

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE LLC,
Petitioner,

v.

SECURITY FIRST INNOVATIONS, LLC
Patent Owner.

Case No. IPR2024-00212
Patent No. 11,178,116

REPLY DECLARATION OF SAMRAT BHATTACHARJEE, PH.D.

Google Exhibit 1111 Google v. SFI

TABLE OF CONTENTS

I.	Materials Reviewed And Considered	1
II.	My Response to SFI Regarding Ground 1: Tajima	3
	A. Response to Arguments That Tajima’s Key Information Excludes the Key	4
	B. Response To Arguments Regarding Tajima’s Claims.....	13
	C. Response to Arguments That Including the Key in Tajima’s Key Information Would Not Have Been Obvious	18
III.	My Response to SFI Regarding Ground 2: Tajima-Foster	23
	A. Response to Arguments That Tajima and Foster Are Directed to Different Fields and Solve Different Problems.....	23
	B. Response to Arguments That Alleged “Lack of Key Disclosures” in Tajima and Foster Preclude Their Combination.....	25
	C. Response to Arguments That Encrypting Tajima’s Encryption Keys is “Redundant”	28
	D. Response To Arguments That Foster Discourages the Tajima- Foster Combination	30
IV.	My Response To SFI Regarding Ground 3: Tajiima-Rivest.....	32
	A. Response to Arguments That Tajima’s Public Key Would Not Be Used in Rivest’s AoNT	32
	B. Response To Arguments That AoNT Would Not Provide Additional Protection	32
	C. Response to Argument That AoNT Has Terrible Error Propagation Properties	33
V.	My Response To SFI Regarding Ground 4: Tajima-Rivest-Shyu	34
	A. Response to Arguments that Shyu’s Technique Violates “Kerckhoff’s Principle”	34
	B. Response to Arguments That Shyu’s Technique Would Not Provide Security in Transmitting Tajima’s Data Fragments	36
VI.	My Respone To SFI Regarding Ground 5: Foster-HAYHURST	36
	A. Response to Argument that Hayhurst (EX1007) is Not Analogous Art to the ’116 Patent	38

1. POSAs would find Hayhurst to be in the same field of endeavor as the '116 patent.	40
2. POSAs would find that Hayhurst is reasonably pertinent to at least one of the problems to which the '116 patent relates.	49
a. Hayhurst is reasonably pertinent to the problem of user-independent security.	49
b. Hayhurst is reasonably pertinent to the problem of computer-network security.	55
c. Hayhurst is reasonably pertinent to problems addressed by distributed-storage (e.g., the problem of data remaining whole and subject to attack).....	56
B. Response to Argument that POSAs Would Not Have Had Motivation For the Foster-Hayhurst Combination	59
1. It is clear to POSAs from Foster's disclosure that Foster stores encrypted content.....	66
2. The Foster-Hayhurst combination discussed in my First Declaration is consistent with Foster's goals.	85
C. Response to Argument that Foster's Encrypted Payloads Are Not "Data Chunks"	108

I, Samrat Bhattacharjee, Ph.D., declare:

1. I have been retained by Wolf, Greenfield & Sacks, P.C., counsel for Petitioner Google LLC, to assess claims 1-3 and 5-14 (the “challenged claims”) of U.S. Patent No. 11,178,116 (“the ’116 patent”) (EX1001). I am being compensated for my time at my standard rate of \$700.00 per hour, plus actual expenses. My compensation is not dependent in any way upon the outcome of the *inter partes* review of the ’116 patent.

I. MATERIALS REVIEWED AND CONSIDERED

2. My findings, as explained below, are based on my years of education, research, experience, and background in field of securing data from unauthorized access or use, as well as my investigation and study of relevant materials for this declaration. *See also* EX1045 (Supplemental Declaration of Samrat First Declaration, Ph.D.). When developing the opinions set forth in this declaration, I assumed the perspective of a person having ordinary skill in the art (“POSA”), as set forth in Section IV of my initial declaration submitted in this proceeding (EX1003, “First Declaration”).

3. In forming my opinions set forth in this declaration, in addition to the references I discussed and reviewed for my First Declaration, I have reviewed SFI’s POR (Paper 17), Dr. Rubin’s declaration (EX2033), the exhibits cited by SFI and Dr. Rubin, and the following exhibits and papers as well as any papers or

exhibits explicitly cited herein that I cite in response to SFI's POR and/or Dr.

Rubin's declaration:

Exhibit	Description
1072	U.S. Patent Application Publication No. 2004/0019900
1073	U.S. Patent Application Publication No. 2002/0083438
1074	U.S. Patent No. 7,076,661
1075	U.S. Patent No. 6,771,657
1076	U.S. Patent No. 6,389,473
1077	U.S. Patent No. 6,662,329
1078	International Application Publication No. WO 2003/026220
1079	Trappe, Key Management and Distribution for Secure Multimedia Multicast (IEEE Dec. 2003)
1080	D. McGovern, "Pushing the internet direct to the user - security issues," IEE Seminar on IP Over Satellite - The next Generation: MPLS, VPN and DRM Delivered Services, 2003., 2003, pp. 122-138, doi: 10.1049/ic:20030063
1081	U.S. Patent No. 7,277,941
1082	Mike Knee, MPEG Video
1083	U.S. Patent No. 7,266,613
1084	U.S. Patent Application Publication No. 2004/0202320
1085	U.S. Patent No. 7,823,009
1086	U.S. Patent No. 8,098,819
1087	U.S. Patent Application Publication No. 2004/0236958
1088	International Application Publication No. WO 2001/015162
1089	U.S. Patent Application Publication No. 2004/0107356
1090	U.S. Patent No. 7,739,501
1091	U.S. Patent Application Publication No. 2004/0162871
1092	U.S. Patent Application Publication No. 2002/0129383
1093	U.S. Patent Application Publication No. 2004/0005061
1094	U.S. Patent Application Publication No. 2005/0160292
1095	U.S. Patent No. 6,959,086
1096	RFC 3394: Advanced Encryption Standard (AES) key wrap algorithm
1097	U.S. Patent No. 7,257,227
1098	U.S. Patent No. 6,993,508
1099	Deposition of Aviel D. Rubin, Ph.D. (Dec. 12, 2024)
1100	U.S. Patent Application Publication No. 2003/0140257

1101	U.S. Patent Application Publication No. 2004/0199771
1102	U.S. Patent Application Publication No. 2005/0177740
1103	U.S. Patent Application Publication No. 2004/0199667
1104	U.S. Patent Application Publication No. 2005/0050218
1105	U.S. Patent Application Publication No. 2003/0187799
1106	National Institute of Standards and Technology, NIST Special Publication 800-57, Recommendations for Key Management – Part 1: General (Aug. 2005)
1107	U.S. Patent Application Publication No. 2004/0236958
1108	U.S. Patent No. 6,169,843
1109	MPEG-2 - a tutorial introduction to the systems layer (IEEE 1995)
1110	U.S. Patent Application Publication No. 2003/0145329
1111	Reply Declaration of Samrat Bhattacharjee, PH.D.
1112	U.S. Patent Application Publication No. 2004/0010699
1113	European Patent Application Publication No. EP1,564,686
1114	U.S. Patent Application Publication No. 2004/0010699
1115	U.S. Patent Application Publication No. 2004/0187014
1116	RESERVED
1117	U.S. Patent Application Publication No. 2006/0143350

II. MY RESPONSE TO SFI REGARDING GROUND 1: TAJIMA

4. I understand that SFI’s arguments for Ground 1 pertain only to claim 1’s limitation [1D]: “*storing each data chunk [Tajima’s data fragments]...together with one of the plurality of obfuscated different encryption keys.*” SFI and Dr. Rubin argue that Tajima’s “key information” does not include encryption keys, nor would it have been obvious to include encryption keys in the key information. I disagree for the reasons set for in my First Declaration and for the reasons provided below in response to SFI’s and Dr. Rubin’s arguments.

A. Response to Arguments That Tajima’s Key Information Excludes the Key

5. Dr. Rubin testifies that he does not believe that a POSA would understand that Tajima uses the terms “information regarding the key” and “key information” to refer to the key. EX2003, ¶¶41-43. I note that SFI acknowledged that Tajima uses the terms “information regarding the key” and “key information” interchangeably. POR, 1. Dr. Rubin likewise says the terms “information regarding the key” and “key information” refer to the same thing: “some type of information about the key, not to the key itself.” EX2003, ¶43.

6. According to Dr. Rubin, if “Tajima meant to say the ‘key,’ there would be no need to use the phrase ‘information regarding the key.’” EX2003, ¶42. But as I explained in my First Declaration, a POSA understood that Tajima’s key information includes the key and may, and likely would, include *additional* metadata about the key to facilitate recovering the personal information. First Declaration, ¶¶103, 131. Tajima uses the terms “information regarding the key” and “key information” instead of just the “key” because those broader terms were understood to encompass information including the key and other information about or indicative of the key or the encrypted data that facilitate decryption.

