1367 (Fed. Cir. 2018)	20
<i>Enplas Display Device Corp. v. Seoul Semiconductor Co., Ltd.</i> , 909 F.3d 398 (Fed. Cir. 2018)	11
<i>Finisar Corp. v. DirecTV Group, Inc.</i> , 523 F.3d 1323 (Fed. Cir. 2008)	16
<i>Forum US, Inc. v. Flow Valve, Ltd. Liab. Co.</i> , 926 F.3d 1346 (Fed. Cir. 2019)	11

<i>Henny Penny Corp. v. Frymaster LLC</i> , 938 F.3d 1324 (Fed. Cir. 2019)	25
<i>In re Benno</i> , 768 F.2d 1340 (Fed. Cir. 1985)	11
<i>In re Bigio</i> , 381 F.3d 1320 (Fed. Cir. 2004)	34
<i>In re Clay</i> , 966 F.2d 656 (Fed. Cir. 1992)	33, 37, 38, 41
<i>In re Gordon</i> , 733 F.2d 900 (Fed. Cir. 1984)	51
<i>In re ICON Health & Fitness, Inc.</i> , 496 F.3d 1374 (Fed. Cir. 2007)	51
<i>In re Kahn</i> , 441 F.3d 977 (Fed. Cir. 2006)	66
<i>In re Klein</i> , 647 F.3d 1343 (Fed. Cir. 2011)	32, 33
<i>In re Magnum Oil Tools Int’l, Ltd.</i> , 829 F.3d 1364 (Fed. Cir. 2016)	66
<i>In re NTP, Inc.</i> , 654 F.3d 1279 (Fed. Cir. 2011)	23
<i>In re Ratti</i> , 270 F.2d 810 (CCPA 1959)	51
<i>In re Wood</i> , 599 F.2d 1032 (CCPA 1979)	34
<i>Intel Corp. v. PACT XPP Schweiz AG</i> , 61 F.4th 1373 (Fed. Cir. 2023)	29
<i>InTouch Techs., Inc. v. VGo Commc’ns., Inc.</i> , 751 F.3d 1327 (Fed. Cir. 2014)	31

<i>Kamstrup A/S v. Axioma Metering UAB</i> , 43 F.4th 1374 (Fed. Cir. 2022)	34
<i>KSR Int’l Co. v. Teleflex Inc.</i> , 550 U.S. 398 (2007).....	30, 66
<i>LBT IP I LLC v. Apple Inc.</i> , 2023 U.S. App. LEXIS 14376 (Fed. Cir. 2023)	13
<i>Mantech Envtl. Corp. v. Hudson Envtl. Servs.</i> , 152 F.3d 1368 (Fed. Cir. 1998)	5
<i>Microsoft Corp. v. Biscotti, Inc.</i> , 878 F.3d 1052 (Fed. Cir. 2017)	12
<i>Net MoneyIN, Inc. v. VeriSign, Inc.</i> , 545 F.3d 1359 (Fed. Cir. 2008)	16
<i>Ruiz v. A.B. Chance Co.</i> , 357 F.3d 1270 (Fed. Cir. 2004)	31
<i>Sanofi-Aventis Deutschland GmbH v. Mylan Pharms. Inc.</i> , 66 F.4th 1373 (Fed. Cir. 2023)	33
<i>Schumer v. Lab. Computer Sys., Inc.</i> , 308 F.3d 1304 (Fed. Cir. 2002)	13
<i>Tec Air, Inc. v. Denso Mfg. Mich., Inc.</i> , 192 F.3d 1353 (Fed. Cir. 1999)	51
<i>Trintec Indus., Inc. v. Top U.S.A. Corp.</i> , 295 F.3d 1292 (Fed. Cir. 2002)	13
<i>Univ. of Strathclyde v. Clear-Vu Lighting LLC</i> , 17 F.4th 155 (Fed. Cir. 2021)	31
<i>Virtek Vision Int’l ULC v. Assembly Guidance Sys., Inc.</i> , 97 F.4th 882 (Fed. Cir. 2024)	29
<i>Wang Labs., Inc. v. Toshiba Corp.</i> , 993 F.2d 858 (Fed. Cir. 1993).....	37, 41

Wyers v. Master Lock Co.,
616 F.3d 1231 (Fed. Cir. 2010) 34

AGENCY DECISIONS

Apple Inc. v. Cellular Comm’ns Equipment, LLC,
IPR2015-00576, Paper 7 (June 12, 2015).....23

Edwards Lifesciences Corp. v. Aortic Innovations LLC,
IPR2023-01232, Paper 10 (Feb. 7, 2024).....30

F5 Networks, Inc. v. Radware, Ltd.,
IPR2017-00124, Paper 48 (Apr. 23, 2018).....24

RPX Corp. v. Parity Networks, LLC,
IPR2018-00097, Paper 7 (Apr. 24, 2018).....29

REGULATIONS

37 C.F.R. § 42.6(a)(3) 13

EXHIBIT LIST	
2001	Declaration of Nathan Lowenstein in Support of Motion for <i>Pro Hac Vice</i> Admission
2002	Declaration of Colette Woo in Support of Motion for <i>Pro Hac Vice</i> Admission
2003	Declaration of Aviel D. Rubin, Ph.D.
2004	Reserved
2005	Bruce Schneier Curriculum Vitae [Schneier-CV]
2006	Bruce Schneier, <i>Why Cryptography Is Harder Than It Looks</i> , SCHNEIER ON SECURITY, 1997, https://www.schneier.com/essays/archives/1997/01/why_cryptography_is.html [Schneier-Why-Cryptography-Is-Harder-Than-It-Looks]
2007	Excerpts from Jonathan Katz et al., <i>Introduction to Modern Cryptography</i> (1st ed. 2007) [Katz]
2008	Bruce Schneier, <i>Security Pitfalls in Cryptography</i> , SCHNEIER ON SECURITY, 1998, https://www.schneier.com/essays/archives/1998/01/security_pitfalls_in.html [Schneier-Security-Pitfalls-in-Cryptography]
2009	Excerpts from Ross Anderson, <i>Security Engineering: A Guide to Building Dependable Distributed Systems</i> (1st ed. 2001) [Anderson]
2010	Excerpts from Bruce Schneier, <i>Applied Cryptography, Second Edition: Protocols, Algorithms, and Source Code in C</i> (2nd ed. 2015) [Schneier-Applied-Cryptography]
2011	Patent Application Publication to Tsai US 2005/0195857 A1 [Tsai]
2012	Edgar Pek et al., <i>Formal Verification of Communication Protocols in Distributed Systems</i> , SYMPOSIUM ON OPERATING SYS. PRINCIPLES, 2003 [Pek]

2013	Jan Friso Groote et al., <i>A Bounded Retransmission Protocol for Large Data Packets: A Case Study in Computer Checked Algebraic Verification</i> , 1996 [Groote]
2014	Microsoft, <i>Overview of MPEG-2 Systems</i> , WWW.LEARN.MICROSOFT.COM , July 27, 2023, https://learn.microsoft.com/en-us/windows/win32/directshow/overview-of-mpeg-2-systems [Overview-of-MPEG-2-Systems]
2015	Excerpts from Harry Newton, <i>Newton's Telecom Dictionary</i> (16th ed. 2000) [Newton's-Telecom-Dictionary]
2016	Excerpts from <i>Oxford Dictionary of Computing</i> (1996) [Oxford-Dictionary-of-Computing]
2017	IBM, <i>Transmission Control Protocol (TCP)</i> , WWW.IBM.COM , https://www.ibm.com/docs/en/zos-basic-skills?topic=4-transmission-control-protocol-tcp [IBM]
2018	<i>Security First Innovations, LLC v. Google LLC</i> , 2:23-cv-00091 (JKW-RJK), Google's Opening Claim Construction Brief (December 6, 2023) [Google's-Opening-Claim-Construction-Brief]
2019	Excerpts from Microsoft Press, <i>Microsoft Computer Dictionary</i> (5th ed. 2002) [Microsoft-Computer-Dictionary]
2020	<i>Security First Innovations, LLC v. Google LLC</i> , 2:23-cv-00091 (JKW-RJK), Google's Responsive Claim Construction Brief (Jan. 8, 2024) (Non-Confidential Redacted Version) [Google-Responsive-Claim-Construction-Brief]
2021	Cambridge Dictionary, <i>Set-Top Box</i> , https://dictionary.cambridge.org/us/dictionary/english/set-top-box (last visited Feb. 27, 2024)

2022	Jim Connors, INFOWORLD, <i>The evolution of cable television to interactive communications service provider</i> (Oct. 1, 1996), https://www.infoworld.com/article/2077255/the-evolution-of-cable-television-to-interactive-communications-service-provider.html [InfoWorld]
2023-2027	Reserved
2028	Declaration of Colette Woo [SERVED, NOT FILED]
2029	Jonathan Katz et al., <i>Introduction to Modern Cryptography</i> (1st ed. 2007) [SERVED, NOT FILED]
2030	Reserved
2031	Ross Anderson, <i>Security Engineering: A Guide to Building Dependable Distributed Systems</i> (1st ed. 2001) [SERVED, NOT FILED]
2032	Bruce Schneier, <i>Applied Cryptography, Second Edition: Protocols, Algorithms, and Source Code in C</i> (2nd ed. 2015) (E-Book) [SERVED, NOT FILED]
2033	Second Declaration of Aviel D. Rubin, Ph.D. [Rubin-2 nd -Decl.]
2034	Deposition Transcript of Samrat Bhattacharjee [Bhattacharjee-Depo.]
2035	C. Patrikakis et al., <i>Streaming Content Wars: Download and Play Strikes Back</i> , UCMedia 2009, LNICST 40, pp. 218–226, 2010. [Patrikakis]
2036	J. Hur et al., <i>Packet Chunking for File Detection</i> , IEEE/ACM Transactions on Networking Vol. 31 (2023). [Hur]
2037	R. Li et al., <i>A Framework for Qualitative Communications Using Big Packet Protocol</i> , 2019, https://doi.org/10.48550/arXiv.1906.10766 . [Li-I]

2038	N. McKeown et al., <i>The Tiny Tera: A Packet Switch Core</i> , IEEE Micro 17, 1998. [McKeown]
2039	<i>Marlin DRM FAQ</i> , Marlin Community, https://www.marlin-community.com/faq/ (last accessed Sept. 9, 2024) [Marlin-DRM]
2040	<i>What is Conditional Access?</i> , Verimatrix anti-piracy (Aug. 1, 2023), https://www.verimatrix.com/anti-piracy/knowledge-base/what-is-conditional-access/ [Verimatrix]
2041	R. Li et al., <i>New IP: A Data Packet Framework to Evolve the Internet</i> , 2020, https://www.researchgate.net/publication/341582478 [Li-II]
2042	Vimal Kumar et al., <i>POP: Privacy and Integrity Preserving Data Aggregation in Wireless Sensor Networks</i> , PROCEEDINGS OF THE IEEE SYMPOSIUM ON RELIABLE DISTRIBUTED SYSTEMS, September 2013 [Kumar]
2043	Yuxin Mao et al., <i>A Feedback-Based Secure Path Approach for Wireless Sensor Network Data Collection</i> , SENSORS, October 22, 2010 [Mao]
2044	Institution Decision in IPR2024-00214 [214-ID]

I. INTRODUCTION

Petitioner fails to prove that any of the challenged claims are invalid. The Board has, rightly, twice rejected Petitioner's Orsini ground. *See* Section II. Petitioner's Foster-Hayhurst combination similarly fails because Petitioner fails to prove that Hayhurst is analogous art, that a POSITA would be motivated to make the combination, or that Foster-Hayhurst teaches the claims. *See* Section III.

II. AS THE BOARD HAS TWICE FOUND, THE ORSINI GROUND FAILS (ALL CLAIMS, GROUND 1).

Petitioner's first ground (anticipation/obviousness over Orsini) has been twice rejected by the Board. ID, 39; Ex. 2044 [214-ID] 51. Even prior to the institution decisions, the claims issued over Orsini in prosecution. *See* Section II.A. Petitioner's anticipation (Section II.B) and obviousness (Section II.C) arguments again fail.

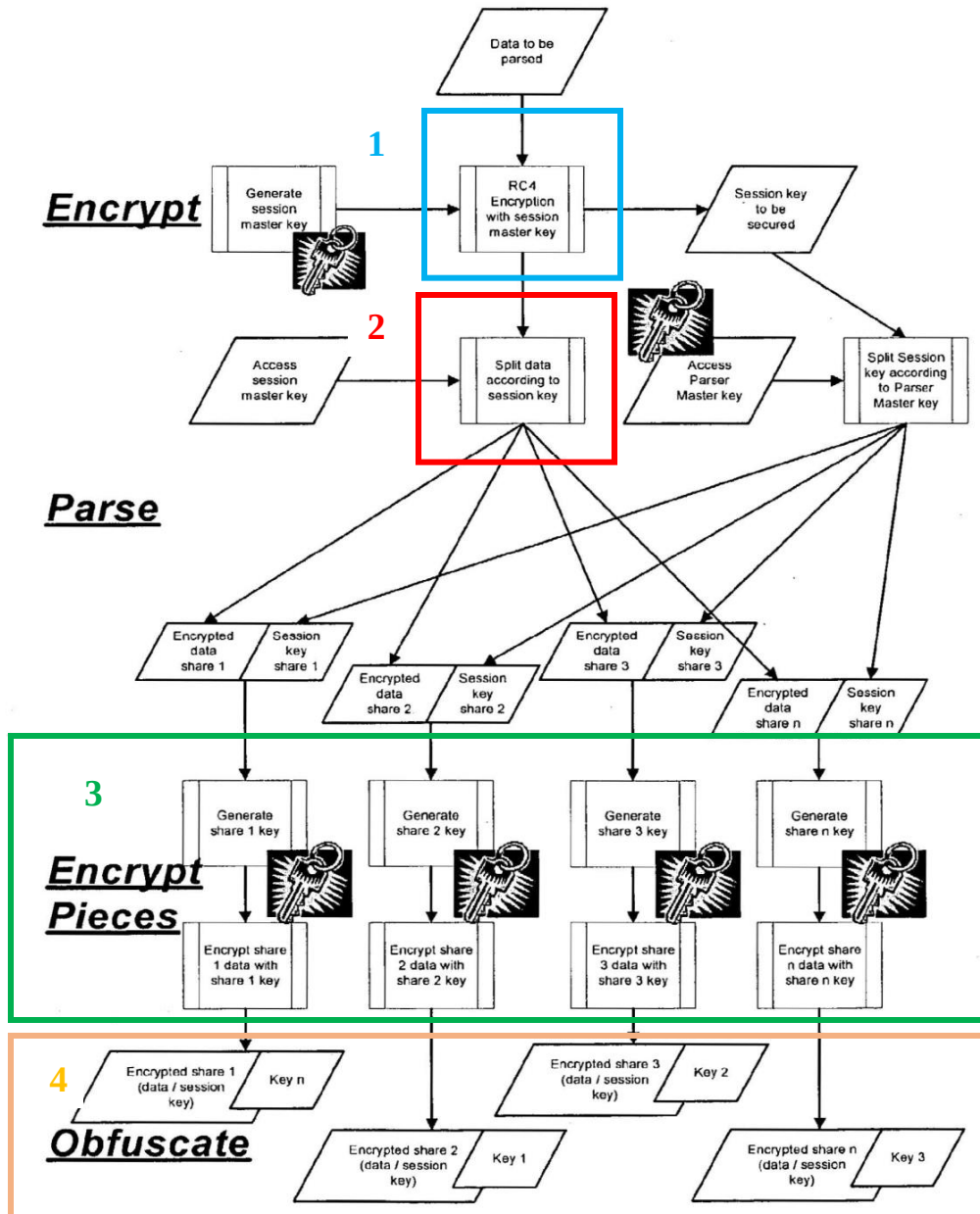
A. The '854 Patent Claims Fundamentally Different Technology Than That Disclosed In Orsini.