7. As I also explained in my First Declaration (¶131), Foster uses the term “key information” in the same way. Specifically, Foster encrypts segments of data with respective “title key[s]” and attaches to each encrypted data segment its

“encrypted title key 112.” Foster, [0064]-[0067]. As shown in FIG. 8 of Foster annotated below, the encrypted title key is included in the “key info” attached to the encrypted data.

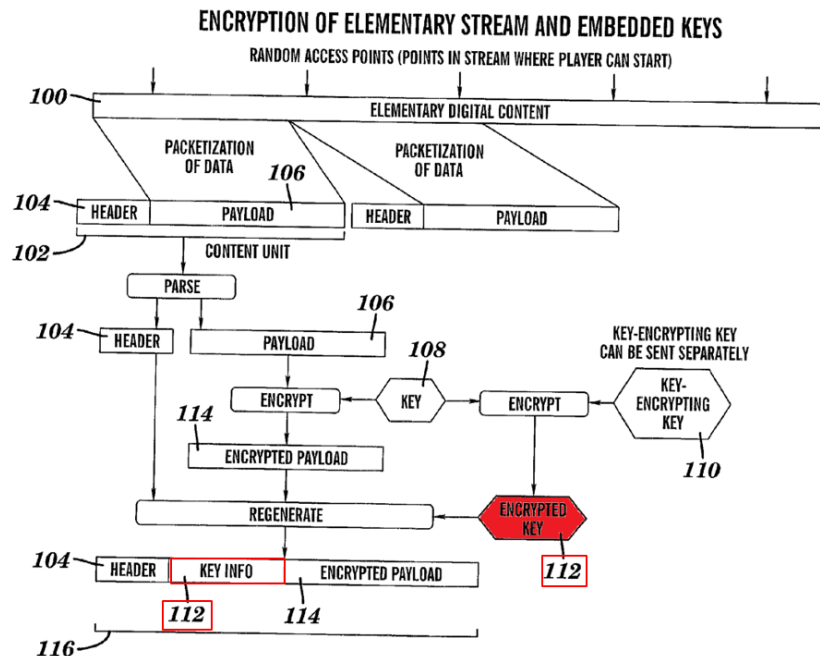


FIG. 8

8. As demonstrated in FIG. 9 of Foster annotated below, the “key information” includes “encrypted key 112” and additional data (“identifier info” and a “verifier”) that relates to the key and/or the encrypted data. Foster, [0066]-[0067]. Foster’s use of the term “key information” to describe information that includes the key and additional information corroborates a POSA understanding that “key information” encompasses information that includes key and other types of metadata, and explains why Tajima did not just say “key” when referring to the information attached to the data fragments.

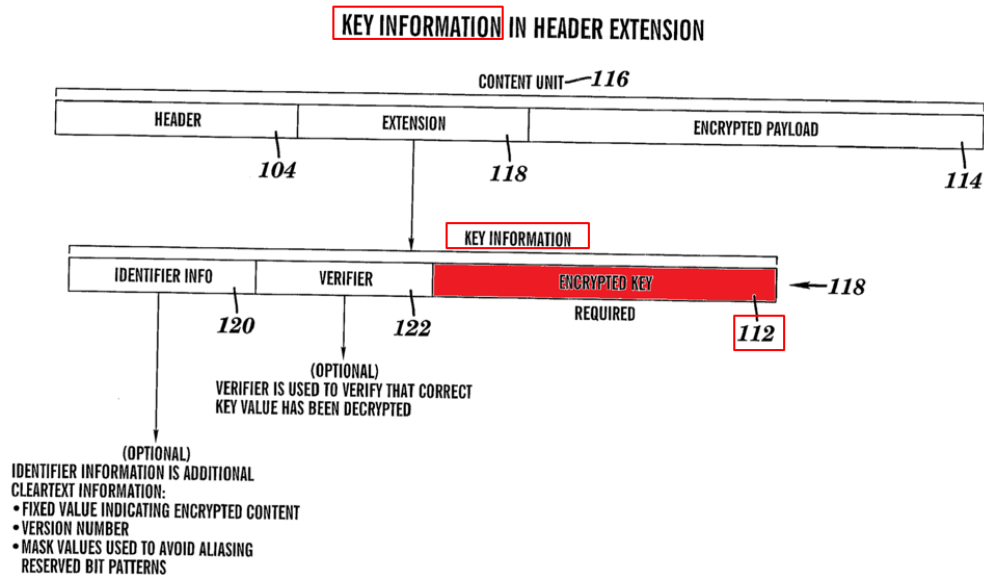


FIG. 9

9. SFI's '854 patent likewise uses the term "key information" to mean information that at least includes the encryption key. EX1015, 79:66-80:38, FIGS. 38 and 39. Specifically, the '854 patent discloses that "[t]he workgroup key concept allows for enhanced protection of information assets by encrypting *key information* stored within the shares."¹ EX1015, 79:66-80:4. The '854 patent then describes a process, illustrated in FIG. 38, in which an encryption key and encrypted data are stored within the shares without using a workgroup key to enhance protection by encrypting key information. EX1015, 80:5-23, FIG. 38. The '854 then discloses that "[w]hen a workgroup key is utilized to protect the

¹ I have added all emphases added throughout this declaration unless I have noted otherwise.

encryption key the above process may be changed slightly to protect the encryption key with the workgroup key”—i.e., “[t]he **encryption key** may then be encrypted with the workgroup key prior to being stored within the shares.” EX1015, 80:24-28. The “modified steps” of the FIG. 38 process are illustrated in FIG. 39 (annotated below) in which the “encryption key [is] encrypted with the workgroup key at stage 3904” before being stored in the shares. EX1015, 80:28-38.

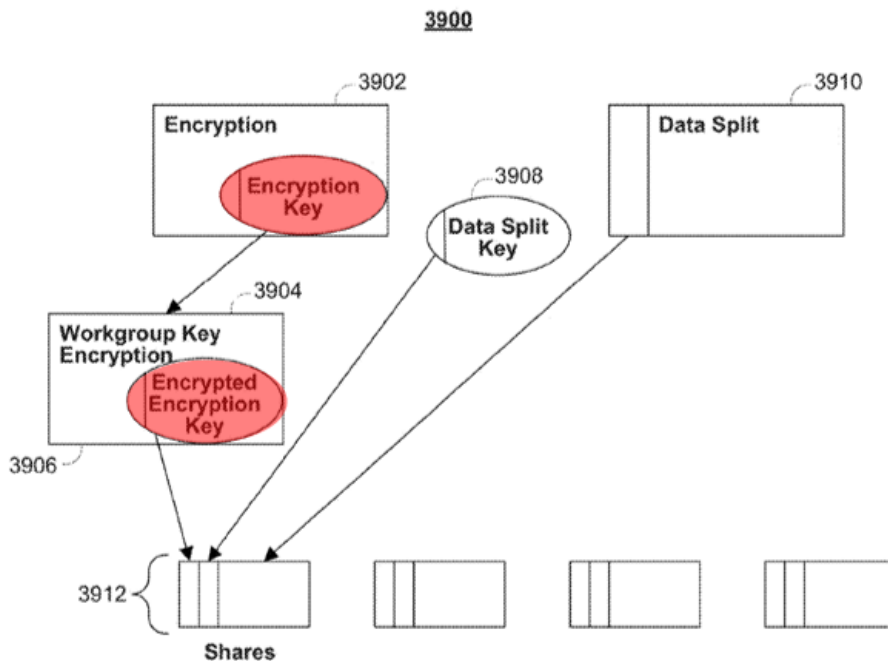


FIG. 39

10. Thus, in the '854 patent's example illustrating how the “workgroup key concept allows for enhanced protection of information assets by encrypting **key information** stored within the shares” (EX1015, 79:66-80:6), the workgroup key enhances protection by encrypting the **encryption key** with the workgroup key

(EX1015, 80:24-46), demonstrating that the '854 patent also used the term “key information” to encompass at least the encryption key.

11. EX1113 likewise uses the term “key information” to encompass at least the key. As shown in FIG. 13 of EX1113 (annotated below), multimedia information is divided into segments D1-D3 (“divided multimedia information”) that are each encrypted with a different encryption key K1-K3. EX1113, [0090]-[0091]. The encryption scheme “adopts the common key system, and therefore the encryption and decryption keys are the same.” EX1113, [0091]. Each encrypted segment D1a-D3a (“encrypted divided multimedia information”) is paired with a decryption key K1a-K3a (i.e., an encryption key K1-K3 that encrypted a different segment) and stored on respective different storage units. EX1113, [0092]-[0093].