Orsini is Patent Owner's application. Figures 1-25 and paragraphs 1-23 of Orsini are included in the '854's specification. The '854, however, includes **17 new figures and over eighteen columns of additional material** (~10,500 words) which contain significantly different technology than that disclosed in Orsini. *See* Ex. 1001 ['854] Figs. 26-41, 6:44-6:62, 8:10-8:61, 66:4-83:10. The claims-at-issue rely on this new matter.

Petitioner's Ground 1 relies, almost entirely, on Orsini's Figure 21. *E.g.*, Pet., 8-15, 17, 29-30, 31, 45-46. Orsini's Figure 21 and the '854's claim 1 are each directed to methods for securing a "dataset." Both methods rely, in part, on dividing the dataset into pieces (called "chunks" or "shares") and encrypting the chunks.

From here, however, the methods diverge. As Dr. Aviel Rubin explains, Figure 21's method of securing data is centered around encrypting the dataset with a key called the "session master key," parsing the dataset and the session master key into shares, applying a share key to each of the shares, and then non-cryptographically storing the share keys with a share it was not used to encrypt:

As can be seen below, Figure 21 teaches **(1) encrypting the entire data set with the session master key**, **(2) parsing the data set and session master key into shares**, then **(3) encrypting those shares and split session master key with a set of share keys**, followed by a **(4) non-cryptographic** step:

Figure 21

Ex. 1005 [Orsini] Fig. 21 (annotated), [0293].

Ex. 2033 [Rubin-2nd-Decl.] ¶ 37.

The '854 discloses and claims a materially different and improved method for securing a dataset that addresses a potential weakness in Figure 21's method. Ex. 2033 [Rubin-2nd-Decl.] ¶ 43. The '854's innovation is embodied in claim 1's limitation requiring "performing an encryption operation based on the external key to further secure the plurality of data chunks." *Id.*, ¶¶ 42-43. Rather than non-cryptographically securing the keys, an "external key" (corresponding to the "workgroup key" of the '854's disclosure) further secures the shares/chunks with **an encryption operation**. This step, which is not disclosed in Orsini, is rooted in the inventors' insight that Orsini's method potentially allows an attacker to reconstruct the dataset if the attacker has access to each of the shares; the '854's claimed inventions eliminate that potential vulnerability. *Id.*, ¶¶ 40-41.

Beyond being different, Figure 21's method is simply incompatible with claim 1 and thus a POSITA would have no motivation or reasonable expectation of success in modifying Orsini to achieve claim 1. Petitioner equates the claimed "external key" with the "session master key" of Orsini's Figure 21. Claim 1 requires generating data chunks from a data set, first encrypting the data chunks with a distinct encryption key before "**further** securing" the "plurality of data chunks" with an "**encryption** operation based on the external key." Pet., 14, 33-39. Petitioner contends a POSITA would modify Orsini to use the session master key to "further secure" the dataset after encrypting the shares. But, as discussed, in

Figure 21, prior to this “further securing” step the session master key was already divided into pieces, the pieces were integrated into the shares, and the session master key/share combinations (each containing a different piece of the session master key) were encrypted with share keys. Thus, by the time the dataset is “further secured” per claim 1, the session master key in Orsini is buried within encrypted shares, and any further use of it would entirely negate the prior steps in Orsini’s method. Ex. 2033 [Rubin-2nd-Decl.] ¶ 44.¹ Critically, to use the already parsed and encrypted session master key in an “encryption operation” to “further secure the plurality of data chunks” as Petitioner suggests, *a priori* knowledge of the session master key would be needed to unpack the encrypted shares. This

¹ There is no dispute that these steps must be performed in the claimed order. Ex. 1064 [Google-Proposed-Claim-Construction] 16 (steps “must be performed in order”); ID, 27-34; *Mantech Envtl. Corp. v. Hudson Envtl. Servs.*, 152 F.3d 1368, 1375–76 (Fed. Cir. 1998) (“steps must be performed in order” where “the sequential nature of the claim steps is apparent from the plain meaning of the claim language”). Here, without being already “encrypt[ed] ... with distinct encryption keys,” the plurality of data chunks could not be “**further** secure[d]” with an “encryption operation based on the external key.” Ex. 2033 [Rubin-2nd-Decl.] ¶ 46.

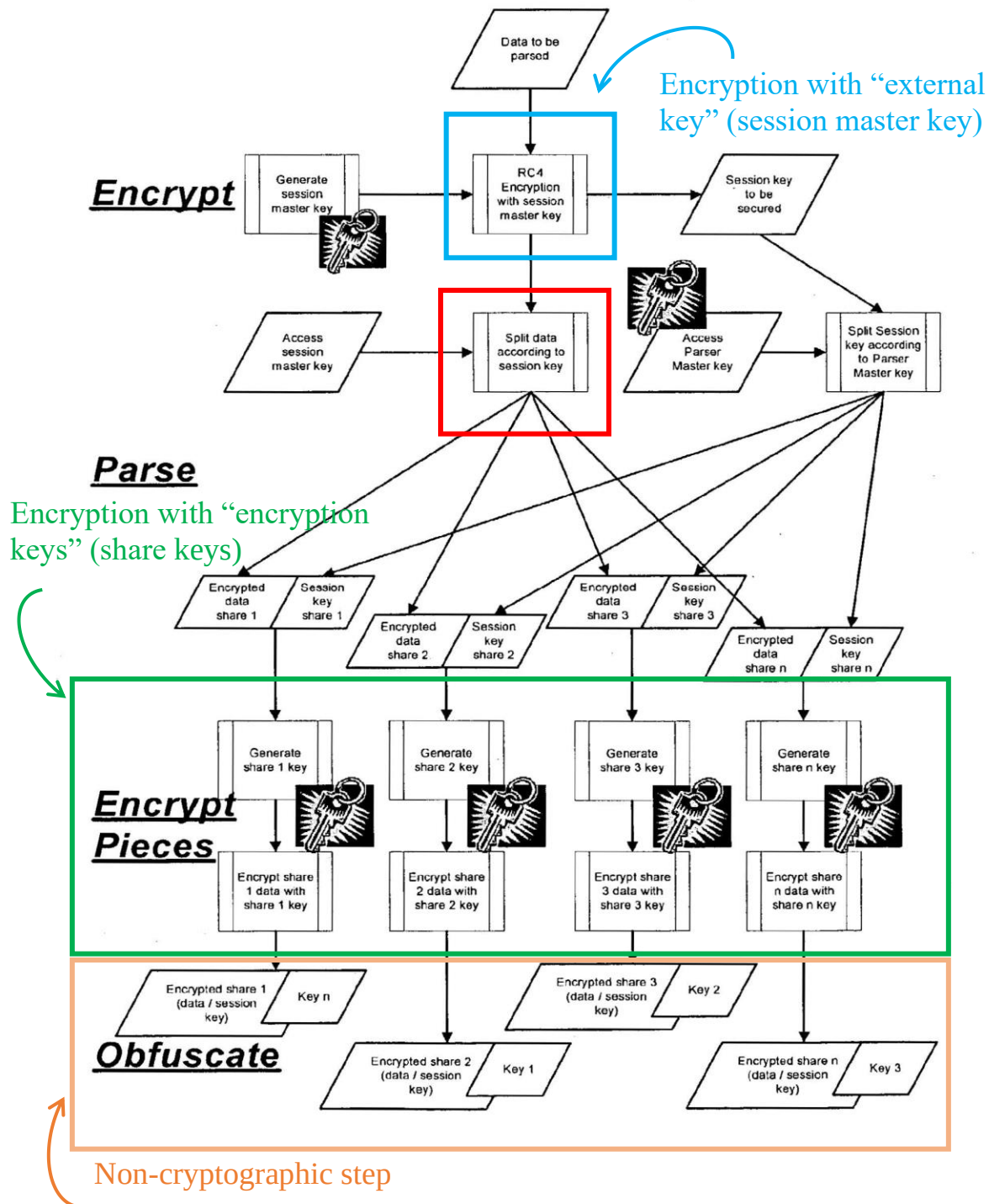
would require complex and undisclosed changes to Figure 21's method due to the need for separate storage of the session master key and would render entirely unnecessary the steps Figure 21 does disclose to parse and store the session master key within the encrypted shares. *Id.*; see also ID, 33-34.

B. The Board Correctly Found That Petitioner Fails To Show That Orsini Anticipates The Claims.

The Board correctly found that “Petitioner has not shown that Orsini anticipates claim 1” because Orsini’s Figure 21 does not expressly or inherently disclose “further securing” the “plurality of data chunks” with an encryption operation based on the external key.” ID, 27-39. As Dr. Rubin explains:

Assuming, as Petitioner does (*e.g.*, Pet., 10, 17, 33-38) that Orsini’s “session master key” corresponds to the claimed “external key,” Figure 21 undisputedly teaches **an encryption operation performed on the entire data set with the session master key** (an “external key”) **before parsing the data set and session master key into shares** and **encrypting those shares and split session master key with a set of share keys**. And any additional protection of the data chunks is achieved through a **non-cryptographic step**:

Figure 21



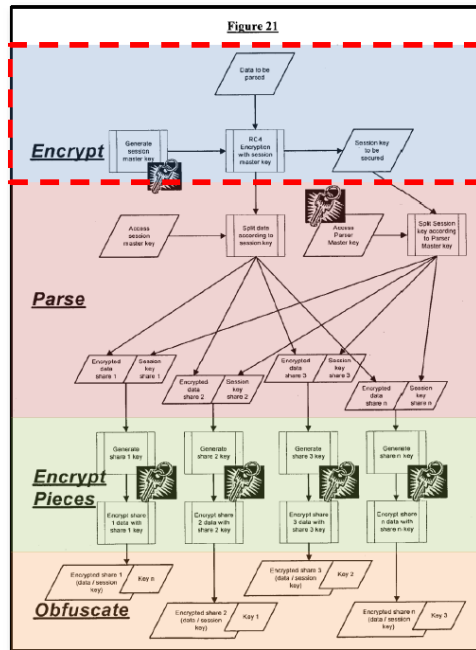
Ex. 1006 [Foster] Fig. 21 (annotated).

Orsini, however, does not disclose performing any “encryption operation . . . to further secure the plurality of data chunks” “based on the external [session master] key.” Indeed, Orsini ***cannot disclose this***, because in Figure 21, after the session master key/external key is used to encrypt the dataset as a whole, it is “separat[ed]” along with the data set and those “portions of the session master key” are “contain[ed]” in the chunks/shares of data which are then encrypted. Ex. 1005 [Orsini] [0293], [0297]-[0299]; *compare id. with* Ex. 1001 [’854] 80:30-33 (using workgroup key instead of session master key). Consequently, by the time the data set has been parsed into shares and those shares have been encrypted, the session master key (*i.e.*, the alleged “external key”) cannot be used for an encryption operation without removing the benefits provided from Orsini’s splitting and encrypting of that key.

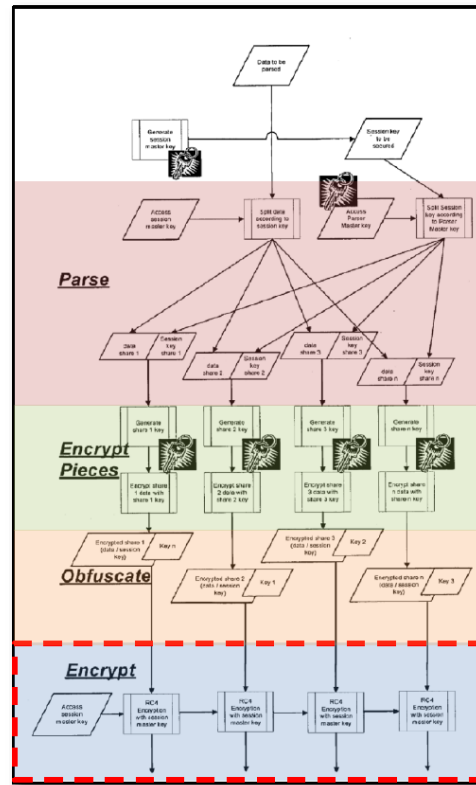
Ex. 2033 [Rubin-1st-Decl.] ¶¶ 49-50.

Rather than pointing to any actual disclosure in Orsini, Petitioner ***alters*** Figure 21 to add another, undisclosed, chunk encryption step, this time using the session master key, after the data chunks are generated and encrypted a first time with distinct encryption keys as shown in Petitioner’s figures below:

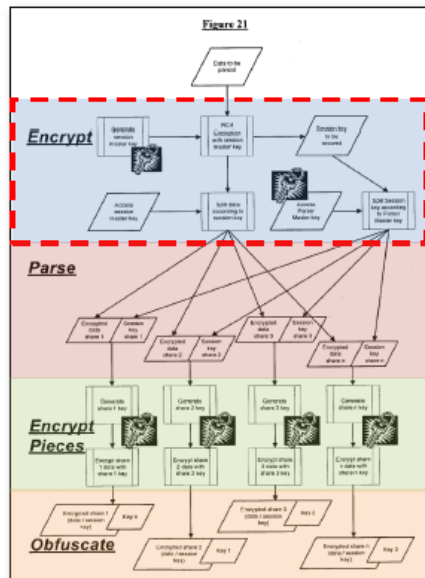
Original



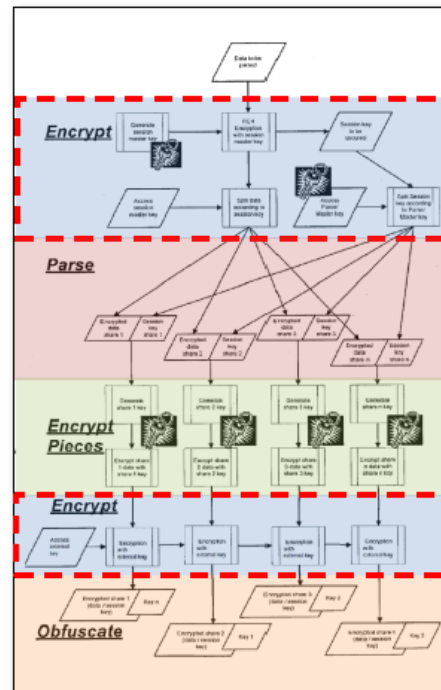
Different-Order Version



Original

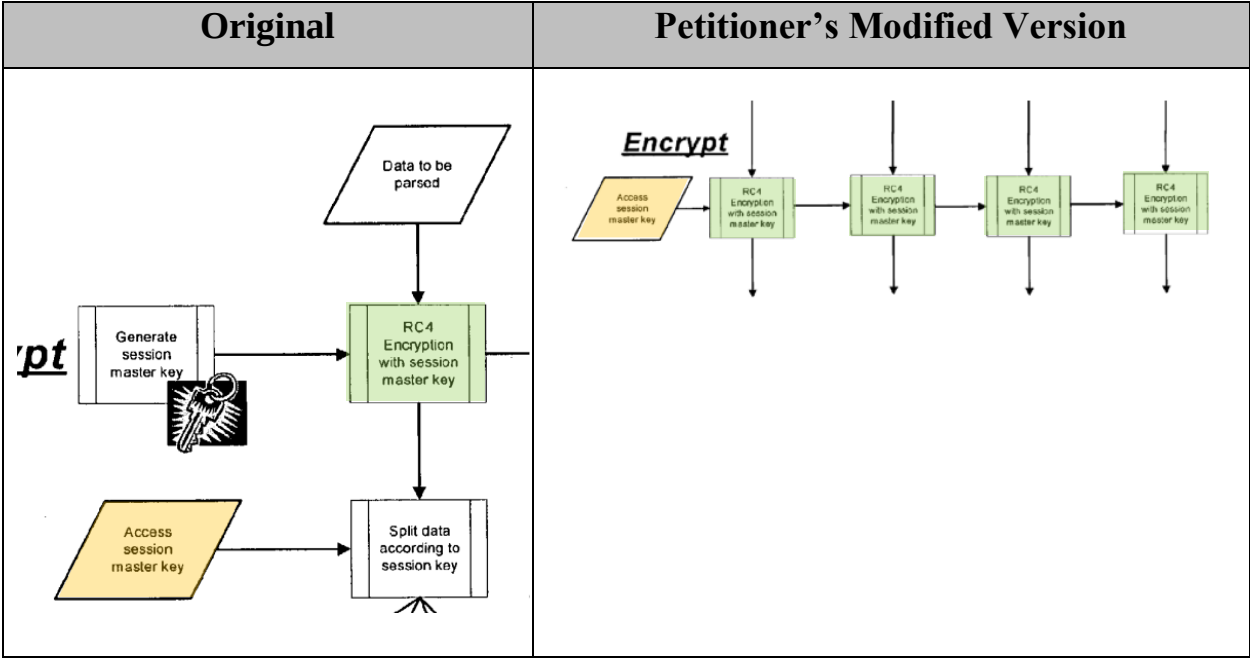


Two "Encrypt" Phases (Version 1)



Pet., 15, 37.