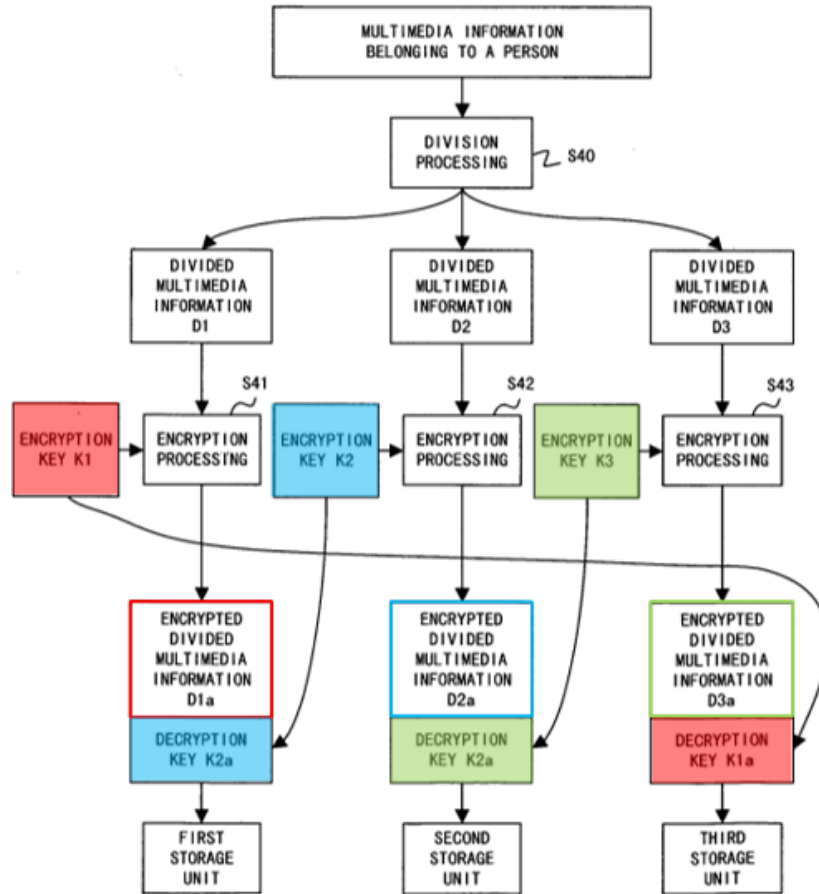


FIG. 13

12. To recover the multimedia information, the encrypted segments are acquired from storage and the “encrypted information is decrypted by using the decryption key being stored in a different storage unit” (EX1113, [0097])—i.e., the encryption key that encrypted a different segment—and then recombined (“collated”) to recover the multimedia information, as shown in FIG. 14 of EX1113 annotated below. This is the very obfuscation technique Tajima discloses. EX1113, [0094]-[0097]. As EX1113 explains, this process enhances security because “[s]toring the *key information* for decrypting the encrypted multimedia in

a storage unit separately from the one storing the encrypted multimedia information itself in the process prevents an illegitimate decrypting if the encrypted multimedia information is stolen, since *the key* for decrypting the encryption is stored in a different storage unit.” EX1113, [0094]. Thus, EX1113 also uses the term “key information” to mean information that includes the encryption key.

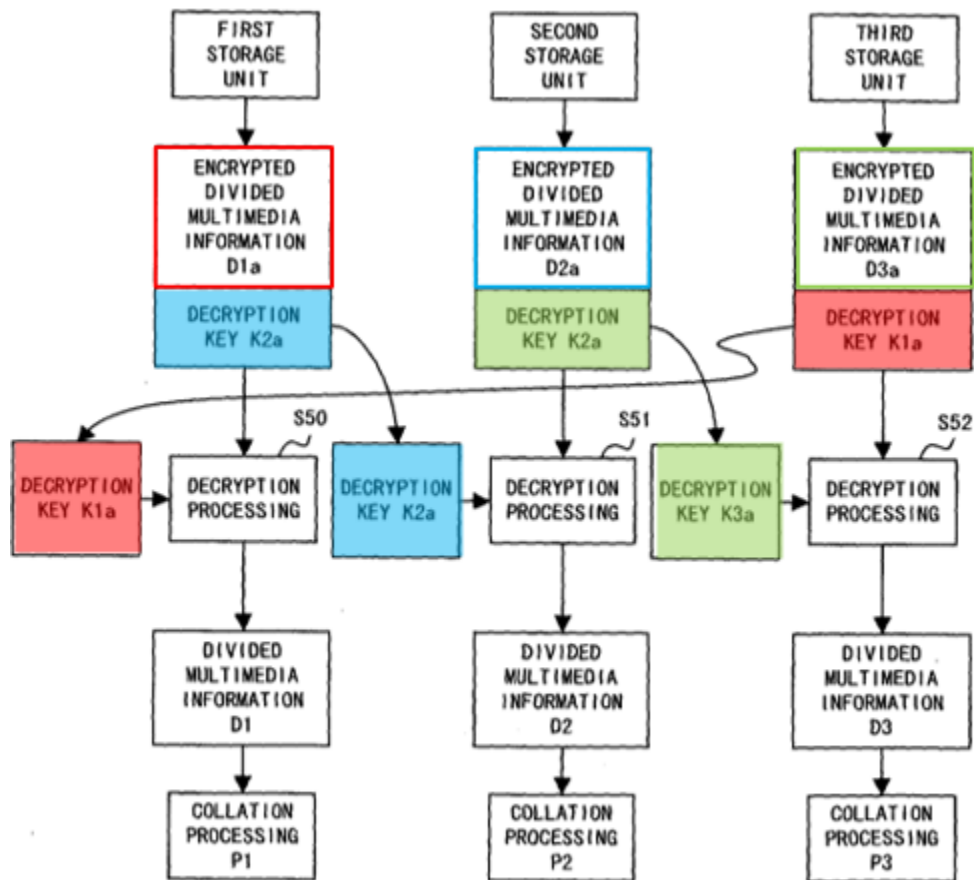


FIG. 14

13. Multiple prior-art references corroborate my opinion, and a POSA’s understanding, that Tajima uses the term “key information” because that term encompasses the key and other information that may be, and often is, included

therein. See also EX1114, [0022] (disclosing that “**key information** preferably comprises **at least a public key** which is accessible to both the owner of the data and the third party, and **a master key** the details of which are known only to the data owner”).

14. Dr. Rubin argues that because in Tajima’s discussion of the creation of a “public key” and “secret key” when registering personal information with a certificate authority, Tajima uses the term “key,” Tajima must be referring to something else entirely when Tajima uses the terms “information regarding the key” or “key information” to describe the information attached to the data fragments. EX2033, ¶42. But instead of reasoning that Tajima uses the terms “key” when Tajima is referring to *just* the key, and uses the term “key information” in its customary way to encompass the key and other information, Dr. Rubin’s concludes that Tajima “appears” to use the term “key information” to refer “some type of information about the key, but ***not*** to the key itself.” That meaning is unsupported by Tajima, contrary to way the term “key information” was conventionally used, and contradicted by Tajima’s claims that demonstrate that Tajima’s key information does include the key (see First Declaration, ¶¶99-102; §II.B *infra*).

15. As I explained in my First Declaration (¶¶96-98, 103), a POSA understood from Tajima’s express disclosure that the “key information” is “used to decrypt” the data fragments that Tajima’s “key information” thus includes the key

because Tajima does not say that anything else is used or is needed to decrypt the data fragments, nor does Tajima disclose any additional steps to obtain the encryption keys that are required to decrypt the data fragments, or any key storage or key management that would be needed if Tajima’s “key information” did not include the key. In response, Dr. Rubin provides his opinion that it “does not logically follow” therefrom that the “key information” includes the key, but he does not explain what does not logically follow nor does he identify any flaw in that logic. EX2033, ¶45.

16. Instead, Dr. Rubin testifies that “[a]ll Tajima discloses is that ‘information regarding the key’ is stored with the data fragment.” EX2033, ¶45. But Tajima also discloses using that key information and *nothing else* to decrypt the data fragments. Dr. Rubin’s testifies that this does not establish that the key information includes the key information because the key information “*could* be ‘used to decrypt’ a data fragment without being the key itself”—e.g., “[i]f the information regarding the key indication, e.g., the type of key that is used, or information that can be used to derive the key.” EX2033, ¶46. Dr. Rubin’s testimony alleges nothing more than that a POSA understood there were other ways of implementing Tajima by excluding the key from the key information and replacing it with other information that Tajima says nothing.

17. I disagree that a POSA reading Tajima's disclosure would understand Tajima to be "silent about what key information is and where the key itself is stored" as Dr. Rubin testifies. EX2033, ¶45. That a POSA would infer nothing from Tajima's disclosure of: (1) using only the key information to decrypt the data fragment, (2) using only information contained within the data fragments to recover the personal information, and (3) without disclosing anything that would be needed if the key information did not include the keys, is in my opinion plainly incorrect. See First Declaration, ¶¶96-98.

B. Response To Arguments Regarding Tajima's Claims

18. Dr. Rubin's testimony that Tajima is "silent about what key information is and where the key itself is stored" is further contradicted by Tajima's claims that confirm that Tajima uses the term "key information" to mean information that includes the encryption key. First Declaration, ¶¶99-102. Dr. Rubin did not comment on this evidence.

19. SFI responded to the Petition's arguments and my testimony that the claims demonstrated Tajima's key information includes the key by alleging that the "Petitioner also argues that key information includes the key itself because one of Tajima's claims (which Petitioner alleges is directed to a Figure 11 embodiment) recites an 'encryption key' where Figure 11 recites 'key information.'" POR, 5. SFI then argues that "the Figure 11 embodiment does not encrypt data fragments,

and thus cannot disclose where the keys used to encrypt data fragments are stored” and that “Petitioner does not rely on Tajima’s Figure 11 embodiment, and so Tajima’s alleged disclosures regarding that embodiment have no bearing on whether the Figure 12 and 13 embodiments (that Petitioner *does* rely on) necessarily disclose attaching the keys used to encrypt the fragments to the fragments themselves.” POR, 5.

20. SFI’s argument (1) ignores the FIG. 13 embodiment entirely discloses everything the FIG. 11 embodiment does, (2) does not refute (nor dispute) that either of those embodiments describe dividing the “key information” into data portions that Tajima’s claims demonstrate includes “the encryption key,” nor (3) that a POSA understood that Tajima uses the same term “key information” for information attached to the data fragments to mean the same thing. See First Declaration, ¶¶99-102 (citing Tajima, [0109]-[0110], [0113]-[0114], FIGS. 11 and 13 and Tajima’s claims 8, 19, 29, 38 and 48).

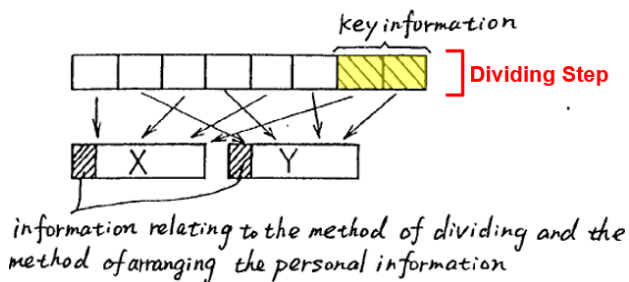
21. As I explained in my First Declaration, (1) both the FIG. 11 and FIG. 13 embodiments describe Tajima’s technique of first encrypting the personal information and then dividing into data portions the encrypted personal information and the information regarding key used to encrypt the personal information, and (2) that this same technique is recited in Tajima’s claim 8 that I used as an example (see also claims 19, 29, 38 and 48). See First Declaration, ¶99.

This is highlighted yellow in the table below. I referred to the FIG. 11 embodiment to demonstrate that **both** of the two embodiments that describe the technique recited in Tajima’s claims refer to the information regarding the key that is divided into data portions as “key information,” both in the respective drawings (the label of which was omitted in FIG. 13) and in the corresponding description of the recovery process. *See* First Declaration, ¶¶100-101. This is also highlighted in yellow in the table below. The FIG. 13 embodiment discloses everything the FIG. 11 embodiment does.

8. A personal information management system according to claim 4, wherein said authentication means encrypts said personal information and divides the encrypted personal information and <u>the encryption key used in the encryption</u> into a plurality of data portions.	
“[0109] As still another example of a method of dividing personal information, as shown in FIG. 11, personal information ... is first encrypted, and the encrypted personal information and <u>information regarding the key that is used in the encryption</u> are then divided into a plurality of data portions each of fixed length as shown in FIG. 8, or of arbitrary length as shown in FIG. 9 or FIG. 10. The plurality of data portions are then arranged into two data fragments, data fragment X and data fragment Y,	“[0113] As yet another method of dividing personal information, as shown in FIG. 13, personal information...is first encrypted, and the encrypted personal information and <u>information on the key that was used in encrypting the personal information</u> are then divided into a plurality of data portions...This plurality of data portions is then arranged into two data fragments, data fragment X and data fragment Y, according to a set method of arrangement, and further, data fragment X and data fragment Y are each encrypted.... <u>Information regarding the key that was used to encrypt data fragment Y</u> is attached to encrypted data

according to a set method of arrangement...”

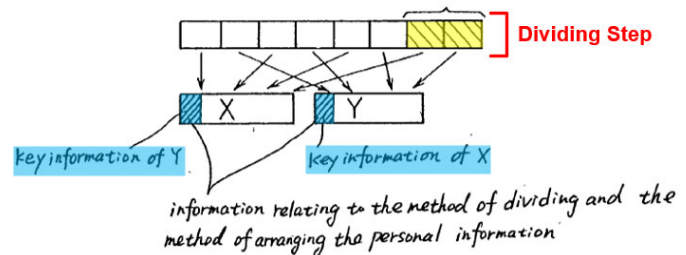
Fig. 11



0110. When combining data fragment X and data fragment Y that have been divided in this way, data fragment X and data fragment Y are combined based on information relating to the method of dividing and the method of arranging if information relating to the method of dividing and the method of arranging has been attached to data fragments X and Y, and **the key information that was attached to the encrypted personal information** is then used to decrypt the encrypted personal information.

fragment X, and **information regarding the key that was used to encrypt data fragment X** is attached to encrypted data fragment Y.”

Fig. 13



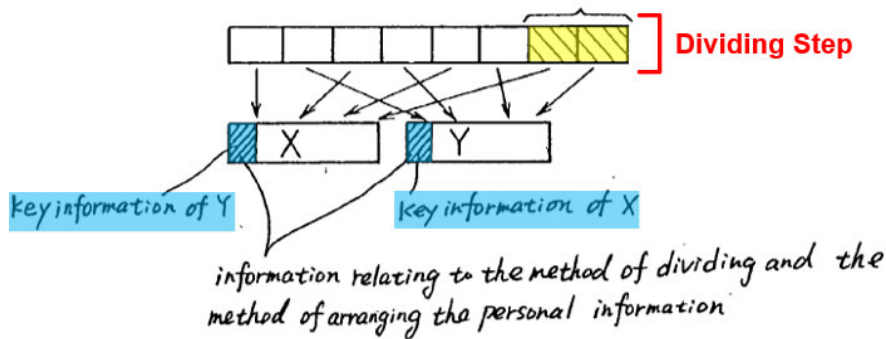
“[0114]. When combining data fragment X and data fragment Y that have been divided in this way, **the key information that was attached to encrypted data fragment X** is used to decrypt data fragment Y, and **the key information that was attached to encrypted data fragment Y** is used to decrypt data fragment X...**the key information that is attached to the combined personal information** is used to decrypt the encrypted personal information.”

22. As highlighted in blue in the table above, (see also paragraphs [0111]-[0112], POSAs understood that Tajima uses the same term “key information” in the same way to describe the information attached to the data fragments (and then used to decrypt them) as the “key information” divided into the data fragments in the same FIG. 13 embodiment (compare yellow and blue highlighting) because Tajima means the same thing: information that includes the encryption key. See

First Declaration, ¶101-102. This argument about what the claims demonstrate does not combine teachings from different embodiments. As shown in the table above, the FIG. 13 embodiment discloses everything the FIG. 11 embodiment discloses and also discloses the further technique of encrypting the data fragments and attaching key information thereto. The FIG. 11 embodiment merely corroborates what the FIG. 13 embodiment also discloses.

23. SFI arguments (POR, 5) ignore both the FIG. 13 embodiment and what Tajima's claims demonstrate in view of that embodiment. SFI never disputes that: (1) Tajima's claims recite "divid[ing] the encrypted personal information and the encryption key used in the encryption into a plurality of data portions" (claims 8, 19, 38, 48); (2) the claimed dividing step is performed on the "key information" in the FIG. 13 embodiment shown in Tajima's FIG. 13 (annotated below); (3) that facts (1) and (2) demonstrate that the "key information" (highlighted yellow below) divided into data portions in this step includes "the encryption key;" nor (4) that a POSA understood that when Tajima uses the same term "key information" for the information attached to the data fragments (highlighted blue below), Tajima means the same thing Tajima means for the "key information" divided into the data fragments, which Tajima's claims confirm includes the encryption key. I note that Dr. Rubin did not comment at all on this evidence.

Fig. 13



C. Response to Arguments That Including the Key in Tajima’s Key Information Would Not Have Been Obvious

24. Dr. Rubin testified that he understands that “Petitioner and Dr. Bhattacharjee present an alternative theory that the inclusion of the key itself in the key information would be obvious,” but that in his opinion, “Petitioner offers no evidence or explanation of why attaching the encryption key itself to a data fragment ‘would have been an obvious and conventional way’ to modify Tajima.” EX2033, ¶47. That is incorrect. My opinion regarding this straightforward implementation of Tajima is based on the same evidence in Tajima’s disclosure demonstrating that Tajima *discloses* that the key information includes the key.