Not only does Petitioner, without explanation, move the dataset encryption step from the beginning of Figure 21 to the end, but it also then entirely re-writes the step (again, without explanation) **to apply to each of the chunks rather than to the dataset as a whole**. Thus, Petitioner does not merely reorder Figure 21, it fundamentally alters what is being encrypted (the dataset versus the chunks/shares). Petitioner additionally moves the step **wherein the session master key is accessed from prior to the data being parsed to after the non-cryptographic step**. Ex. 2033 [Rubin-2nd-Decl.] ¶ 52.



As the Board found, “unlike Orsini’s Original Figure 21 (left), Petitioner’s Different-Order Version lacks an initial encrypt step. Instead, there is an encryption step after the obfuscate step.” ID, 28 (cleaned up).

Petitioner’s and Dr. Bhattacharjee’s reliance upon materially-altered versions of Figure 21 establishes as a matter of law that Orsini does not anticipate claim 1. *Enplas Display Device Corp. v. Seoul Semiconductor Co., Ltd.*, 909 F.3d 398, 405 (Fed. Cir. 2018) (“Prior art that must be modified to meet the disputed claim limitation does not anticipate the claim.”). It was for this reason that the Board correctly found that Petitioner had not shown that Orsini expressly discloses the claimed inventions. ID, 28, 30.

1. Orsini Does Not Inherently Disclose Claim 1.

Lacking an express disclosure of the “further securing” step, Petitioner nonetheless contends that Orsini suggests the step to a POSITA in its statement that “[i]t is readily apparent to those of ordinary skill in the art that certain steps of the methods described herein may be performed in different order, or repeated multiple times, as desired.” Pet., 30-36 (citing Ex. 1005 [Orsini] [0301]). But, such boilerplate language is not sufficient to disclose a claim limitation, for “[a] [prior art] patent discloses only that which it describes” *In re Benno*, 768 F.2d 1340, 1346 (Fed. Cir. 1985); *Forum US, Inc. v. Flow Valve, Ltd. Liab. Co.*, 926 F.3d 1346, 1352 (Fed. Cir. 2019) (“the boilerplate language that modifications

can be made to the original disclosed invention does not even suggest an arbor-less embodiment of the disclosed invention.”); *D Three Enters., LLC v. SunModo Corp.*, 890 F.3d 1042, 1050-51 (Fed. Cir. 2018) (similar).

Equally important, Petitioner does not identify anything in Orsini that could be “performed in a different order” or “repeated” to further secure the shares (per claim 1), in contrast to the dataset as a whole. No matter how many times one repeats the step of encrypting ***the original data set*** using the session master key, and regardless of the order in which that step is carried out, it cannot disclose encrypting ***the chunks***. Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 53, 56.

Instead, Petitioner asserts that a POSITA would understand that its proposed alterations of Figure 21 were taught by Orsini. Petitioner claims that “POSAs understood that one ‘certain step[]’ capable of being reordered was the first ‘encrypt’ step with the session master key in Orsini’s Figure 21 [] because it was well-known to reorder whether an encryption step is performed before and/or after splitting.” Pet., 33 (citing Ex. 1003 [Bhattacharjee-Decl.] ¶¶ 110-111).

But, to anticipate, a reference must disclose each limitation either expressly or inherently. An anticipation argument based on what a POSITA might “understand” must meet the requirements for an inherent disclosure. *Microsoft Corp. v. Biscotti, Inc.*, 878 F.3d 1052, 1065 (Fed. Cir. 2017) (statements that “a [POSITA] would understand the [evidence] as disclosing [the limitation]” could

support an argument that a limitation was “inherently disclose[d] ...”); *LBT IP I LLC v. Apple Inc.*, 2023 U.S. App. LEXIS 14376, *8 (Fed. Cir. 2023) (similar). Inherency requires demonstrating that a limitation is “‘necessarily present,’ not merely probably or possibly present, in the prior art.” *Trintec Indus., Inc. v. Top U.S.A. Corp.*, 295 F.3d 1292, 1295 (Fed. Cir. 2002). Petitioner, however, does not even sufficiently show a POSITA’s understanding, much less that Orsini “necessarily” discloses performing an encryption operation to further secure the chunks based on the external/session master key.

As to a POSITA’s understanding, Petitioner provides only a conclusory statement with a single citation to its expert declaration, no explanation or analysis, and a string of unexplained citations in its expert’s declaration, none of which is sufficient to show anticipation. *See* Pet., 33; 37 C.F.R. § 42.6(a)(3); *Schumer v. Lab. Computer Sys., Inc.*, 308 F.3d 1304, 1315-16 (Fed. Cir. 2002) (conclusory testimony is “insufficient”). In accord, the Board found that Petitioner’s conclusions were insufficiently supported. ID, 30.

But even if it were considered, Orsini’s statement that “certain steps may be performed in different order, or repeated multiple times” does not ***necessarily disclose*** “performing an encryption operation based on the external key to ***further secure the plurality of data chunks.***” That “certain” steps can be repeated does not mean that the data set encryption step would necessarily be repeated, and

certainly does not establish that the chunks, each encrypted with a separate share key, would ***necessarily*** be again encrypted using the session master key, as required by Petitioner’s argument. As Dr. Rubin explains:

Orsini, moreover, only specifies that “***certain*** steps” “may be performed in different order.” Ex. 1005 [Orsini] [0301]. Orsini does not teach that the encryption of the data set step is one such step. Rather, Orsini discloses that “[f]or example, multiple parsing steps may be performed on only one portion of the parsed data,” and that “[e]ach portion of parsed data may uniquely secured.” *Id.* Orsini, therefore, only identifies the parsing and encryption with the plurality of distinct encryption keys as steps which may be repeated or reordered, and on “only one portion of the parsed data” at a time at that. There is simply no disclosure that would necessitate that the data chunks, after already being individually encrypted with the alleged plurality of distinct encryption keys, would be encrypted again, let alone with the alleged external key, or that the “plurality of data chunks” rather than an individual data chunk would be encrypted based on the external key. Indeed, a POSITA would recognize that encryption with the alleged external key would not be an appropriate step to repeat or reorder because it is parsed along with the data set into portions and those portions are then encrypted. Consequently, encrypting the shares with the alleged external key (*i.e.*, the session master key) would render the splitting and encryption of the alleged external key pointless.

Ex. 2033 [Rubin-2nd-Decl.] ¶ 55.

The Board agreed that Petitioner’s arguments go “beyond mere” “reordering” and “repetition” and fundamentally change Figure 21. *Id.*, 28-29, 33-34. The Board “credit[ed] Dr. Rubin’s testimony ... over Dr. Bhattacharjee ... on th[ese] issue[s]” and correctly found that “Petitioner has not shown that Orsini discloses” or “anticipates” the Different-Order or Two-Encrypt-Phases versions. *Id.*

2. Petitioner’s Other Arguments Cannot Establish Anticipation Of Claim 1.

Petitioner’s expert cites a number of references for the generic proposition that it was known to encrypt data before and/or after fragmenting a dataset. Ex. 1003 [Bhattacharjee-Decl.] ¶¶ 111, 113 (citing Exs. 1028, 1053, and 1045). But his/Petitioner’s invalidity grounds are not based on these references. They have no bearing upon what **Orsini** necessarily discloses. Orsini does not disclose, expressly or implicitly, “further securing” “data chunks” via an “encryption operation” based on an “external key.”

Additionally, Petitioner points to certain dependent claims in Orsini that claim encrypting shares/chunks a second time using “a different encryption algorithm.” Pet., 35 (citing Ex. 1005 [Orsini] cls. 12, 29, 39). But these claims do not expressly or inherently disclose claim 1’s limitations. As Dr. Rubin explains:

Dependent claims 12, 29, and 39, directed to repeating encryption of shares with a “different encryption algorithm,” do not

disclose claim 1, which requires that the encrypted data shares undergo an “encryption operation ***based on the external key*** to further secure the plurality of data chunks.” Simply repeating encryption using a “different algorithm” does not even suggest, much less necessarily disclose, that the “different algorithm” would be based on the session master key. See Ex. 1005 [Orsini] [0071] (explaining “a wide number of algorithms [are] available”).

Ex. 2033 [Rubin-2nd-Decl.] ¶ 58; ID, 33 (cited claims “provide little guidance on how the different encryption algorithm works.”).

The dependent claims cited are, moreover, different embodiments from Figure 21. Ex. 2033 [Rubin-2nd-Decl.] ¶ 59. Petitioner may not “mix and match” features from different unconnected embodiments to anticipate claim 1. “‘It is not enough that the prior art reference . . . includes multiple, distinct teachings that the artisan might somehow combine to achieve the claimed invention’ to show anticipation.” *BlephEx, LLC v. Myco Indus.*, 24 F.4th 1391, 1400 (Fed. Cir. 2022) (citations omitted); *Net MoneyIN, Inc. v. VeriSign, Inc.*, 545 F.3d 1359, 1371 (Fed. Cir. 2008) (anticipation requires “the limitations [be] arranged or combined in the same way as recited in the claim”); *Finisar Corp. v. DirecTV Group, Inc.*, 523 F.3d 1323, 1334 (Fed. Cir. 2008) (similar). As a matter of law, the dependent claims cannot show that the Figure 21 embodiment on which Petitioner relies discloses, expressly or inherently, claim 1’s invention.

Finally, Petitioner suggests that Orsini’s disclosure of the use of public keys, in another context, suggests the “further securing” limitation of claim 1. Petitioner argues that a public key could be the claimed “external key,” arguing that “Orsini’s public encryption keys are additional ‘*external keys*’” used “to further encrypt already-encrypted ‘*data chunks*.’” Pet., 38-39 (citing Ex. 1005 [Orsini] [0095], [0097], [0107], [0119], [0172]). However, Orsini does not disclose, expressly or inherently, the use of a public key in connection with Figure 21 and none of the embodiments Petitioner cites are rooted in the Figure 21 embodiment. Ex. 2033 [Rubin-2nd-Decl.] ¶ 61; ID, 34 (“As Patent Owner points out, the public key is not shown in Figure 21. ... Rather, Petitioner combines the features from multiple embodiments to arrive at the claimed invention.”). Petitioner does not even allege a POSITA would be motivated to combine Orsini’s embodiments or modify Orsini’s Figure 21 embodiment in view of these disclosures. See Pet., 38-44.

Thus, as the Board correctly found, Petitioner’s reliance upon Orsini’s public-key encryption fails because the use of public keys disclosed in Orsini has nothing to do with Figure 21. ID, 34. As Dr. Rubin explains:

Orsini’s depository or data storage facility teachings (Ex. 1005 [Orsini] [0095], [0097], [0119]) are of no help as Petitioner does not allege that the claimed data chunks are stored there. Indeed, Petitioner maps the “plurality of storage devices” to Orsini’s “storage devices” rather than Orsini’s “data depositories” and never argues that Orsini’s

storage devices can perform public-key encryption. See Pet., 46.... Therefore, Petitioner has not identified any entity in Figure 21 that performs public-key encryption. The claim, moreover, requires “performing **an** encryption operation based on the external key to further secure the **plurality of data chunks**.” Assuming for the sake of argument that the encrypted chunks are eventually distributed and stored on the depository or data storage facilities, an encryption step performed at a depository or data storage facility would only secure a single chunk rather than “the plurality of data chunks.”

Petitioner also cites Ex. 1005 [Orsini] [0097] and [0172], but these do not disclose public-key encryption of already-encrypted chunks/shares. See *id.*, [0097] (“**data** from the transaction engine 205 or the depository 210, which was encrypted with a corresponding public key”); [0172] (“the requesting application or vendor uses the public key of the user, for example, to encrypt the session key that will be used to encrypt the **document or message**”). Indeed, these embodiments never parse data into chunks or shares, let alone encrypt chunks with distinct encryption keys and then perform an encryption operation based on an external key to further secure the plurality of chunks. *Id.*, [0097], [0168]-[0172].

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 62-63.

C. Petitioner Fails To Show That Orsini Renders The Claims Obvious.

Petitioner alternatively contends that claim 1 is obvious over Orsini, relying heavily upon a POSITA’s purported common sense. As the Board correctly found,

Petitioner's arguments are "unsupported and conclusory," "circular," and "based on impermissible hindsight." ID, 35-36; *id.* ("not credit[ing] Dr. Bhattacharjee's testimony on this issue."); 36-37 (similar for "externally received group key"); 38 (similar for "cryptographically reordering or shuffling the data"); 38-39 (similar for "externally-stored KEK"). If more is needed, Petitioner's argument fails for the additional reasons stated below.

1. Petitioner Cannot Rely On Common Sense To Supply Missing Limitation 1[c].

Instead of combining Orsini with another reference, Petitioner contends a POSITA "had multiple reasons to implement Orsini to perform" limitation 1[c] ("performing an encryption operation based on the external key to further secure the plurality of data chunks"). Pet., 39; 39-44 (describing allegedly "known" encryption operations and asserting POSAs "had reason" to implement them in Orsini's Figure 21 method).

But "common sense" can be used to supply a missing limitation "**only** when 'the limitation in question [is] unusually simple and the technology particularly straightforward.'" *DSS Tech. Mgm't v. Apple Inc.*, 885 F.3d 1367, 1374-75 (Fed. Cir. 2018) (citations omitted). And where, as here, a "missing limitation goes to the heart of an invention" a court's review of any purported "reasoned basis for

resort to common sense” must be especially “searching” *Arendi S.A.R.L. v. Apple Inc.*, 832 F.3d 1355, 1363 (Fed. Cir. 2016).

In this case, the limitation-at-issue lies at the heart of the invention and involves complex technology. As Dr. Rubin explains:

The inventions-at-issue concern cryptography, which is a technologically-involved and complex field involving incredibly high stakes and safeguarding against a wide variety of highly sophisticated and ever-evolving cybersecurity attacks. ...

And, as previously discussed, the ’854’s new teachings and the claimed inventions arose from the inventor’s insight regarding a potential vulnerability in Orsini’s Figure 21 approach wherein if an attacker obtains each of the shares, the dataset can be reconstructed. The ’854 eliminates this vulnerability by encrypting the encryption keys using a workgroup key—corresponding to the “external key” of claim 1—a technique not disclosed in Orsini. The workgroup key provides “enhanced protection of information assets by encrypting key information stored within the shares” (Ex. 1001 [’854] 79:66-80:1) adding “an extra level of security to a secured data set” (*id.*, 77:22-23). As the ’854 explains “even if all required shares and external keys are discovered, an attacker has no hope of recreating the information without access to the workgroup key.” *Id.*, 80:1-4. Thus, this limitation goes to the heart of the invention. Moreover, given the benefits associated with the “workgroup key” delineated in the ’854, it further seems clear that the inventors of the ’854 (and Orsini) considered the

“workgroup key” concept to be a fundamentally different approach that solved a problem that Orsini did not recognize or address.

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 66-67.

Consequently, Petitioner’s attempt to rely on a POSITA’s common sense to add the missing limitation should be rejected.

2. Petitioner Fails To Show A POSITA Would Have Been Motivated To Modify Orsini to Add Limitation 1[c].

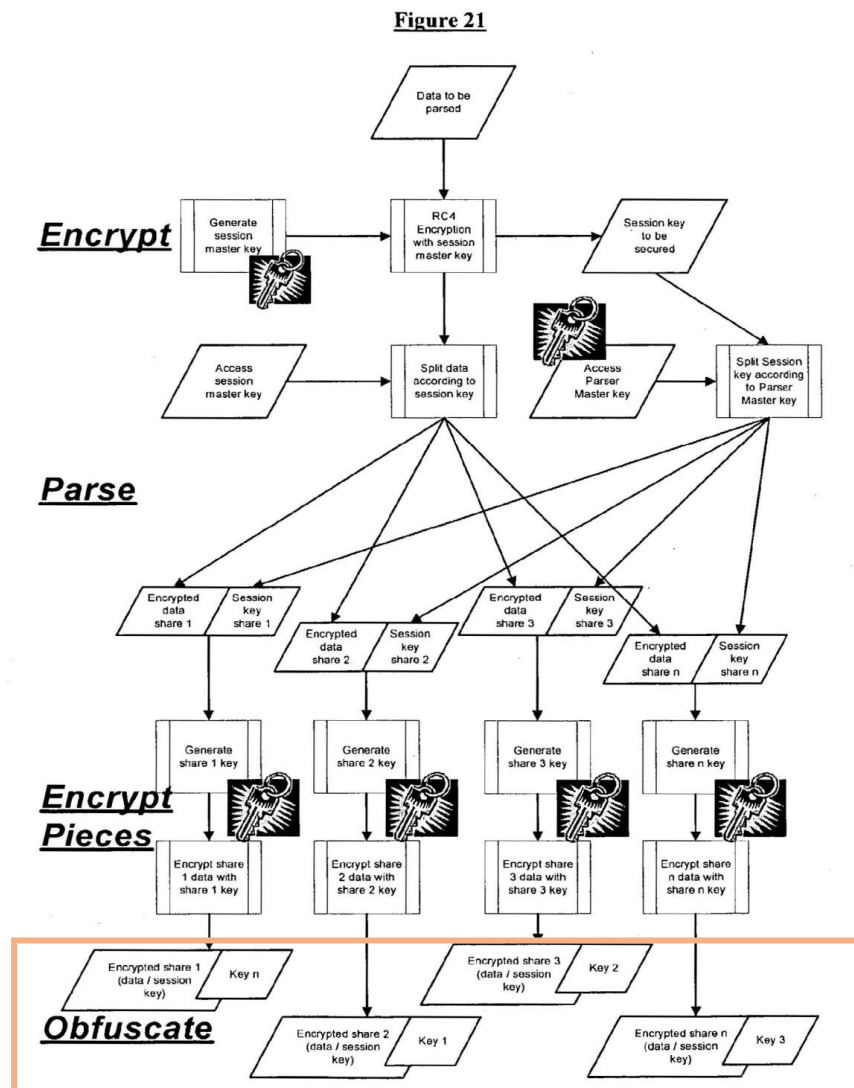
As discussed, like claim 1, Orsini includes the steps of generating and encrypting the chunks but does not disclose, expressly or inherently, limitation 1[c]: “performing an encryption operation based on the external key to further secure the plurality of data chunks.” See Section II.A. Petitioner suggests that the limitation is obvious because “POSAs would have been motivated” by Orsini’s disclosures to add this limitation because they “would have understood that doing so would provide additional security and efficiency benefits.” Pet., 39-40. Petitioner, however, fails to demonstrate that a POSITA would be motivated to modify Orsini to “perform[] an encryption operation based on the external key to further secure the plurality of data chunks.”

Petitioner’s purported motivation should be rejected because adding an “encryption operation based on the external key” to Orsini is useless because, unlike the ’854’s claim 1, Orsini *already* uses a *non*-cryptographic operation *that*

is not based on any key to further secure the data chunks. Ex. 1005 [Orsini] Fig.

21, [0299]. As Dr. Rubin explains:

Orsini already teaches hiding the encryption key used to encrypt each data chunk with a different chunk through a **non-cryptographic process**, illustrated in Figure 21 below.



Ex. 1005 [Orsini] Fig. 21. As can be seen in Figure 21, in this step, the system “store[s] the encryption keys in different locations from the

encrypted data portions or shares ...” *Id.*, [0299]; *see also* [0353]...; [0396]; cl. 26.

As Petitioner does not dispute, the step taught in Orsini’s Figure 21 is not an encryption operation (much less one involving the external key). *See generally* Pet.; Ex. 1003 [Bhattacharjee-Decl.]. As there is no encryption scheme or “scrambling” or altering of the data or the alleged distinct encryption keys (Ex. 2007 [Katz] 4), it cannot be an encryption operation.

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 69-70.

Petitioner fails to explain why a POSITA would be motivated to modify Orsini using other allegedly “known” methods, particularly when it non-cryptographically hides each share key with a share it did not encrypt, and teaches that this provides a “comprehensive package for data security.” Ex. 1005 [Orsini] [0380]; Ex. 2033 [Rubin-2nd-Decl.] ¶ 71. It is not obvious to modify a reference that already achieves the benefit supposedly served by the proposed modification. *In re NTP, Inc.*, 654 F.3d 1279, 1299 (Fed. Cir. 2011) (modification “based on improper hindsight reasoning” where it rendered existing structure “superfluous”); *Apple Inc. v. Cellular Comm’ns Equipment, LLC*, IPR2015-00576, Paper 7, 12 (June 12, 2015) (denying institution where “Petitioner ... fails to provide a specific explanation why [the primary reference] did not already accomplish [the asserted benefits] by itself.”); *F5 Networks, Inc. v. Radware, Ltd.*, IPR2017-00124, Paper

48, 64-65 (Apr. 23, 2018) (no motivation for modification that “serve[s] no purpose”).

To the extent Petitioner argues that a motivation to modify Orsini is found in the purported knowledge of a POSITA that “when ‘encryption keys are included with the encrypted message’ ‘[s]ecurity’ comes from ‘encrypt[ing] the encryption keys themselves’ with ‘a key encryption key’” (Pet., 43 (citing Exs. 1035, 1036)), that concern is addressed by Orsini’s non-cryptographic step. In this step, each share key is secured not by encryption but by hiding it “in different locations from the encrypted data portions or shares” that particular share key was used to encrypt. Ex. 1005 [Orsini] [0299]. Orsini teaches that this is sufficient because an attacker would not have “enough information to decrypt the” data chunk stored in each data depository. *Id.*, [0353]; Ex. 2033 [Rubin-2nd-Decl.] ¶ 72. Thus, in Figure 21’s data-securing method, the session master key, data chunks, and session keys are all either encrypted, parsed, and/or hidden, and thus each potentially vulnerable part of the encryption method is secured in some way. Ex. 2033 [Rubin-2nd-Decl.] ¶ 72.

In addition to its failure to demonstrate that its proposed modifications add any benefit to Orsini, Petitioner also fails to consider their drawbacks. “The Board must weigh the benefits and drawbacks of the modification against each other, to determine whether there would be a motivation to combine.” *Arctic Cat Inc. v.*

Polaris Indus., 795 Fed. App'x. 827, 833 (Fed. Cir. 2019); *Henny Penny Corp. v. Frymaster LLC*, 938 F.3d 1324, 1331-32 (Fed. Cir. 2019) (similar) *Arctic Cat Inc. v. Bombardier Rec. Prods.*, 876 F.3d 1350, 1363 (Fed. Cir. 2017) (“potential problems” with combination supported lack of motivation).

The drawbacks of Petitioner’s proposed modification of Orsini are considerable. As Dr. Rubin explains:

Simply concluding that adding an encryption layer would “benefit” Orsini’s method oversimplifies the myriad considerations a POSITA must make in cryptography, even for what may seem to be a “simple” or “small” change. For example, a POSITA would recognize that when fully considering the costs and benefits, an additional layer of encryption may not in reality benefit an encryption scheme or “further secure the plurality of the data chunks” as the claims require. Rather, an additional encryption step may in fact introduce a weakness that an attacker can exploit. *See, e.g.*, Ex. 2007 [Katz] 22 ...; Ex. 2008 [Schneier-Security-Pitfalls-in-Cryptography] ...; Ex. 2009 [Anderson] One such weakness is in the critical area of key management, as it is where systems tend to be most vulnerable. *See* Ex. 2010 [Schneier-Applied-Cryptography] 169

Thus, while adding an additional round of encryption with the same or additional keys may appear to provide improved security, doing so may, on balance, compromise security.

Petitioner's proposal to "re-encrypt data chunks with an external encryption key" (Pet., 36-37) has significant security flaws. Ex. 2010 [Schneier-Applied-Cryptography] 357 ("Encrypting a plaintext block twice with the same key, whether with the same algorithm or a different one, is not smart."). If Orsini were modified such that the session master key (the alleged "external key") were either reapplied or otherwise utilized after the data has been split, that would undermine Orsini's method which protects the alleged external key by splitting it amongst the data shares. And it would entail further complications. By involving the session master key after the splitting process, Petitioner's proposal would necessitate the key management and transport of the session master key, effectively making the division and inclusion of the session master key within the encrypted shares pointless. Under Orsini's existing method, the master session key has already been parsed among the data shares and encrypted. Consequently, the session master key which was previously secured in Figure 21 would be unnecessarily exposed in this proposed modification. ...

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 74-76.

To the extent Petitioner proposes that a POSITA would be motivated to modify Orsini's Figure 21 to use another key, such as a "group key," "encryption key," or "key-encryption key" (Pet., 40-45), Petitioner too fails to address the drawbacks associated with managing and securing such keys against discovery by an attacker. Per Dr. Rubin:

Petitioner and Dr. Bhattacharjee argue that “Orsini provides ‘external **session** key management’” and discloses a “single point of...key management” that could be modified to “centrally generat[e] Orsini’s session master keys (***and any of the conventional encryption keys [used in Petitioner’s obviousness argument]***).” Pet., 25-26 (citing Ex. 1005 [Orsini] [0176]-[0179], [0293], [0381]-[0386], [0412]). But **session** key management is very different than key management for longer-lived keys such as the other “conventional encryption keys” Petitioner relies upon, especially as Figure 21’s session keys (the session master key and share keys) are stored alongside the encrypted data. For example, “session keys should be changed at least daily” if not for each “particular communications session” while “key-encryption keys don’t have to be replaced as frequently,” perhaps “once a year.” Ex. 2010 [Schneier-Applied-Cryptography] 184, 47; *see also id.*, 184

This longer lifespan, however, makes potential compromise of a key-encryption key all the more consequential. *Id.* Consequently, a POSITA would recognize that a key management solution for session keys would not be appropriate for the “group key,” “encryption key,” or “key-encryption key” used in Petitioner’s alternative proposed modifications. *See* Ex. 1001 [’854] 62:28-29 (parser master, session master, and share encryption “keys may have individual key storage, retrieval, security and recovery options.”).

Moreover, key management for a “group key,” “encryption key,” or “key-encryption key” would necessarily be different than for key

management for the “session” keys in Figure 21. In Figure 21, the session master key used to encrypt the dataset is parsed within the data shares (Ex. 1005 [Orsini] [0297]) and the session share keys are stored along with the data shares (*id.*, [0299]). Therefore, so long as the party reconstructing the dataset possesses the shares, he or she also possesses the session keys necessary to decrypt the data. By contrast, the “group key,” “encryption key,” or “key-encryption key” would not be stored alongside the data but would rather need to be distributed out-of-band to the decrypting party. Consequently, those keys would need to be maintained and stored on both ends and in transit such that no adversary gets access to them. Thus, the threat model for the “group key,” “encryption key,” or “key-encryption key” is very different than for a “session key.” Petitioner and Dr. Bhattacharjee have not mentioned, let alone adequately addressed, these different threat models. Their superficial consideration of key management, an incredibly important aspect of cryptography, is thus insufficient and flawed.

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 78-80.

Petitioner attempts to justify its failure to properly consider the lack of benefits and substantial drawbacks of its proposed modification of Orsini by claiming that it is merely applying a “known” method disclosed in other references not included in Ground 1. Pet., 39-44. The Board properly rejected this argument, finding that the Petition failed to “explain how the cited prior art would have been used to modify Orsini” and “thus, it is not apparent how Petitioner intends to rely

on these references in its challenge.” ID, 37. Although references beyond a Petitioner’s grounds may “document the knowledge that skilled artisans would bring to bear in reading the prior art identified as producing obviousness,” *Ariosa Diagnostics v. Verinata Health, Inc.*, 805 F.3d 1359, 1365 (Fed. Cir. 2015), Petitioner cannot use them to supply a missing limitation, effectively subverting the requirement to properly identify reference combinations it relies upon. Nor can it use such references to establish a motivation to modify Orsini. *Virtek Vision Int’l ULC v. Assembly Guidance Sys., Inc.*, 97 F.4th 882, 886-7 (Fed. Cir. 2024) (“It does not suffice to meet the motivation to combine requirement to recognize that two alternative arrangements ... were both known in the art.”); *RPX Corp. v. Parity Networks, LLC*, IPR2018-00097, Paper 7, 14 (Apr. 24, 2018) (“knowledge in the art does not . . . show a motivation to combine existing components in the manner claimed free of hindsight motivation.”).²

² Petitioner cites *Intel Corp. v. PACT XPP Schweiz AG*, 61 F.4th 1373, 1380 (Fed. Cir. 2023) for the proposition that “if there’s a known technique to address a known problem using prior art elements according to their established functions, then there is a motivation to combine” (Pet., 37), but, it is inapposite because Petitioner does not attempt to show a motivation to combine Orsini with any of

Ultimately, as the Board found, Petitioner’s proposed modification of Orsini, and its purported motivation to make that modification, are entirely hindsight-driven, based on its knowledge today of what the inventors disclosed in the ’854 Patent and claimed. ID, 35-36 (“we agree ... that Petitioner’s reason to modify Orsini is based on impermissible hindsight.”); *see also*, 36-39. It is well-settled that “a patent composed of several elements is not proved obvious merely by demonstrating,” as Petitioner tries to do here, “that each of its elements was, independently, known in the prior art.” *KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 418-419 (2007). Rather, Petitioner must “identify **a reason** that would have prompted a person of ordinary skill in the relevant field to combine the elements in the way the claimed new invention does.” *Id.* And this it has failed to do. As Dr. Rubin explains:

Petitioner’s proffered modifications do not seem to create any advantage for Orsini’s system and, instead, would seem to undermine Orsini’s approach. At the very least, Petitioner’s proffered modifications are inadequately developed. Instead, Petitioner’s

these references. *Edwards Lifesciences Corp. v. Aortic Innovations LLC*, IPR2023-01232, Paper 10, 36 (Feb. 7, 2024) (“Petitioner’s rationale to modify [reference] based on inchoate and **unasserted combinations** is also unpersuasive.”).

modifications ultimately seem to be driven by hindsight. To reiterate, in each instance, Petitioner begins with Orsini's Figure 21 which sets out a fully functional system (Ex. 1005 [Orsini] [0380]) that Petitioner has not improved upon. Adopting the solution of "performing an encryption operation based on the external key to further secure the plurality of data chunks" seems driven by hindsight, not any technological need or benefit that would improve upon Orsini. It seems the claim limitation itself is being used as a roadmap driving the modification.

Ex. 2033 [Rubin-2nd-Decl.] ¶ 81.