25. Specifically, as I explained in my First Declaration (¶¶96-98, 103) and in my response to SFI’s arguments above, Tajima’s disclosure that nothing more than the key information attached to the data fragments is used to decrypt them informs a POSA that the key information includes the key. *See* First Declaration, ¶97. Additionally, Tajima’s disclosure that all of the information needed to

recombine the personal information is stored in the data fragments reinforces that Tajima is disclosing data fragments that contain all the information sufficient to recover the personal information. *See* First Declaration, ¶¶98. And Tajima’s claims confirm that Tajima’s “key information” includes the key. *See* First Declaration, ¶¶99-102. Thus, Tajima’s disclosure itself demonstrates that Tajima’s key information includes the key.

26. But even if all this evidence from Tajima’s express disclosure, which in my opinion is conclusive, were found not to demonstrate that Tajima *discloses* that the key information includes the key, that would have been a conventional and obvious way to implement what Tajima *discloses* so that the service site authorized to recover personal information beneficially “need only retrieve the stored encrypted data fragments to acquire the keys needed to decrypt them,” rather than “implementing a separate keystore or key management system that would be needed if the key information did not include the key itself.” First Declaration, ¶104.

27. I note that Dr. Rubin does not argue that including the key in the key information would *not* have been obvious, but instead merely gives his opinion that “Petitioner offers no evidence or explanation” of why it would have been obvious. Nor does Dr. Rubin dispute that this obvious implementation would have the above-mentioned benefit. Instead, Dr. Rubin offers his opinion that that

“Petitioner’s proposed modification to Tajima involve where and how to store encryption keys and thus cannot be *merely ‘conventional’*” because cryptography is “a technologically-involved and complex field involving incredibly high stakes” and a POSA “would be hesitant to modify existing practices in favor of untested techniques with no history of peer review.” EX2033, ¶¶48-49.

28. POSAs knew that to decrypt the data fragments using the key information as Tajima discloses, either the keys are included in the key information, or the keys are stored and maintained elsewhere. Both techniques were known, “conventional” and obvious. If Tajima were in fact “silent about what key information is and where the key itself is stored,” as Dr. Rubin alleges (EX2033, ¶45), any implementation would require modifying Tajima in some way that “involve[s] where and how to store encryption keys.” EX2033, ¶49. But according to Dr. Rubin, any such technique cannot be “merely ‘conventional’”—i.e., there would be no conventional way to implement Tajima. I disagree.

29. Storing keys with encrypted data was known and conventional. Tajima itself discloses storing encryption keys with encrypted data. As I explained in my First Declaration (¶¶99-102) and in my response arguments above, Tajima’s claims explicitly disclose storing both the encrypted personal information and the encryption key that encrypted it in the data fragments. Tajima, claims 8, 19, 29, 38 and 48. This fact does not rely on what these claims demonstrate the term key

information means as explained above, but rather this fact is explicit in the claims themselves, which recite an authentication means that “encrypts said personal information and divides the encrypted personal information and **the encryption key used in the encryption** into a plurality of data portions” that are arranged into the data fragments. See claims 8, 19, 38 and 48 and the claims from which they depend. Thus, Tajima itself demonstrates that storing encryption keys with the encrypted data was conventional.

30. Furthermore, as explained in my First Declaration (§§32, 54), EX1013 obfuscates encryption keys in the same way as Tajima. Specifically, EX1013 divides data to be secured into a plurality of shares, encrypts each of the shares, and stores each encrypted share (SD1-SDn) with a key (K1-Kn) that did not encrypt it on respective different storage depositories, as shown in [0354]-[0357] of EX1013 annotated below. EX1013, [0293]-[0368].

[0354] Depository 1: SD1, Kn

[0355] Depository 2: SD2, K1

[0356] Depository 3: SD3, K2

[0357] Depository n: SDn, K3.

31. Furthermore, as discussed above (§II.A), EX1113 likewise describes the same obfuscation technique as Tajima. As shown in FIG. 13 of EX1113 (annotated below), each encrypted multimedia segment D1a-D3a is stored on a

different storage unit with an encryption key K1-K3 that encrypted a different multimedia segment. EX1113, [0090]-[0093].

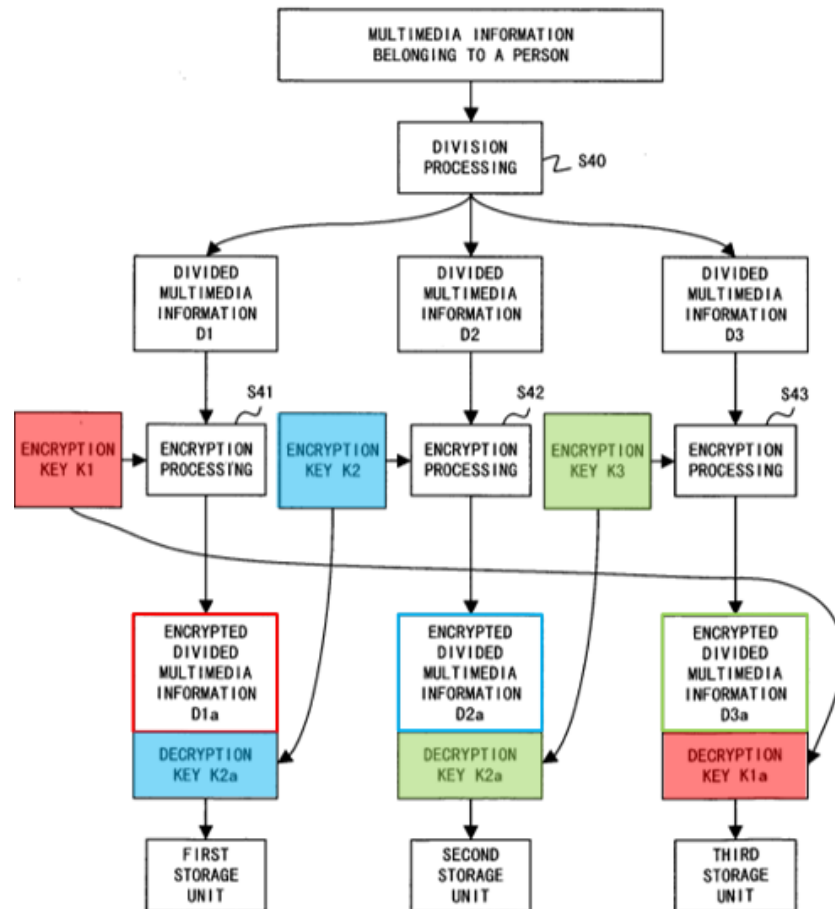


FIG. 13

32. Thus, implementing Tajima to include the keys in the key information attached to a data fragment it did not encrypt *was* conventional. As I pointed out above, Dr. Rubin never says this known implementation would not have been obvious. Instead, Dr. Rubin points to my discussion that storing plaintext keys has known risks, but again Dr. Rubin does not allege that including the key in the key information attached to a data fragment it cannot decrypt would not have been

obvious. EX2033, ¶49 (citing EX1003, ¶140). As I also explained in my First Declaration (¶141), Tajima addresses some of the risk of storing plain text keys by obfuscating them so that key information is not stored with the data fragment it encrypted. The fact that a POSA knew that storing plaintext keys with encrypted data, even when obfuscated, still has risks does not make that known implementation non-obvious.

33. Including keys in Tajima’s key information meant the authorized service site need only retrieve the encrypted data fragments to obtain the keys needed to decrypt them without implementing a separate key-storage system (*see* First Declaration, ¶104), while Tajima’s technique of obfuscating the key information protects against attackers that acquire some but not all encrypted data fragments. First Declaration, ¶141. This known, obvious and beneficial implementation is not rendered non-obvious by other obvious modifications to Tajima that could improve Tajima’s security.

III. MY RESPONSE TO SFI REGARDING GROUND 2: TAJIMA-FOSTER

A. Response to Arguments That Tajima and Foster Are Directed to Different Fields and Solve Different Problems

34. Dr. Rubin provides his opinion that “Tajima and Foster are directed fields and different problems.” EX2033, ¶54. To support his opinion, Dr. Rubin describes differences between the *application* for which Tajima and Foster are

securing data. EX2033, ¶¶54-62. Regardless of the specific applications for which Tajima and Foster secure data, Tajima and Foster are both in the same field as the '116 patent and thus the same field as each other: “securing data from unauthorized access or use” (EX1001, 1:21-22), which Dr. Rubin noted is the '116 patent’s field of the invention and even more generally that “securing data” is the “'116’s Field.” EX2033, ¶170.