Petitioner's proffered modifications thus provide no benefit, entail significant drawbacks, and are precisely the sort of hindsight-driven analysis the Federal Circuit has rejected. *Ruiz v. A.B. Chance Co.*, 357 F.3d 1270, 1275 (Fed. Cir. 2004); ID, 36, 37, 38.

Petitioner's invocation of using a "group key" or encrypting the "distinct encryption keys" with a "key-encryption key" are likewise hindsight-driven, merely following the '854's inventive concept of encrypting the share encryption keys with a workgroup key. *Compare* Pet., 37-41 *with* Ex. 1001 ['854] 79:52-80:4. But "[t]he inventor's own path itself never leads to a conclusion of obviousness; that is hindsight." *Univ. of Strathclyde v. Clear-Vu Lighting LLC*, 17 F.4th 155, 165 (Fed. Cir. 2021); *InTouch Techs., Inc. v. VGo Commc'ns., Inc.*, 751 F.3d 1327, 1351 (Fed. Cir. 2014) (rejecting obviousness theory that "relied on

the [] patent itself as her roadmap for putting what she referred to as pieces of a ‘jigsaw puzzle’ together.”).

Petitioner, thus, contends a POSITA would be motivated to modify Orsini in a fashion that would yield no benefit identified in the prior art (in view of Orsini’s non-cryptographic, “comprehensive package for data security”), entails significant drawbacks, is based upon inchoate and nonasserted combinations, and is ultimately driven by using the inventor’s own path against him. As the Board found, Petitioner’s proffered modification is rank hindsight and should be rejected.

III. THE FOSTER-HAYHURST GROUND FAILS (ALL CLAIMS, GROUND 2).

Petitioner’s Foster/Hayhurst combination fails because Petitioner fails to prove that Hayhurst is analogous art or that a POSITA would be motivated to combine Foster/Hayhurst as proposed.

A. Petitioner Fails To Demonstrate That Hayhurst Is Analogous Art.

The Foster/Hayhurst ground fails because Petitioner fails to prove Hayhurst is analogous art. Petitioner’s combination relies upon “Hayhurst’s techniques for storing and distributing encrypted data segments.” Pet., 61. Hayhurst, however, is not analogous art to the ’854.

“A reference qualifies as prior art for an obviousness determination . . . only when it is analogous to the claimed invention.” *In re Klein*, 647 F.3d 1343, 1348

(Fed. Cir. 2011). An obviousness ground that relies upon non-analogous art fails, because a POSITA would have no reason to consider the non-analogous art, much less a motivation to combine it with another reference. *Id.*, 1352. Petitioner bears the burden of demonstrating a reference is analogous. *Sanofi-Aventis Deutschland GmbH v. Mylan Pharms. Inc.*, 66 F.4th 1373, 1377 (Fed. Cir. 2023).

To be analogous, a reference must be either (1) from the same field of endeavor as the patent-at-issue, or (2) reasonably pertinent to the particular problem with which the inventor of the challenged patent is involved. *In re Clay*, 966 F.2d 656, 658-59 (Fed. Cir. 1992). Petitioner fails to prove that Hayhurst is analogous under either prong.

1. Hayhurst And The '854 Patent Do Not Share The Same Field Of Endeavor.

Petitioner makes no attempt to show that Hayhurst and the '854 share the same field of endeavor. Nor could it. As Petitioner and Dr. Bhattacharjee admit, the '854's field is securing data while Hayhurst's is decentralized electronic media delivery. Pet., 4 (“[t]he '854 patent relates to techniques for securing electronic data from unauthorized access.”); Ex. 1003 [Bhattacharjee-Decl.] ¶ 33 (same); Pet., 59 (“Hayhurst concerns ‘**electronic media delivery, such as video-on-demand**’”); Ex. 1003 [Bhattacharjee-Decl.] ¶ 178 (same); Ex. 2034 [Bhattacharjee-Depo.] 101:19-25.

To discern the field of endeavor, the Federal Circuit has looked to various sources, including a patent’s title, claim preamble, abstract, “background of invention” and “field of the invention,” and the function and structure of the claimed inventions. *Airbus S.A.S. v. Firepass Corp.*, 941 F.3d 1374, 1380-81 (Fed. Cir. 2019) (determining field from title, claim preamble); *Kamstrup A/S v. Axioma Metering UAB*, 43 F.4th 1374, 1384 (Fed. Cir. 2022) (discerning field by comparing abstracts); *In re Bigio*, 381 F.3d 1320, 1326 (Fed. Cir. 2004) (“The Board ... correctly set the field ... by consulting the structure and function of the claimed invention ...”); *In re Wood*, 599 F.2d 1032, 1036 (CCPA 1979) (determining field from “Background of Invention”); *Wyers v. Master Lock Co.*, 616 F.3d 1231, 1237-38 (Fed. Cir. 2010) (similar); *Airbus*, 941 F.3d at 1381 n. 8 (“[T]he specification of each reference includes a ‘Field of the Invention’ section that distinctly describes each applicable field of endeavor.”).

As Dr. Rubin explains:

Each of these traditional sources for identifying a field of endeavor uniformly confirms that the ’854’s field is securing data.

The ’854’s Field Is “Securing Data”	
Title	“ secure data parser and network”
Abstract	“ secure data parser for “ securely storing and communicating data ”
Field of Invention	“ securing data from unauthorized access or use.” Ex. 1001 [’854] 1:25-26.

Background of the Invention	focuses on the need “for ensuring data stored and transmitted ... cannot be read or otherwise compromised ” and seeks “to provide a cryptographic system whose security is user-independent ...” <i>Id.</i> , 1:35-37, 2:42-44.
Summary of the Invention	“one aspect of the present invention is to provide a method for securing virtually any type of data from unauthorized access or use.” <i>Id.</i> , 2:45-47.
Claims	each claim concerns “securely storing a data set” or “securing a data set”

Ex. 2033 [Rubin-2nd-Decl.] ¶ 102.

Hayhurst does not share this field:

Hayhurst’s field, per these same traditional sources, is “decentralized media delivery.”

Hayhurst’s Field Is “Decentralized Media Delivery”	
Title	“Decentralized Media Delivery”
Abstract	“A decentralized approach to delivering a media file ...”
Technical Field	“This application relates to the field of electronic media delivery ...” <i>Id.</i> , [0002].
Background of the Invention	Focuses entirely upon delivering movie files. <i>Id.</i> , [0003]-[0004].
Summary of the Invention	“The present invention addresses the problem of prohibitive bandwidth requirements for a content provider in delivering video-on-demand or other large media files. ” <i>Id.</i> , [0005].
Specification/Figures	“FIG. 1 is a block diagram of a system adapted for carrying out the decentralized media delivery of the present invention.” <i>Id.</i> , [0012]

	“FIG. 4 is a flow diagram showing how the <i>decentralized media file delivery</i> of the present invention is carried out.” [0015]
Claims	each claim concerns “securely storing a data set” or “securing a data set”

Ex. 2033 [Rubin-2nd-Decl.] ¶ 103.

Petitioner may respond that Hayhurst includes disclosures about the security of the media files to which it is directed. But Federal Circuit precedent confirms that art is not analogous merely because it includes disclosures that tangentially relate to the field of another reference. Indeed, even art that is in the same general technological area as the subject patent may not be in the same field of endeavor. While Hayhurst mentions security, that is not Hayhurst’s field of endeavor as established by the traditional sources discussed above. Hayhurst does not teach, for example, that prior art systems did not adequately secure media files, or that its invention addresses a need for more security. Ex. 1007 [Hayhurst] [0003]-[0005]; Ex. 2033 [Rubin-2nd-Decl.] ¶ 104.

Federal Circuit precedent, moreover, confirms that even art with some overlapping disclosures and in the same technological area as the subject patent may not be in the same field of endeavor. In *Clay*, for example, the patent-at-issue and the allegedly analogous art (Sydansk) were two patents owned by the same company (Marathon Oil), both references related to gels used in the oil industry,

and the district court noted that the gels were “similar” in some ways. *Clay*, 966 F.2d at 657-60. But the Federal Circuit found that the Board clearly erred in finding Sydanski analogous because the patent-in-question’s “field of endeavor is the *storage* of refined liquid hydrocarbons” whereas “[t]he field of endeavor of Sydanski’s invention ... is the *extraction* of crude petroleum.” *Id.*, 659 (italicization in original). Thus, the court held, the “storage” and “extraction” of oil are two separate fields of endeavor.

Similarly, not all references that relate in some way to “semiconductor memory” are analogous:

The Allen-Bradley art is not in the same field of endeavor as the claimed subject matter merely because it relates to memories. It involves memory circuits in which modules of varying sizes may be added or replaced; in contrast, the subject patents teach compact modular memories.

Wang Labs., Inc. v. Toshiba Corp., 993 F.2d 858, 864-65 (Fed. Cir. 1993).

Hayhurst’s field is decentralized delivery of media files. That it also mentions the “secure handling and storage” of these files, does not alter or expand its field from decentralized electronic media delivery to also encompass the “securing data” field of the ’854.

2. Hayhurst Is Not Reasonably Pertinent To The Problem The Inventors Of The '854 Patent Were Addressing.

The Federal Circuit has made clear that “the purposes of both the invention and the prior art are important in determining whether the reference is reasonably pertinent to the problem the invention attempts to solve.” *Clay*, 966 F.2d at 659. Thus, “when addressing whether a reference is analogous art with respect to a claimed invention under a reasonable-pertinence theory, the problems to which both relate must be identified and compared.” *Donner Tech., LLC v. Pro Stage Gear, LLC*, 979 F.3d 1353, 1359 (Fed. Cir. 2020). If the reference “is directed to a different purpose, the inventor would accordingly have had less motivation or occasion to consider it.” *Clay*, 966 F.2d at 659.

Here, Petitioner does not attempt to meet its burden of establishing that Hayhurst is reasonably pertinent because the Petition does not identify the problems Hayhurst’s or the ’854, respectively, are directed to, let alone compare them. A proper analysis shows that Hayhurst is not reasonably pertinent to the problem the ’854’s inventors were trying to solve. As Dr. Rubin explains:

The ’854 is directed towards creating a cryptographic system whose security is user independent, while still supporting mobile users. Ex. 1001 [’854] 2:42-44. The ’854 notes the “ever-increasing need ... for ensuring data stored and transmitted” to not “be read or otherwise compromised.” *Id.*, 1:30-37. The ’854 discusses certain prior art methods of securing data and the ways in which those methods may be

compromised because they rely upon users. *Id.*, 1:38-44 (deficiencies in login/password systems); 1:46-2:14 (deficiencies in public-key systems that are “highly reliant on the user for security”); 2:15-38 (deficiencies in biometric systems that “suffer[] from a variety of drawbacks” from user losing/breaking smartcard). Consequently, the ’854 addresses the need “to provide a cryptographic system whose security is user-independent while still supporting mobile users.” *Id.*, 1:35-37, 2:42-44.

Ex. 2033 [Rubin-2nd-Decl.] ¶ 106.

Hayhurst, by contrast, is directed towards “address[ing] the prohibitive bandwidth requirements for a content provider associated with delivering video-on-demand other large media files.” Ex. 1007 [Hayhurst] [0005]. As Hayhurst explains:

It is usually not economically feasible for the operator of, for example, a cable television (CATV) system (or other content provider) to provide for delivery-on-demand a large number of movie selections to subscribers ***because such a system requires a extensive amount of bandwidth between the content provider and each of the subscriber units*** (televisions or other display devices) of the network.

Id., [0004]. In order to solve this problem, Hayhurst teaches “a system that shifts the bandwidth requirements away from the content provider and to the network of subscriber units.” *Id.*; Ex. 2033 [Rubin-2nd-Decl.] ¶ 107. Petitioner’s expert agrees that Hayhurst is directed towards the “problem” of extensive bandwidth associated

with providing “delivery-on-demand” media content. Ex. 1003 [Bhattacharjee-Decl.] ¶ 178; Ex. 2034 [Bhattacharjee-Depo.] 87:7-13; 92:1-11.

Thus, Hayhurst and the '854 are directed towards addressing entirely different problems. As Dr. Rubin explains:

The '854, again, addresses eliminating user vulnerabilities in storing sensitive information. Hayhurst, by contrast, addresses the prohibitive bandwidth requirements for video-on-demand from the perspective of the content provider. Ex. 1007 [Hayhurst] [0003]-[0005].

Security from piracy is merely an ancillary benefit to Hayhurst's solution of decentralizing content delivery. And, what little Hayhurst has to say about encryption does not suggest that it is directed towards securing data in a manner that is “user-independent.” Rather, in Hayhurst's system, a subscriber requests both an encrypted movie and an encryption key. Ex. 1007 [Hayhurst] [0028], [0030]. While the movie is originally dispersed amongst thousands of subscribers, it is reassembled, along with the key, by a single user. Thus, the security of the data (the movie) is user-dependent.

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 108-109.

Thus, Hayhurst is not reasonably pertinent to the '854 because they are directed to different problems; a POSITA would not have expected to solve a problem with securing data sets in a user-independent fashion by considering a reference directed to reducing the bandwidth required to transmit video-on-

demand. The mere fact that Hayhurst mentions the security of media files does not change this analysis as even technologically similar references may not be reasonably pertinent. *See, e.g., Clay*, 966 F.2d at 660 (district court finding of reasonable pertinence was “clearly erroneous” where a POSITA “would not reasonably have expected to solve the problem of dead volume in tanks for storing refined petroleum by considering a reference dealing with plugging underground formation anomalies.”); *Wang*, 993 F.2d at 864 (prior art memory modules not reasonably pertinent to patented memory modules); *Polygroup Ltd. v. Willis Elec. Co., Ltd.*, IPR2016-01610, Paper 187, 31, 35-40 (Feb. 26, 2018) (reference not reasonably pertinent despite including “structurally and functionally similar” electrical connectors).

In sum Petitioner’s second ground fails because it cannot meet its burden of showing that Hayhurst analogous art under either the “same field of endeavor” or “same problem” tests.

B. Petitioner Fails To Demonstrate That A POSITA Would Be Motivated To Combine Foster And Hayhurst (All Claims, Ground 2).

Petitioner’s second ground should also be rejected because a POSITA would not be motivated to combine the Foster and Hayhurst references, which are incompatible and have conflicting goals. Foster is directed towards avoiding delays that interfere with random access to content (*e.g.*, when a user changes a

channel). See Section III.B.1. Hayhurst, by contrast, is directed to reducing the bandwidth required by a content provider in a video-on-demand context, where immediate access is unnecessary and, in fact, impossible. Hayhurst's method shifts the bandwidth burden from the content provider but does so in a fashion that introduces, rather than avoids, delays. A POSITA would not be motivated to combine Hayhurst's decentralized content storage into Foster because doing so would frustrate Foster's goal of avoiding delays, render Foster unsuitable for streaming live television, and make Foster's system less secure. Fundamentally, Petitioner's combination would make Foster's system worse for no discernible benefit.

1. Foster Provides Synchronized Transmission To Avoid Delays.

Foster's system is directed towards synchronizing the transmission of encrypted content packets with the keys used to encrypt those packets, so that each packet can be viewed as soon as it is received. This fundamental goal is reflected in Foster's title, abstract, field of the invention, background of the invention, summary of the invention, and its disclosure generally. Ex. 1006 [Foster] Title; Abstract, [0002], [0008]. Both experts agree that Foster is directed towards the synchronized transmission of encrypted content with corresponding encryption keys. Ex. 2033 [Rubin-2nd-Decl.] ¶ 114; Ex. 2034 [Bhattacharjee-Depo.] 16:8-12. Indeed, at his deposition Dr. Bhattacharjee offered his opinion that "synchronous

transmission to a receiver [to] foster random access to content” was “sort of the whole invention in Foster.” Ex. 2034 [Bhattacharjee-Depo.] 32:20-23.