35. Thus, Foster and Tajima are both in the same field of “securing data from unauthorized access or use.” *See, e.g.*, Foster, [0003] (relates to “security of content [data]”); Tajima, [0044] (invention “improve[s] the security of personal information [data]”). Tajima and Foster also both describe techniques for securing data using encryption and securely storing keys with encrypted data. *See e.g.*, Tajima, [0109]-[0114]; Foster, [0064]-[0071].

36. I have been informed that, in any event, obvious combinations are not limited to references in the same field, addressing the same problems. Nonetheless, both Tajima and Foster are in the same field of securing data and both encrypt different pieces of data with different encryption keys to do so. Tajima uses the known technique of obfuscating its encryption keys and Foster uses the known technique of encrypting the encryption keys with a key-encrypting key to allow storing keys with the encrypted data. The Tajima-Foster combination straightforwardly uses “Foster’s key-encryption technique” to “encrypt Tajima’s

keys before attaching them to different encrypted fragments to benefit from the increased level of security that would prevent an attacker who successfully acquired all of Tajima’s data fragment[s] from possessing the plaintext key information need to decrypt the data fragments” (*see* First Declaration, ¶154)—i.e., Tajima-Foster uses Foster’s known KEK technique to address the same known security risk of storing plaintext keys with encrypted data that Foster addresses, thus increasing the security of Tajima’s system.

B. Response to Arguments That Alleged “Lack of Key Disclosures” in Tajima and Foster Preclude Their Combination

37. Dr. Rubin provides his opinion that a POSA “would not be motivated to combine Tajima and Foster because they lack key disclosures” (EX2033, ¶65) and specifically that the “silence of Tajima, on the one hand, about attaching the key themselves to data fragments, and the lack of disclosure in Foster about anything corresponding to a data fragment, means that a POSITA would not be motivated to combine (and would not have any reasonable expectation that) these references could be successfully combined to achieve the inventions disclosed in independent claims 1 and 9, as these claims require attaching the key itself to a data chunk.” EX2033, ¶73.

38. First, I disagree that Tajima does not disclose attaching the keys themselves to data fragments for the reasons I discussed in my First Declaration

and in my response in §II.A above. See First Declaration, ¶¶96-104. Second, even if Tajima did not disclose attaching the keys themselves to data fragments, the Tajima-Foster combination uses “Foster’s key-encryption technique” to “encrypt Tajima’s keys before attaching them to different encrypted fragments to benefit from the increased level of security that would prevent an attacker who successfully acquired all of Tajima’s data fragment[s] from possessing the plaintext key information need to decrypt the data fragments.” First Declaration, ¶154. Dr. Rubin’s arguments directed to what Tajima allegedly does not disclose are not relevant to the Tajima-Foster *combination*.

39. Dr. Rubin’s allegation that Foster lacks disclosure “about anything corresponding to a data fragment” (EX2033, ¶73) is likewise irrelevant to the Tajima-Foster combination. Dr. Rubin acknowledges that “Tajima discloses dividing a data set into data fragments corresponding to the claimed data chunks” (EX2033, ¶66) and that “Petitioner maps the claimed ‘plurality of data chunks’ to Tajima’s data fragments (EX2033, ¶67). I disagree that Foster’s “contents units” are not “data chunks” (*see* Section VII.C below), but whether Foster’s “content units” are “data chunks” or whether they “correspond to Tajima’s data fragments” (EX2033, ¶67) is irrelevant because the claimed data chunks are met in Tajima-Foster by Tajima’s data fragments (not Foster’s content units). Both Tajima and Foster encrypt pieces of data and Foster encrypts the encryption keys to protect

them when storing them with the encrypted data. Dr. Rubin never explains how any alleged differences between Foster's content units and Tajima's data fragments would have dissuaded a POSA from using Foster's well-known KEK technique to encrypt Tajima's encryption keys. A POSA knew the enhanced security provided by a KEK is not dependent on the content of data being secured. Moreover, Foster's KEK encrypts *encryption* keys, not the data for which Dr. Rubin alleged differences.

40. Dr. Rubin alleges parenthetically that because of these same alleged lack of disclosures in Tajima and Foster that a POSA "would not have any reasonable expectation that [] these references could be successfully combined." EX2033, ¶73. Dr. Rubin neither explains what would be beyond the ordinary skill of a POSA, nor addresses the actual Tajima-Foster combination. A POSA would have unquestionably had a reasonable expectation of success of encrypting Tajima's keys using Foster's KEK techniques before attaching them to Tajima's data fragments. As I explained in my First Declaration (¶¶160-163), implementing Tajima-Foster is a routine matter of performing an encryption step before attaching Tajima's keys to the data fragments and maintaining the KEK. Dr. Rubin does not dispute that these well-known steps would have been well within the ordinary skill of a POSA.

C. Response to Arguments That Encrypting Tajima's Encryption Keys is "Redundant"

41. Dr. Rubin further alleges that a POSA "would not be motivated to combine Foster to Tajima because Tajima's key information is already secured by hiding it—by attaching it to a data fragment it cannot decrypt, and storing different data fragment/key information combinations in different locations" and therefore using Foster's KEK to encrypt Tajima's key would be "redundant." EX2033, ¶74.

42. As I explained in my First Declaration (*see* EX1003, ¶141), even though Tajima obfuscates the encryption keys, the risk remains that if an attacker is able to recover all of Tajima's data fragments, the attacker will be in possession of the plaintext keys used to encrypt the entire data set. Decrypting the data fragments would simply require determining the correct encryption key/data fragment combinations which an attacker could easily do. Thus, while Tajima's key obfuscation provides protection against attackers who obtain a subset of data fragments, Tajima's key obfuscation provides little deterrence to an attacker that has successfully acquired all fragments.

43. By using Foster's key-encryption technique to encrypt Tajima's keys before attaching them to different encrypted fragments, an attacker that successfully acquired all data fragments would not possess all the keys in plaintext form, but instead would possess only undecipherable encrypted keys that could not be used to decrypt any data fragments unless and until an attacker was also

somehow able to obtain the corresponding KEK. *See* First Declaration, ¶147. Requiring an attacker to obtain KEKs to decrypt Tajima’s encryption keys in addition to de-obfuscating them is not “redundant” to only de-obfuscating them. Encrypting Tajima’s encryption keys before obfuscating them increases security because the brute force tactic of trying different combination could not defeat Tajima-Foster’s security whereas it could defeat Tajima’s obfuscation alone if an attacker acquired all encrypted data fragments. Addressing this vulnerability was specifically identified as a reason a POSA would have been motivated to apply Foster’s KEK to Tajima. First Declaration, ¶154.

44. Dr. Rubin does not dispute that Tajima-Foster increases security in this way. Instead, Dr. Rubin points out a different situation where Tajima’s obfuscated encryption keys would not be defeated merely by brute force: “an attacker with access to a system that stores fragments according to Tajima does not have the key information needed to decrypt that fragment.” EX2033, ¶75. But Dr Rubin ignores the situation in which Tajima-Foster indisputably increases security—i.e., if an attacker successfully acquires all the encrypted data fragments—and never refutes that applying Foster’s key-encryption to Tajima’s keys would have provided a more secure system. POSAs understood that Tajima-Foster increases security even if an attacker acquires less than all of the encrypted data fragments. For example, by finding correct key/data fragment combination(s),

partial information could be decrypted if an attacker acquires multiple encrypted data fragments where one or more of those encrypted data fragments includes the key for one or more other encrypted data fragments. Foster's key-encryption prevents decryption of any of the data fragments unless an attacker **also** acquires the KEK (i.e., using a KEK prevents recovering any piece of data **even if** the attacker acquires all of the encrypted data fragments **and** data-fragment keys). EX1090, 2:39-41 ("The more difficult it is for an unauthorized entity to obtain a valid key, the more secure the cryptographic scheme.").

D. Response To Arguments That Foster Discourages the Tajima-Foster Combination

45. Dr. Rubin alleges that "[a]ny combination [of Tajima and Foster] would require eliminating the benefits of both systems to create a system that cannot achieve the objective of either reference." EX2033, ¶76. But there is only one relevant Tajima-Foster combination—"using Foster's key-encryption technique [] to encrypt Tajima's keys before attaching them to a different encrypted fragment" (First Declaration, ¶154)—and none of Dr. Rubin's arguments pertain to that combination.

46. First, Dr. Rubin argues that Tajima attaches key information to a data fragment the key information did not encrypt, whereas it is "essential to Foster's system" to "attach[] the key to content it can decrypt for synchronized transmission of the two together." EX2033, ¶78. That distinction is irrelevant because Tajima-

Foster does not modify Foster. Instead, Tajima-Foster modifies Tajima's system to improve security for Tajima's data, which does not require synchronization because, unlike Foster, Tajima is not concerned with playing back media data, where synchronization can be helpful. The benefits of using Foster's KEK in Tajima is independent of Foster's synchronization and synchronization of Foster's data is irrelevant to Tajima-Foster because Foster's data is neither being secured nor restored in Tajima-Foster.

47. Encrypting Tajima's key with a KEK does not eliminate Tajima's benefits or objectives. Instead, Tajima-Foster improves Tajima's security, furthering Tajima's objective of preventing unauthorized access to personal information. Tajima, [0026], [0042]-[0043]. That Tajima-Foster improves security of Tajima's data fragments should an attacker successfully acquire all of them is undisputed. Foster keeps encrypted media segments with their keys to facilitate randomly accessing different media segments from Foster's media content for playback. Foster, [0006]-[0007]. That does not "discourage" the Tajima-Foster combination from using Tajima's well-known obfuscation technique that stores a key separate from the data it secures given that Tajima's data is not randomly accessed, nor would there be any benefit in randomly accessing Tajima's data. It was well known that KEK's protected encryption keys stored with encrypted data. That Foster applies a KEK's protection to facilitate random access to different

media segments of media content does not discourage applying Foster's KEK protection to protect Tajima's obfuscated encryption keys to provide the security benefit that Dr. Rubin never disputes.

IV. MY RESPONSE TO SFI REGARDING GROUND 3: TAJIIMA-RIVEST

A. Response to Arguments That Tajima's Public Key Would Not Be Used in Rivest's AoNT

48. Dr. Rubin argues that a POSA "would not be motivated to apply Rivest's AoNT using the user's public key" allegedly because "Rivest says nothing about the use of public keys and is not designed to be used with public keys." EX2033, ¶84. But Tajima-Rivest merely uses the user's public key as the random key K_0 in Rivest's transform. See First Declaration, ¶189. Tajima's public key which would perform the same transform function as Rivest's key K_0 that Rivest says is a "publically-known encryption key" (EX1034, 214). Dr. Rubin does not allege any properties of Tajima's public key that would make it unsuitable for use as the key K_0 in Rivest's transform.

B. Response To Arguments That AoNT Would Not Provide Additional Protection

49. First, Dr. Rubin argues that without a secret key or subsequent splitting, Rivest's AoNT provides no security, because anyone who knows the user's public key could use it to invert the transform. EX2033, ¶¶85-87. Dr. Rubin's argument ignores that an attacker would have to know or ascertain that

AoNT was applied in the first place. Even if an attacker somehow could ascertain that AoNT was applied, an attacker would still have to ascertain that a user's public key was used as for K_0 and which user's public key was used on the data fragment the attacker was able to acquire. None of this information is *a priori* known to an attacker nor is any of this information apparent from the data fragment itself.

50. Second, Dr. Rubin argues that a secret key or “splitting” would need to be used for AoNT to provide additional security. EX2033, ¶87. But this contradicts the very way the '116 patent proposed using AoNT—i.e., the '116 patent discloses that “it will be understood that the AoNT may be applied to encryption keys used to encrypt the data portions *following parsing and splitting*.” EX1001, 77:13-43.

C. Response to Argument That AoNT Has Terrible Error Propagation Properties

51. Dr Rubin argues that a POSA would not have applied AoNT because Rivest says “all-or-nothing encryption has terrible error propagation properties” because if any message block is damaged “then it is likely that every message block will be damaged” (Rivest, 216). EX2033, ¶88. I disagree that a POSA would understand Rivest to be discouraging the use his own transform. Rivest is simply pointing out the importance of using reliable transmission means to reduce data corruption and/or to use error-correction techniques. EX1034, 216. Second, in the

Tajima-Rivest combination the transformed data—both the first implementation where the key information is transformed, and in the second implementation in which the data fragment is transformed using the key information (see First Declaration, ¶¶189-190)—the individual transformed message blocks are kept together, transmitted and stored together. Thus, there is no “error propagation” in Tajima-Rivest.

V. MY RESPONSE TO SFI REGARDING GROUND 4: TAJIMA-RIVEST-SHYU

A. Response to Arguments that Shyu’s Technique Violates “Kerckhoff’s Principle”

52. Dr. Rubin argues that a POSA would not apply Shyu’s technique of applying different encryption algorithms to different pieces of the same dataset because “doing so would violate Kerckhoff’s [sic] Principle, a long-standing and fundamental tenet of cryptography.” EX2033, ¶93. I both disagree that Kerckhoffs’ principle applies to Shyu’s and that a POSA would understand that Shyu’s techniques would not provide the security that Shyu discloses.

53. Dr. Rubin cites EX2007 (“Katz”), which explains that what is now known as Kerckhoffs’ principle (the “cipher method must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience.”) was articulated in a late 19th century paper outlining design principles for military ciphers. (EX2007, 17). As Katz explained in the next

sentence, this means that “the encryption scheme itself should not be kept secret”—i.e., Kerckhoffs was referring to actual secret *algorithms* where the secret is *how* the algorithm itself works. EX2007, 17. As Katz further explains, the two primary arguments in favor of Kerckhoffs’ principle are: (1) it is simpler to share a secret key than a secret program thousands of times longer, and (2) in the event of a “leak” it is easier to simply switch keys than switch to a different secret algorithm. EX2007, 18.

54. Shyu does not use secret algorithms in first place, but rather discloses choosing among a set of *known* algorithms for different data segments to make it more difficult for an attacker who has deciphered an algorithm used to scramble one data segment cannot apply it to all segments. Shyu, 1:24-42, 3:8-11 (“The *scrambling algorithms* employed in the present invention are *known in the art*”), 3:12-38, 3:56-60; First Declaration, ¶202. Varying between different known algorithms is no more a “secret” than if Shyu had used just one. That the known algorithm that is used is unknown to an attacker is not the secret Kerckhoffs’ principle refers to. Second, Kerckhoffs’ principle does not say that secret algorithms do not provide security—merely that they are less preferable to secret keys. EX2007, 18. Also, EX1036 (Takeda) that I cited in my First Declaration (¶203) recognized the same risk as Shyu and disclosed the same solution of using different ciphering algorithms. This corroborates that a POSA understood this

technique provided additional security and a POSA would have had no reason to doubt that.

B. Response to Arguments That Shyu's Technique Would Not Provide Security in Transmitting Tajima's Data Fragments

55. Dr. Rubin further argues that because Shyu's technique provides protection during transmission from a single source to a single destination and Tajima discloses that different data fragments are transmitted to different destinations, Shyu's technique would not protect Tajima's data fragments during transmission. EX2033, ¶¶98. But increasing security of Tajima's data fragments during transmission is not what the Petition proposes. The Tajima-Rivest-Shyu combination applies Shyu's principle to Tajima to provide the same increased security for unauthorized access to Tajima's data fragments once they are stored.

VI. MY RESPONSE TO SFI REGARDING GROUND 5: FOSTER-HAYHURST

56. As I discussed in my First Declaration (¶¶212-275), Ground 5 combines Foster (EX1006) with Hayhurst (EX1007). In particular, the Foster-Hayhurst combination combines Foster's encryption methods with Hayhurst's distributed-storage technique. First Declaration, ¶¶240-253. As I discussed in my First Declaration (¶¶213-224, ¶¶255-266), Foster divides a *dataset* (e.g., a movie file) into segments (which Foster calls "payloads"), encrypts each segment with a respective key, encrypts those keys with the same key-encryption key (KEK), and

stores all encrypted segments plus corresponding encryption keys on a storage device. *E.g.*, Foster, [0064]-[0075], FIG. 8. As I also discussed in First Declaration (¶¶227-234), Hayhurst, like Foster, also “segments” a “media file (movie),” encrypts each segment, and then separately distributes those encrypted segments to different storage devices for secure storage (and other benefits that Hayhurst identifies and that POSAs would have understood from Hayhurst’s disclosure). Hayhurst, [0006]-[0007].

57. As I discussed in my First Declaration (¶¶240-246), POSAs would have recognized that utilizing Hayhurst’s distributed-storage technique in Foster would have provided several benefits to Foster’s system, such as (1) reducing bandwidth requirements (and increasing the speed of delivery) for a content service provider delivering content to customers, (2) adding redundancy and backup to the storage and distribution of data, and (3) increasing the “security and a deterrence to piracy” (Hayhurst, [0008]).

58. My opinions in that regard were corroborated not only by Hayhurst’s explicit disclosure but also by prior art I cited that that demonstrated POSAs background knowledge about the benefits of using a distributed storage technique, both in terms of storing and delivering content. First Declaration, ¶¶240-246 (citing Hayhurst itself, EX1028, EX1029, EX1030, EX1031). As I discuss below (§VI.B), in response to SFI’s and Dr. Rubin’s arguments otherwise, my opinion regarding

POSAs' motivation to pursue Foster-Hayhurst and their reasonable expectation of success in doing so are corroborated by the evidence I cited in my First Declaration and the additional corroborating evidence I cite in this declaration in response to SFI's arguments and Dr. Rubin's opinions.