Foster’s synchronization is directed towards reducing delays in viewer access to content as it is transmitted by a provider. Per Dr. Rubin:

Foster seeks to improve upon prior art systems that were susceptible to “loss of synchronization between the content and the keys.” *Id.*, [0006]. The lack of synchronization impacted user experience with “inherent latencies” and “delays,” particularly when “providing random access of protected content,” *e.g.*, when the user changes the channel and when the image the new image displays. *Id.*, [0006], [0052] (“because delivery of keys and protected content are non-synchronized in a CAS, certain latencies such as delays are inherent in attempting to gain random access to the content.”); [0056] (without synchronization, “a delay could be incurred between when encrypted content is received and when it [is] accessed (*e.g.*, between when a channel is changed and when an image is displayed).”); [0063] (similar); [0064] (“both DRMS and CAS have inherent latencies (*e.g.*, delays) in providing random access [to] content.”). Foster, moreover, indicates that “synchronous transmission to ... a receiver fosters random access to the content.” *Id.*, [0007]; [0052].

Ex. 2033 [Rubin-2nd-Decl.] ¶ 117.

To achieve synchronization, and avoid delays, Foster processes content into “content unit[s]” consisting of an encrypted content packet and the “title key” used

to encrypt the packet. The title key is itself encrypted with a “key-encrypting-key,” or “KEK,” before it is combined with the content packet in a single content unit. *Id.*, [0008]. As illustrated in Figure 16 below—which Petitioner consistently relies upon (Pet., 58, 64)—Foster’s system transmits content units **directly** to

recipient 430:

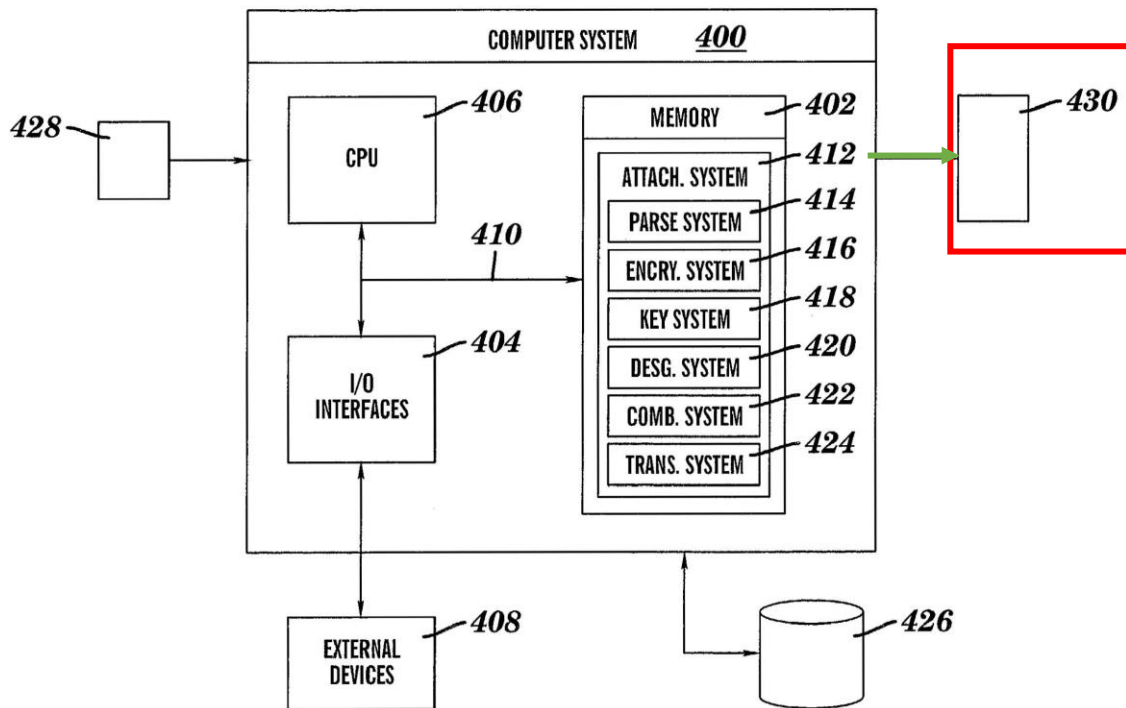


FIG. 16

Ex. 1006 [Foster] Fig. 16; [0078] (“computer system 400 ... process[es] content units ... and transmit[s] the processed content units to a recipient 430.”). Because each content unit contains both the content and the key, each content unit may be

encrypted and transmitted by the source as it is generated, and then decrypted and viewed without delay. *Id.*, [0064]; Ex. 2033 [Rubin-2nd-Decl.] ¶ 118.

Foster teaches that its system processes and transmits content units between a “Content Service Provider” “such as a cable service provider” (*id.*, [0039]) and a “Receiver” “such as a set-top box” (*id.*, [0040]). *Id.*, Fig. 16, [0078]. It seeks to improve upon prior art cable systems providing real-time television. *Id.*, [0052] (Foster is directed towards “Conditional Access System[s] (CAS),” which are “cable or satellite television system[s]”); [0006]. As Dr. Rubin explains, Foster’s innovations are primarily useful in the streaming context, where the individual content units are the serial components of a real-time content stream:

Foster’s teachings of synchronized transmission to reduce delays/latencies are particularly significant in a streaming context contemplated by Foster’s Figure 8, which Petitioner relies upon (*e.g.*, Pet., 56, 65-66). “In streaming, multimedia data is transferred in a stream of packets that are interpreted and rendered in real time by a software application, as the packets arrive in the user’s terminal.” Ex. 2035 [Patrikakis] 220. Because packets are transmitted as a stream and rendered in real-time, any delays have an especially noticeable impact. When watching, *e.g.*, a live sporting event, any delay in receiving, decrypting and/or rendering each content unit in a stream would be a significant drawback. Figure 8 depicts the “Encryption of [an] Elementary Stream,” provided to a set-top box which has “random

access points (points in stream where player can start [displaying the stream]))”:

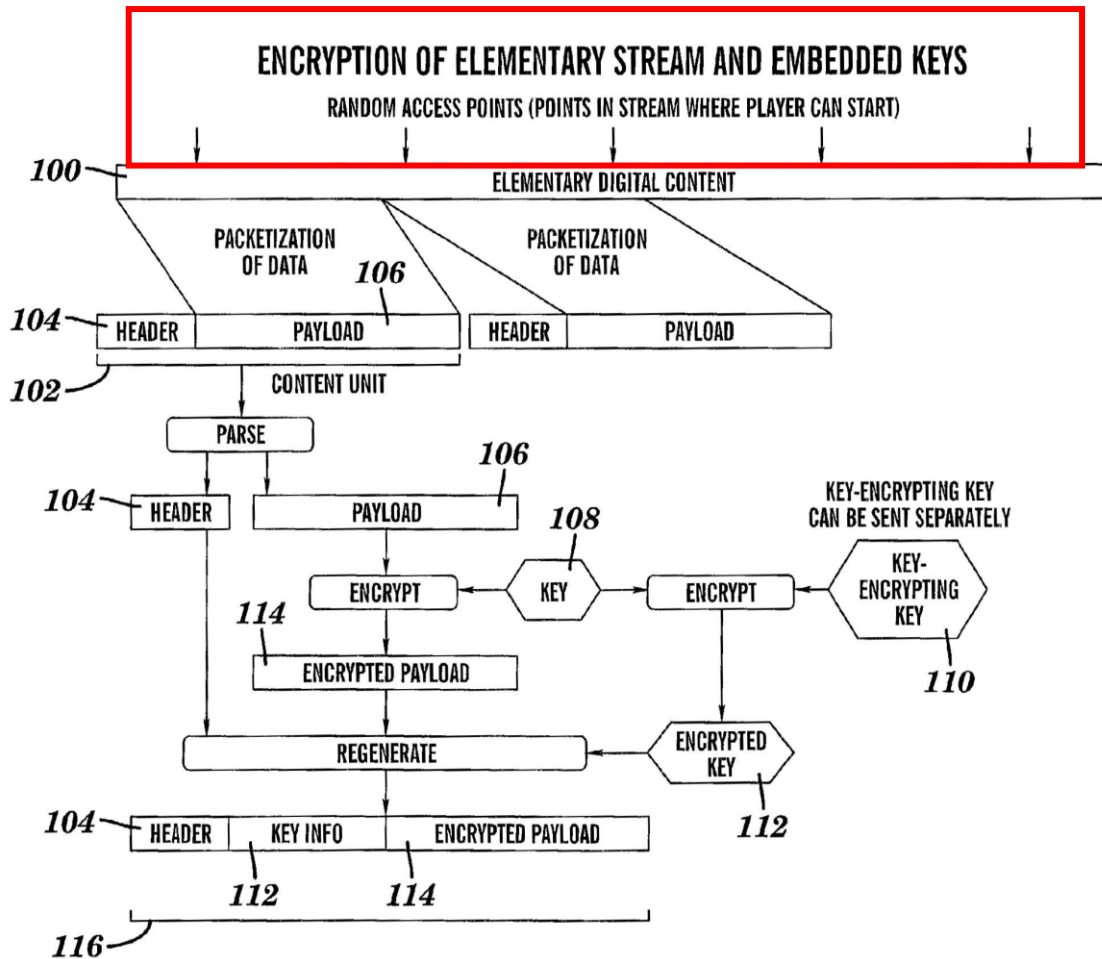


FIG. 8

Ex. 1006 [Foster] Fig. 8 (annotated); *see also* [0064]. Foster makes clear that when content and keys are not sent synchronously, the type of random access of content contemplated by Figure 8 “is not possible” in the streaming context. *Id.*, [0063]; *see also* [0064], [0071]. This is

because without synchronization, the resulting delays would prevent streaming.

Ex. 2033 [Rubin-2nd-Decl.] ¶ 120.

2. Hayhurst's System Does Not Allow For Real-Time Streaming Because Its System Requires Delays.

Hayhurst, by contrast, is not concerned with synchronous transmission, avoiding delays, or facilitating real-time streaming. Hayhurst purports to facilitate the transmission and playback of content *files*, not content *streams*. Specifically, Hayhurst aims to overcome “prohibitive bandwidth requirements for a content provider in delivering video-on-demand or other large media files” to a subscriber network. Ex. 1007 [Hayhurst] [0005]. Hayhurst teaches that, in light of the size of these files and the bandwidth required to transmit them, it is “usually not economically feasible” for a content provider “to provide for delivery-on-demand [of] a large number of movie selections to subscribers.” *Id.*, [0004]; Section III.A.2.

To address the bandwidth problem, Hayhurst “shifts the bandwidth requirements away from the content provider [] to the network of subscriber units.”

Id. As Dr. Rubin explains:

Hayhurst divides a movie file into segments that are then stored across a network of subscriber units. When a movie is requested by one subscriber, the system directs other subscribers to send their respective

segments of the file to the requesting subscriber. When all of the segments are received at the requester's computer, they are then decrypted and the file is reassembled from the collection of segments for viewing. Ex. 1007 [Hayhurst] [0006]. Hayhurst's system is a network of "thousands of subscribers." *Id.*, [0016]; Ex. 2034 [Bhattacharjee-Depo.] 101:15-25.

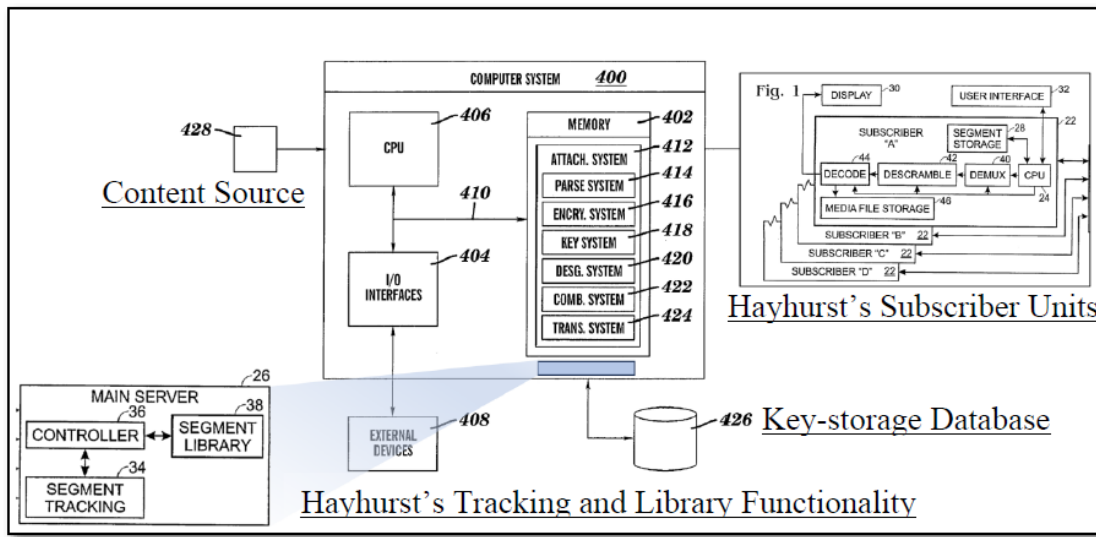
Hayhurst's methodology is not compatible with real-time streaming or random-access viewing of live television. Hayhurst requires that an entire movie file be segmented, and the segments transmitted to, and stored among, the members of a subscriber network before the movie can even be requested for viewing by any individual subscriber. Beyond that, because the segments are not stored in or received in any particular order, the movie cannot be viewed at all until (i) **all** segments are received from the subscriber network by the subscriber, (ii) the requesting unit obtains authorization, and (iii) the packets are descrambled and then reassembled into the complete movie file. Ex. 1007 [Hayhurst] [0029]; [0030]; Ex. 2034 [Bhattacharjee-Depo.] 101:19-25. Thus, Hayhurst's system requires all segments to be received and for the entire file to be reassembled before the system can play the movie, thus, "excluding the use for covering live events."

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 123-124 (quoting Ex. 2035 [Patrikakis] 219). As others in the field have recognized, in "download and play" systems such as Hayhurst, "the client has to download and store first ***the whole media content*** on his PC and only then he/she will be able to view and use it." Ex. 2035 [Patrikakis]

219. This requirement “*exclud[es] the use [of such systems] for covering live events*”, as the multimedia information that is to be transmitted cannot be ‘a-priori’ available.” *Id.*

3. Petitioner’s Combination Involves Storing And Distributing All Of Foster’s Content Units Per Hayhurst’s Methodology Which Precludes Real-Time Streaming.

Petitioner posits a combination of Foster and Hayhurst in which Foster’s media **content units** are distributed and stored in Hayhurst’s decentralized media **file** storage system. Specifically, “Foster’s encrypted content units” would be stored “on multiple storage devices ... based on Hayhurst’s techniques for storing and distributing encrypted data segments” of a media file. Pet., 61. In Petitioner’s “exemplary implementation of Foster-Hayhurst,” illustrated below, “Foster’s ‘computer system 400’ is augmented to ... control distribution of and access to Foster’s encrypted content portions on [Hayhurst’s] ‘subscriber units.’” *Id.*, 64. Thus, in Petitioner’s combination, Foster’s “content stream 100” “is parceled into content units 102” (*id.*, 56) and rather than sending content units directly to the requesting subscriber, the content units would be distributed amongst Hayhurst’s network of thousands of subscribers.



Foster-Hayhurst (exemplary implementation)

Id., 64; 61 (“POSAs would have recognized several benefits in storing Foster’s encrypted content units distributed on different storage devices as Hayhurst teaches.”); 62 (similar); 63 (similar). At his deposition, Dr. Bhattacharjee admitted that this is the only implementation of the combination that is described or relied on in his declaration. Ex. 2034 [Bhattacharjee-Depo.] 126:3-9.³ As discussed in the next section, Petitioner fails to show a POSITA would be motivated to make this combination.

³ While Dr. Bhattacharjee was referring to the Foster-Hayhurst ground in IPR2024-00212, that ground is substantively identical to present ground, including the sole “exemplary implementation.”

4. A POSITA Would Not Be Motivated To Combine Foster And Hayhurst As Proposed.

In evaluating a POSITA's motivation to make a combination, "[t]he Board must weigh the benefits and drawbacks of the modification against each other, to determine whether there would be a motivation to combine." *Arctic Cat Inc. v. Polaris Indus.*, 795 Fed. App'x. 827, 833 (Fed. Cir. 2019); *see also* Section II.C.2 (additional citations). Critically here, the Federal Circuit has held that a POSITA would not be motivated to render a reference "inoperable for its intended purpose" or to change its "principle of operation." *Plas-Pak Indus. v. Sulzer Mixpac AG*, 600 Fed. App'x 755, 759 (Fed. Cir. 2015) (POSITA not motivated to make reference "inoperable for its intended purpose"); *In re Gordon*, 733 F.2d 900, 902 (Fed. Cir. 1984); *In re ICON Health & Fitness, Inc.*, 496 F.3d 1374, 1382 (Fed. Cir. 2007); *Tec Air, Inc. v. Denso Mfg. Mich., Inc.*, 192 F.3d 1353, 1360 (Fed. Cir. 1999); *In re Ratti*, 270 F.2d 810, 813 (CCPA 1959) (not motivated to change reference's principles of operation); *Plas-Pak*, 600 F. App'x. at 759 ("a change in a reference's 'principle of operation' is unlikely to motivate a person of ordinary skill to pursue a combination with that reference").

Here, far from providing any significant benefit, the proposed combination would thwart Foster's goals, make Foster inoperable for its intended purpose, and compromise any security benefits otherwise provided by Foster alone.

Petitioner's combination suffers from three critical flaws. First, the combination would introduce the specific delays Foster is designed to avoid, making the synchronization that is the primary purpose of Foster unnecessary. *See* Section III.B.4(a). Second, the combination would not be capable of providing streaming (*e.g.*, real-time television) or random access to content, contravening Foster's intended purpose. *See* Section III.B.4(b). Third, Petitioner's combination would reduce security as compared to Foster by exposing Foster's key-encryption keys. *See* Section III.B.4(c). The Petition does not address any of these issues. *See* Section III.B.4(d).

- a. *Adding Hayhurst's Delivery Method To Foster's System Exacerbates The Underlying Problem Of Delay Foster Seeks To Solve.*

As explained, in Foster's system, because each packet in an elementary content stream can be generated as the content is created, and then sent directly to the recipient with its title encryption key, each packet can be viewed without delay, allowing for immediate, random access to the content. *See* Section III.B.1; *see also* Ex. 1006 [Foster] [0008], [0064].

Replacing Foster's direct distribution of content streams with Hayhurst's decentralized delivery and storage of media files would introduce the delays it was Foster's goal to avoid. Hayhurst, again, concerns a decentralized delivery system where a movie file is divided up into segments stored amongst thousands of

subscribers. Hayhurst's system mandates significant delays before content can be played. Section III.B.2.

A POSITA would not be motivated to combine Foster and Hayhurst as proposed. As Dr. Rubin explains:

Petitioner's Foster/Hayhurst combination would result in inevitable, significant, and crippling delays, the very thing Foster was intended to avoid. Petitioner's proposed combination of Foster and Hayhurst relies upon Hayhurst's methods of storing and delivering segments of media files. Pet., 61. In Petitioner's proposed combination, all of Foster's encrypted content units would, rather than be sent directly to the requestor as in Foster, instead be distributed across thousands of subscriber units before they are then, when requested, sent to a requesting unit, as taught in Hayhurst. Ex. 1007 [Hayhurst] [0016].

This introduces delay in a number of ways. First, in Hayhurst the content must exist as a discrete file before it can be divided into segments and distributed for storage among the subscriber network. (This is in stark contrast to Foster, where the content units can be created and transmitted as the stream of content is generated.) Then, in Hayhurst, the main server, upon receiving a request, must direct the thousands of subscribers to send segments to the requestor. *Id.*, [0028] ("after the main server receives a movie request 104 from a requesting subscriber ..., the server responds 106 by sending over the network instructions to the network subscriber units to direct all of the segments

of the requested movie to the requesting subscriber.”). Should there be one or more missing packets, which there inevitably would be for subscribers whose systems are turned off, internet is down, are no longer subscribers, etc., the main server must either direct another subscriber with the same packet to resend it or redistribute the packet. *See id.*, [0025] (“Each segment [] is copied and distributed to a plurality of different subscribers to provide for redundant availability of each segment in the event a subscriber’s segment storage 28 becomes inaccessible. As noted, the main server’s segment-tracking module 34 can ...facilitate replacement or redistribution of segments when necessary.”); Ex. 2034 [Bhattacharjee-Depo.] 106:17-109:6. Accordingly, the requesting unit must wait until “***all*** packets are received” before sending a request to the main server to decrypt the segments and reassemble the movie file. *Id.*, [0029] (“The subscriber unit CPU 24 and associated buffers (not shown) log the received packets and, once ***all packets*** are received, sends a notice to the main server 26 that ***all packets*** now reside in the unit of the requesting subscriber.”). Only once all packets are on the requesting unit does the main server “transmit[] authorization.” *Id.* After obtaining authorization, the receiving unit may descramble, process, and reassemble the entire media file for playback. *Id.*, [0029]-[0030]; Ex. 2034 [Bhattacharjee-Depo.] 109:8-111:1. Only after all of these steps are completed, the movie can be viewed.

Consequently, Hayhurst’s methods of storing and delivering content make playback delays—the underlying problem Foster seeks to address—exponentially worse. Rather than immediately streaming

content in packets that are synchronized with the encryption keys as Foster does, Petitioner would have Foster's system store and deliver content using Hayhurst's decentralized, time-consuming method. In other words, Petitioner proposes changing Foster's method which has been designed to reduce delays to one that prioritizes bandwidth at the expense of creating delays.

In my opinion, a POSITA would not be motivated to change Foster to implement methods which greatly exacerbate the underlying problem Foster seeks to address.

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 127-130.

b. *Hayhurst's Delivery Method Is Incompatible With Foster's System.*

The principal application for Foster's method is real-time cable television broadcasting to a set-top box. As Dr. Rubin explains, Foster seeks to securely stream content, *e.g.*, live television, without the delay found in previous systems:

Foster's system serves as an intermediary between "source 428" and "recipient 430," where the "source" includes "a cable service provider" (Ex. 1006, [0039], [0042]) and the "recipient" includes a "set-top box" (*id.*, [0040], [0041]. As Dr. Bhattacharjee agrees, Foster specifically seeks to improve upon the "Conditional Access System (CAS)," "such as a cable or satellite television system." *See id.*, [0052]; Ex. 2034 [Bhattacharjee-Depo.] 41:10-15, 30:8-10. Consequently, Foster's principle of operation concerns delivering live cable television to a set-top box. Ex. 2039 [Marlin-DRM] 1 (CAS used for OTT digital

video streaming service, “such as live sports”); Ex. 2040 [Verimatrix] 2 (CAS used for “content [transmitted] as a signal, which virtually anyone can pick up via satellite, terrestrial, or cable network.”); see Ex. 1006 [Foster] [0056] (Foster seeks to avoid delays “when a channel is changed”); see also Ex. 2034 [Bhattacharjee-Depo.] 37:11-39:9.

Foster’s system is able to deliver live or real-time content as required for cable television because Foster’s methods do not require a user to have an entire media file before it can view content. Rather, Foster’s system allows viewers to receive and process packets in real-time as they are generated and sent by the content provider. Ex. 1006 [Foster] [0065] (“the teachings of the present invention can be applied to the scenario where the content is received as content units”); Ex. 2035 [Patrikakis] 220 (“In streaming, multimedia data is transferred in a stream of packets that are *interpreted and rendered in real time* by a software application, *as the packets arrive in the user’s terminal.*” As a result, Foster avoids “delay[s]” “between when encrypted content is received and when it [is] accessed (e.g., between when a channel is changed and when an image is displayed).” *Id.*, [0056].

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 131-132.

Petitioner contends that “POSAs had reason to store Foster’s encrypted content units on multiple storage devices (*e.g.*, consumer set-top boxes like Hayhurst’s subscriber units and Foster’s recipients 430)” among the various subscribers in Hayhurst’s subscriber network. Pet., 61. But doing so would render the combination inoperable for viewing steamed content, Foster’s intended

purpose, because Hayhurst requires that all segments be delivered to a requesting unit before the content can be reassembled and playback can begin. As Dr. Rubin explains:

Hayhurst is directed toward delivering video-on-demand and not live television. In particular, Hayhurst's focus is upon delivering movies—*i.e.*, an already defined media file—in a fashion that avoids bandwidth limitations. Ex. 1007 [Hayhurst] [0003]-[0004]. As such, Hayhurst utilizes a decentralized delivery system in which segments of a movie are stored on set-top boxes throughout Hayhurst's system until one set-top box requests the entire movie. *Id.*, Abstract, [0006], [0028]. When a requesting set top box requests a movie, it cannot be played immediately. Rather, viewing can only begin after all segments are received, because “the segments are separate in the sense that they represent discontinuous or non-sequential portions of the movie and must be soiled and integrated with the remaining segments (distributed elsewhere in the network) to provide the complete movie.” *Id.*, [0021]; Ex. 2034 [Bhattacharjee-Depo.] 104:2-13. There is no indication in Hayhurst that the segments arrive in order to the requesting unit. Nor would it be feasible for Hayhurst's main server to facilitate such a thing, as there would be significant delays incurred as the main server would have to ensure successfully receipt of each segment before directing the next subscribing to send to following segment. Thus, only after all segments are received from the different members of the subscriber network can authorization to decode them be sought and obtained, and the segments are assembled and processed. *Id.*, [0028]-[0030].

Ex. 2033 [Rubin-2nd-Decl.] ¶ 133.

Because all segments must be received before the content can be reassembled and displayed, Foster/Hayhurst would not be able to broadcast live television. As Dr. Rubin explains:

In the Foster/Hayhurst combination, a set-top box requesting cable television content would have to wait to receive all of the packets sent from the set-top boxes of a multitude of other subscribers before seeking and obtaining authorization and before reassembling and processing the entire content file. This wait-and-see approach would not work when trying to watch live television. Ex. 1006 [Foster] [0063].

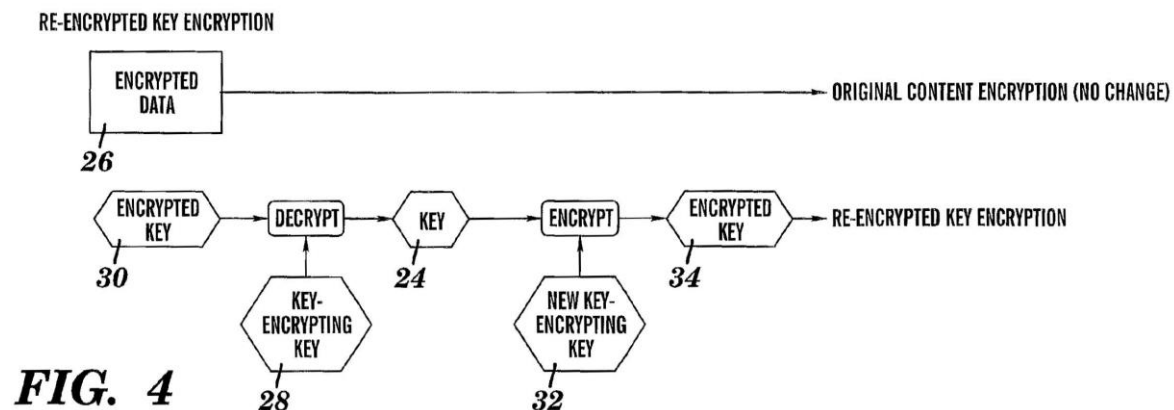
Ex. 2033 [Rubin-2nd-Decl.] ¶ 134.

c. *Hayhurst's Methods Make Foster's System Less Secure.*

Foster's content units are encrypted with title keys which are, in turn, encrypted with a key-encrypting-key ("KEK"). Pet., 64. In Petitioner's proposed Foster/Hayhurst combination, the encrypted content units would be, as in Hayhurst, distributed across thousands of subscribers. *Id.* A POSITA would not be motivated to do this because it would compromise the security of the system, as compared to Foster, by making Foster's KEKs more vulnerable while eliminating the ability (disclosed in Foster) to change the KEK as necessary.

Because Foster teaches attaching the title key used to encrypt content to the content unit, the title key itself must also be encrypted with a key-encrypting-key. See Ex. 1006 [Foster] [0053]. Critically, however, the key-encrypting-key may be replaced if, *e.g.*, it is compromised. As Dr. Rubin explains:

In Foster, should a key-encrypting-key be compromised or “protection on the content [need] to be varied,” “an encrypted title key 30 can be re-encrypted with a different key encrypting key 32 without having to re-encrypt the encrypted content 26.” Ex. 1006 [Foster] [0054].



Id., Fig. 4. Thus, Foster contemplates seamlessly switching the KEK for security purposes without re-encrypting the underlying content. Thus, if a key-encrypting-key is compromised, or a switch is otherwise desirable, a new KEK may be utilized going forward.

Moreover, Foster allows for a key-encrypting-key to be unique to each recipient. Foster teaches that “key encrypting key 28 is generally determined based on a key management block (KMB), which

is typically delivered to a recipient concurrently with or prior to the delivery of encrypted content 26.” *Id.*, [0054]. Because each recipient receives a KMB from which a key-encrypting-key can be determined, that key-encrypting-key can be unique to each recipient. This further increases the security of Foster’s content as the damage from any compromise of a recipient’s key-encrypting-key can be limited to that recipient and further prevented by simply changing that recipient’s key-encrypting-key.

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 135-136.

Adding Hayhurst’s method of storing and distributing content to Foster’s system, however, both increases the likelihood that KEKs become compromised and interferes with Foster’s ability to swap out a KEK as necessary. As Dr. Rubin explains:

Foster/Hayhurst uses Hayhurst’s method of storing and distributing content. Consequently, any subscriber (of thousands) is able to request content that has been divided into segments across thousands of subscribers. Because any one of Hayhurst’s subscribers can request the content, which has already been segmented and divided in advance, in any combination, Foster’s content units would all have to be secured using the same KEK(s), regardless of which subscriber will eventually request the content.

This, however, makes the combined system less secure for two reasons. First, it makes the KEKs more vulnerable. Because the KEK(s) is/are distributed amongst thousands of subscribers, there are

thousands of potential points of failure. And, if one set-top-box (of thousands) with the KEK(s) gets compromised, all of the content is compromised because all of the content is encrypted using the same KEK(s) and title keys. Second, it would be more complicated to switch KEKs than in Foster alone as all of the content units are distributed across the subscribers instead of streamed in real time as in Foster. Thus, adding Hayhurst's methods to Foster's system would make Foster's system less secure in contravention of Foster's aims. Ex. 1006 [Foster] [0004] ("In transmitting such content, however, the security thereof must be ensured.").

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 137-138.

d. *Petitioner Fails To Prove That The Alleged Benefits Outweigh The Drawbacks Of The Proposed Combination.*

As Dr. Rubin explains, the drawbacks to Petitioner's proposed combination outweigh any alleged benefits:

Petitioner argues that a POSITA would be motivated to combine Foster and Hayhurst as proposed because Hayhurst's methods would reduce bandwidth requirements, provide more security, and support redundancy. Pet., 61-62. Petitioner also argues that Hayhurst's methods result in aggregate speed and efficiency/intelligence. *Id.*, 62.

Foster's system, however, has no need for any of these alleged benefits. Foster's system simply processes content and transmits streamed content to a particular recipient. Ex. 1006 [Foster] Fig. 16; [0078]. Because Foster's system focuses upon streamed content (or

content stored on a DVD) and provides a one-to-one transmission from Foster's system to a particular recipient, there is no indication that Foster's system suffers from the bandwidth constraints confronted by Hayhurst's video-on-demand system. Foster's system, moreover, has sufficient security at the source, security which would be undermined in the combination, as discussed previously. Further, Foster's system has no need for redundancy as Foster's system can simply resend dropped packets, nor any need for efficiency/intelligence when sending on a one-to-one basis.

Moreover, any benefits are easily outweighed by the drawbacks. Again, Foster is directed towards reducing delays, particularly delays frustrating random access to content. The Foster/Hayhurst combination, however, would introduce enormous delays preventing random access to content and rendering Foster's system unsuitable for streaming live television. Further, the combination would also decrease Foster's security by exposing KEKs and hindering Foster's ability to replace a KEK when desirable.

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 139-141.

Petitioner, however, has failed to even consider, much less weigh, these manifold drawbacks against these purported benefits. For this additional reason, Petitioner fails to demonstrate that a POSITA would be motivated to combine Foster/Hayhurst as proposed.

C. Petitioner Fails To Demonstrate That Foster-Hayhurst Teaches “Generating A Plurality Of Data Chunks Based On The Data Set” By “Distributing The Data Set Into A Plurality Of Shares” (All Claims, Ground 2).

Even if combined, petitioner’s references fail to disclose all the claim elements. Limitation 1[b] requires “generating a plurality of data chunks based on the data set” and limitation 1[b1] requires “wherein generating the data chunk comprises[] distributing the data set into a plurality of shares.” Petitioner maps the claimed “plurality of shares” to Foster’s content units (more specifically, MPEG packet payloads), the claimed “plurality of data chunks” to Foster’s encrypted content units or “content packets,” and the claimed “data set” to Foster’s content.⁴ Pet., 72-73. Petitioner, however, fails to show that Foster’s “payloads” are “shares” or that its “encrypted content units” are “data chunks.”

Petitioner asserts with no further explanation that Foster’s “payloads” are equivalent to “shares” and “encrypted payloads” “are” “data chunks”:

As . . . illustrated in Figure 8 . . . , the payloads ([1B1]’s “shares”) are encrypted to generate encrypted payloads, which are “a plurality of data chunks based on the data set,” as claimed.

⁴ Petitioner does not contend it would be obvious to modify Foster alone or to combine Foster with Hayhurst to satisfy this limitation. Pet., 74.

Pet., 72, 73 (“all of the encrypted content packets (‘data chunks’) are required”); 74 (“As illustrated in Foster’s Figure 8 ..., Foster ‘parcel[s]’ (‘distribut[es]’) the elementary content stream (‘data set’) into a plurality of ‘payloads’ (‘shares’). [0064].”). As Dr. Rubin explains:

Petitioner and Dr. Bhattacharjee rely on Foster’s “payloads” as the claimed “data shares” and the encrypted payloads as the claimed “data chunks.” Pet., 72-74. Foster teaches that these “payloads” refer to the payloads of MPEG Packetized Elementary Stream (PES) packets (also referred to in Foster as “units”), as illustrated below in Figure 13:

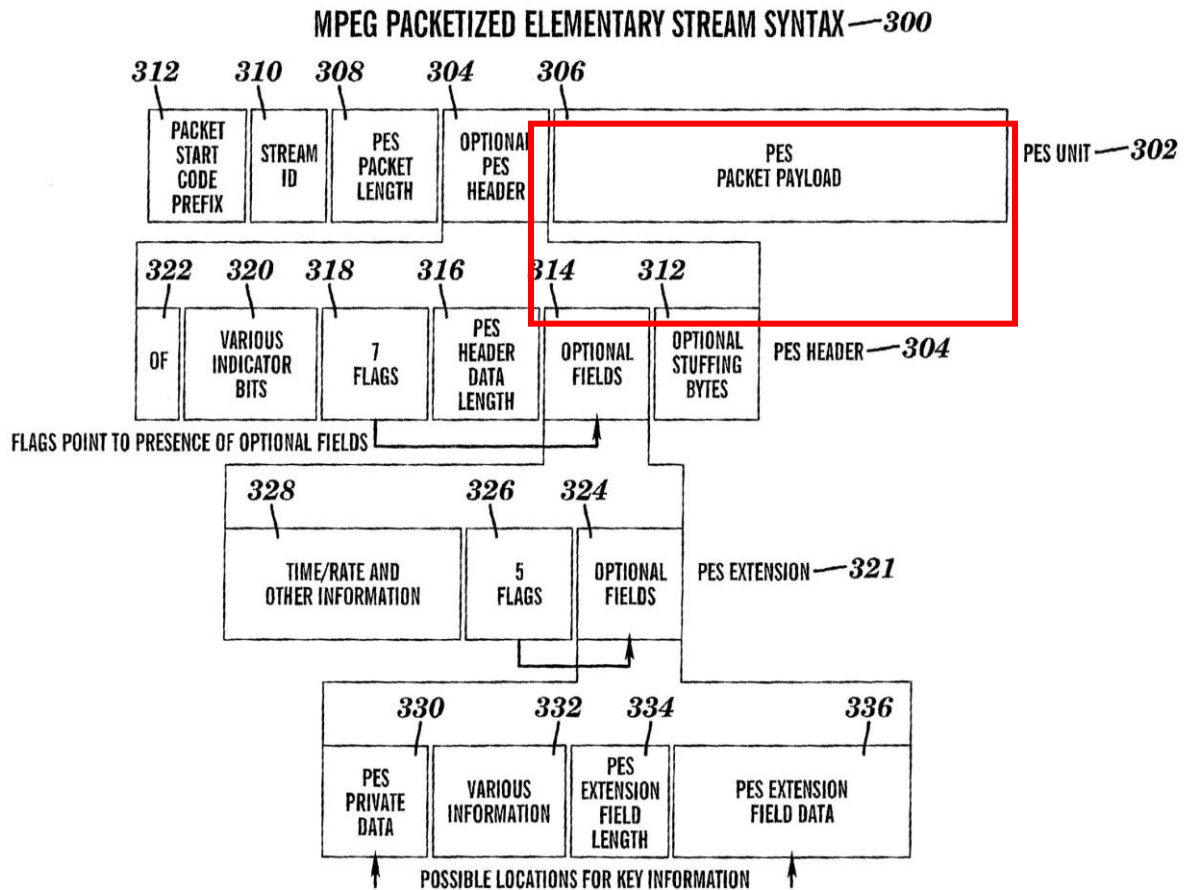


FIG. 13

Ex. 1006 [Foster] Fig. 13, [0072] (“a MPEG packetized elementary stream 300 is parceled into PES (packet) units 302 PES unit 302 includes packetized elementary stream (PES) packet payload 306” which “will be encrypted with a title key”). Foster’s PES packets are also referred to as “data packets.” See Ex. 2011 [Tsai] [0001], cl. 1. Consequently, Petitioner and Dr. Bhattacharjee map the claimed “data shares” to Foster’s packets (specifically, the payloads of those packets) which are then individually encrypted, resulting in “data chunks.” There is no teaching in Foster that these packet payloads are themselves sub-divided into shares or chunks.

Ex. 2033 [Rubin-2nd-Decl.] ¶ 145.

But neither Petitioner nor Dr. Bhattacharjee tells us what “data chunks” or “data shares” are or why packet “payloads” are “data shares” (or “data chunks” when encrypted). Certainly, Foster never refers to its packet payloads as “data chunks” or “shares.” At best, Petitioner’s assertion is conclusory and does not satisfy its burden of proof. *KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 418 (2007) (citing *In re Kahn*, 441 F.3d 977, 988 (Fed. Cir. 2006)) (“[R]ejections on obviousness grounds cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness.”); *In re Magnum Oil Tools Int’l, Ltd.*, 829 F.3d 1364, 1380 (Fed. Cir. 2016) (“To satisfy its burden of proving obviousness, a petitioner cannot employ mere conclusory statements. The petitioner must instead articulate specific reasoning, based on evidence of record, to support the legal conclusion of obviousness.”). That is sufficient reason to reject Petitioner’s Foster-Hayhurst ground.

Nor is there any reason to just assume that a packet payload would be deemed a chunk. As Dr. Rubin explains, the literature frequently refers to a packet or packet payload as something that is divided into “chunks” or “shares”:

Petitioner and Dr. Bhattacharjee, again, rely upon Foster’s packet payloads as data shares and encrypted packet payloads as data chunks.

Numerous technical articles, however, refer to a packet or packet payload being divided into chunks or shares. *See, e.g.*, Ex. 2042 [Kumar] 9 (“each node sends out 3 shares in a single packet”); Ex. 2043 [Mao] 9532 (“We break [data packet] D into n shares according to the (t, n) -threshold secret sharing algorithm.”); Ex. 2036 [Hur] 1202 (“The new scheme uses a content-based chunking algorithm to consistently divide both files and packet payloads into multiple byte sequences, called chunks.”); Ex. 2037 [Li-I] 1 (“These services allow breaking down packet payloads into smaller units (called chunks) ...”); *see also id.*, 2 (“It provides a packetization method that breaks down the payload into multiple chunks, each with a certain significance.”); Ex. 2012 [Pek] 3 (“[The d]ata packet is delivered as a sequence of chunks.”); Ex. 2013 [Groote] 3 (“the input is a large data packet ... which is delivered in small chunks.”); Ex. 2038 [McKeown] 4-5 (“A packet consists of multiple chunks...); Ex. 2041 [Li-II] 7 (“the payload could be divided into multiple equal-sized chunks ...”).

Ex. 2033 [Rubin-2nd-Decl.] ¶ 147.

This usage, wherein a packet is divided into data chunks is, moreover, reflected in the ’854 patent itself. Per Dr. Rubin:

Consistent with this usage in the literature, the ’854 identifies a data packet (or packet payload) as something that is divided into shares. Ex. 1001 [’854] 82:21-24 (“At step 4106, ***the data packet or data block may then be partitioned into two or more data shares*** ...”), 81:36-37 (“***Packets of data (e.g., network packets) may also be efficiently split and restored***”), 81:54-58 (“...simplified ***data partitioning process 4010***”).

... *may be performed on network packets*, ...”). Therefore, the ’854 identifies data packets for streaming, such as those disclosed in Foster and in Hayhurst, as the unit to be parsed, not as “data chunks [generated] based on a data set” by “distributing the data set into a plurality of shares” and encrypting. In other words, by disclosing a method for secure transmission of MPEG data packets, neither Foster nor Hayhurst disclose “generating a plurality of data chunks” by “distributing the data set into a plurality of shares” as a POSITA would understand the limitation based on the ’854’s specification. Notably, neither Petitioner nor Dr. Bhattacharjee identifies any teaching in the ’854 or in the technical literature suggesting that a data packet payload would qualify as a “data share” or a “data chunk” when encrypted.

Ex. 2033 [Rubin-2nd-Decl.] ¶ 148.

Moreover, the difference between the claimed “data chunks” and Foster’s packet payloads is significant. As Dr. Rubin explains whereas the ’854’s data chunks or shares are generated to further secure the data, Foster’s packet payloads merely exist as a by-product of the pertinent MPEG streaming standard:

Nor does the ’854’s discussion of shares suggest that a packet payload would be deemed a “data share” or a “data chunk” when encrypted. As the ’854 explains, partitioning the data set into chunks increases security because when “data ... remains whole and intact,” it is “subject to attack.” The ’854 addresses this concern, in part, by “performing a cryptographic parsing and splitting of the encrypted

file.” Ex. 1001 [’854] 56:47-61. Thus, the ’854 teaches that splitting the data set into shares helps secure the data.

A packet payload (as in Foster as well as Hayhurst), however, does not exist to protect the data. Rather it exists by virtue of the MPEG packetized elementary stream standard. Ex. 1006 [Foster] [0072]-[0073]; Ex. 2014 [Overview-of-MPEG-2-Systems] (“An elementary stream is broken up into packets, forming a *packetized elementary stream* (PES).”). These packets allow for transmission of the content. See Ex. 2015 [Newton’s-Telecom-Dictionary] 626 (“**Packet** 1. Generic term for a bundle of data, usually in binary form, organized in a specific way *for transmission*.”); Ex. 2016 [Oxford-Dictionary-of-Computing] 257 (“a group of binary digits forming a message. The group is *transmitted* together”).

Thus, the ’854 splits a data set into shares or chunks as a means of protecting a dataset. This is why a packet in the ’854 is an example of a data set that is split into shares. See Ex. 1001 [’854] 82:21-24. By contrast, Foster’s packet payloads exist not to protect data but, rather, because it is just a by-product of a pre-existing standard. Consequently, a POSITA reading the ’854 patent would not consider a packet payload to be a data share (or a data chunk when encrypted) in view of the ’854’s disclosure.

Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 149-151.

In the ID, the Board preliminarily found that Patent Owner “focuses on Foster individually, and does not address the combination of Foster and Hayhurst”

and that “Hayhurst also generates content segments to secure the data.” ID, 21-22. Patent Owner respectfully disagrees. Petitioner expressly relies on only Foster for the alleged generation of data shares and chunks and relies on Hayhurst for storing and distributing Foster’s encrypted content units. Pet., 74; *see also* 61 (“POSAs had reason to store *Foster’s encrypted content units* ... based on *Hayhurst’s techniques for storing and distributing* data segments.”), 63 (combination involves “*Foster’s* key-storage system and *encryption method to generate encrypted data segments* with *Hayhurst’s method for storing* encrypted data segments ... to yield the predictable result of *Foster’s encrypted content units* being *stored ... as taught in Hayhurst.*”).

To the extent, moreover, that Petitioner changes its combination to rely upon Hayhurst’s segments as the shares (and encrypted segments as the chunks), Hayhurst is of no help either. Hayhurst’s segments are also data packets. Ex. 1007 [Hayhurst] [0020] (“*a segment is a digital packet* that includes data representing a small portion of the movie (video and audio)”). Thus, for the same reasons that Foster’s packet payloads are not shares and encrypted packet payloads are not chunks, Hayhurst’s segments are also not shares (or chunks when encrypted). Ex. 2033 [Rubin-2nd-Decl.] ¶¶ 152.

Thus, Petitioner fails to demonstrate that Foster discloses or renders obvious “generating a plurality of data chunks based on the data set” “wherein generating the data chunk comprises[] distributing the data set into a plurality of shares.”

IV. CONCLUSION

Petitioner fails to demonstrate by a preponderance of the evidence that any of the claims are invalid.

Respectfully submitted,

/ Stephen J. Elliott /

Stephen J. Elliott (Reg. No. 52,858)

Andrei Iancu (Reg. No. 41,862)

SULLIVAN & CROMWELL LLP

Kenneth J. Weatherwax (Reg. No. 54,528)

Nathan Lowenstein, *pro hac vice*

Colette Woo, *pro hac vice*

LOWENSTEIN & WEATHERWAX LLP

Date: September 12, 2024

CERTIFICATE OF COMPLIANCE WITH TYPE-VOLUME LIMITS

This Patent Owner Response (the “POR”) consists of 13,998 words, excluding table of contents, table of authorities, certificate of service, this certificate, or table of exhibits. The POR complies with the type-volume limitation of 14,000 words as mandated in 37 C.F.R. § 42.24. In preparing this certificate, counsel has relied on the word count of the word-processing system used to prepare the paper (Microsoft Word).

Respectfully submitted,

/ Colette Woo /

Date: September 12, 2024

CERTIFICATE OF SERVICE

The undersigned hereby certifies that the following documents were served
by electronic service, by agreement between the parties, on the date below:

PATENT OWNER RESPONSE

EXHIBITS 2033-2044

The names and addresses of the parties being served are as follows:

Elisabeth H. Hunt	EHunt-PTAB@wolfgreenfield.com
Gregory S. Nieberg	GNieberg-PTAB@wolfgreenfield.com
Nathan R. Speed	nspeed@wolfgreenfield.com

Respectfully submitted,

/ Colette Woo /

Date: September 12, 2024