59. As I discuss further below (§§VI.A-VI.C), in reviewing SFI's POR and the Rubin Declaration, SFI appears to have three primary arguments challenging how Foster-Hayhurst meets the challenged claims: (1) SFI argues that Hayhurst is not analogous art (POR, 38-47); (2) SFI argues that POSAs would have lacked motivation to pursue the Foster-Hayhurst combination (POR, 47-68); and (3) SFI argues that Foster does not disclose claim 1's *data chunks* (POR, 68-70). As I discuss in the sections below (§§VI.A-VI.C), in my opinion POSAs would disagree with each of these arguments.

A. Response to Argument that Hayhurst (EX1007) is Not Analogous Art to the '116 Patent

60. I have been informed that, in an obviousness analysis, prior art must be analogous to the patent being considered. I have been informed that there are two ways by which a prior art reference can be considered to be analogous, or in the same field of art: the reference can **either** be (1) in the same field of endeavor as the challenged patent, regardless of the problems the challenged patent and the prior art address; **or** (2) reasonably pertinent to at least one of the problems being solved by the challenged patent. I have been informed that a prior art reference

may be both in the same field of endeavor as the patent and reasonably pertinent to a problem in the patent, but the reference only has to meet one or the other to qualify as analogous art (i.e., in the same field **or** reasonably pertinent to one problem faced by the inventors). I have been formed that a prior art reference can be in the same field of endeavor even if it addresses different problems than the patent. I have also been informed that a reference is reasonably pertinent if, though in a different field, it would have commended itself to an inventor's attention in considering one of the problems the inventor is addressing. I have further been informed that in determining whether a reference is reasonably pertinent to a problem, one should consider a problem faced by the inventor, as reflected either explicitly or implicitly, in the specification. I have further been informed that courts have been directed to construe the scope of analogous art broadly because POSAs understand that familiar items may have obvious uses beyond their primary purposes, and POSAs often will be able to fit the teaching of multiple patents together like pieces of a puzzle. I have been informed that the analysis of whether a prior art reference is analogous art should take into account any relevant evidence in the record cited by the parties to demonstrate the knowledge and perspective of POSAs.

61. SFI argues, and Dr. Rubin opines, that Hayhurst is not analogous art because it is allegedly not in the same field of endeavor as the '116 patent and

allegedly not relevant to any of the problems addressed by the '116 patent. POR, 38-47.

62. As I explain below (§§VI.A.1-VI.A.2), it is my opinion that POSAs would have disagreed with SFI and Dr. Rubin on both accounts: in my opinion, POSAs would have recognized that Hayhurst is **both** in the same field of endeavor as the '116 patent **and** reasonably pertinent at least one problem (and in fact several programs) addressed by the inventors of the '116 patent.

1. POSAs would find Hayhurst to be in the same field of endeavor as the '116 patent.

63. SFI argues “the '116’s field is securing data,” including securely storing and communicating the secured data. POR, 40. I have not been asked to opine on whether SFI has identified the correct field of endeavor for the '116 patent. That is not necessary to my opinion because, for the reasons I discuss below, it is my opinion Hayhurst is (at least) in the same field of endeavor that SFI has identified for the '116 patent.

64. My opinion that Hayhurst is in the field of securing data is supported by Hayhurst’s explicit disclosure. For example, Hayhurst’s “Technical Field” section explicitly says that Hayhurst “relates to the field of electronic media delivery...including the **secure handling and storage** of such media.” Hayhurst, [0002]. That Hayhurst explicitly says its “Technical Field” includes **secure** data

handling (*e.g.*, data communication/transmission) and **secure** data storage would be dispositive to POSAs' recognition that Hayhurst is in the field of securing data.

65. But even if Hayhurst had not explicitly identified its own field of endeavor as including secure handling and storage of data, the remainder of Hayhurst's disclosure reinforces that Hayhurst's teachings are in the field of securing data. *See also* First Declaration, ¶¶232-233, ¶¶242-243 (describing how Hayhurst's technique provides security for stored data).

66. Hayhurst provides (increased) **security** and deterrence via its segmented and distributed storage of the content. At [0008], Hayhurst says that its "segmented and distributed storage of the media file **provides a high level of security** and a deterrence to piracy." POSA would agree because, in Hayhurst's storage technique, "[n]o single subscriber unit has more than a segment of a media file, which segment, by itself, has little value." Hayhurst, [0008]. That is, in Hayhurst data has "a high level of security" (Hayhurst, [0008]) and are protected from privacy because there is not a single location where the data is stored and no single storage device / user can, by itself, undermine the secure storage of the file overall because each storage device has only a small portion of the data set that, on its own, cannot be used to recover the data set. For example, even if one storage device / user were subject to attack, other segments would still be secured at the other storage devices. As I discuss further below (§VI.B), the security that

Hayhurst provides via its distributed storage would have provided a benefit to Foster's system because Foster generally contemplates storing the encrypted data segments contiguously (i.e., the segments are stored with each other) at Foster's media server and then again at an end-user. *See, e.g., Foster, [0048]*. Thus, unlike in Hayhurst, an attacker would only have to breach the security of a single storage device in Foster to obtain access to all *data chunks* of Foster's dataset.

67. Consistent with the above explicit disclosures in Hayhurst (and POSAs' understand thereof), Hayhurst expressly teaches that its "approach to segmenting the movie [data] provides **increased security** from piracy." Hayhurst, [0021]. Hayhurst further demonstrates that it is in the field of securing data because it teaches that its "segments are each **encrypted**...to prevent unauthorized viewing." Hayhurst, [0007]. Similarly, Hayhurst explicitly says that its technique provides additional data-storage security (i.e., Hayhurst's technique is in the field of securing data) because restoring the data in Hayhurst also requires knowledge of "the correct method for recombining the segments," which is "stored in a password-protected file, linked to an encryption key." Hayhurst, [0021]. In other words, Hayhurst's technique would provide additional security for and is consistent with a system (like Foster's) that is encrypting segments of data but not using a distributed storage technique (like Hayhurst's).

68. Thus, as I’ve discussed above, Hayhurst explicitly and repeatedly provides disclosures in its specification that demonstrates that Hayhurst’s field is securing data, which is the same field as the ’116 patent that SFI alleges.

69. Despite the explicit and repeated disclosures in Hayhurst demonstrating that its field is securing data, SFI and Dr. Rubin attempt to limit Hayhurst to the field of “decentralized media delivery.” POR, 41-42. To the extent SFI is alleging that is outside the field of securing data, then in my opinion POSAs would disagree because decentralized delivery and distributed storage of media is the primary way **how** Hayhurst provides “**security** and [] deterrence to piracy” of stored data. Hayhurst, [0008]. Hayhurst generates content segments and stores the segments on separate storage devices for the purposes of securely storing the data by making it more challenging for an attacker to obtain all of the *chunks* derived from the original data set. Indeed, like Hayhurst, the ’116 patent **itself** says that one way that it secures data is via “geographical separation of the data storage facilities.” EX1001, 20:32-38. In other words, the ’116 patent itself recognizes that using distributed storage devices (as in Hayhurst) is within the field of securing data. Again, increasing security of the data is one of the main reasons Hayhurst proposes to store segments of encrypted data on different storage devices.

70. I further note that the '116 patent also says that its disclosures are useful in “peer to peer” networks and “video broadcasting systems” (EX1001, 6:48-53, 70:23-29), which are two of the very same applications to which Hayhurst’s teachings are also directed, reinforcing that their techniques apply to securing data in similar contexts. *See* Hayhurst, [0005] (“**The present invention addresses** the problem of prohibitive bandwidth requirements for a content provider in **delivering video-on-demand or other large media files.**”), [0010] (“It is contemplated that the request for the media file and subsequent responses for directing the segments to the requester can be handled without the intervention of the network’s main server **by using peer-to-peer architecture** or the like.”). POSAs would have known that peer-to-peer / distributed storage and distribution networks like Hayhurst’s could be used to deliver live television broadcasts.

71. Indeed, as I discussed I was co-author of several peer-reviewed journal articles (such as those listed below) that were intended for POSAs in the field of securing data that described techniques for using systems like Hayhurst’s to distribute live (and stored) media.

- Suman Banerjee and Samrat Bhattacharjee. “Scalable Application-Layer Multicast for Content Distribution”, Computer Communications Workshop, 2001.

- Suman Banerjee, Bobby Bhattacharjee, and Christopher Kommareddy. “Scalable Application- Layer Multicast”, Proceedings of ACM SIGCOMM, pages 205–217, 2002.
- Suman Banerjee, Christopher Kommareddy, Koushik Kar, Bobby Bhattacharjee, and Samir Khuller. “Construction of an Efficient Overlay Multicast Infrastructure for Real-time Applications”, Proceedings of IEEE Infocom, pages 1521–1531, 2003.
- Suman Banerjee, Seungjoon Lee, Bobby Bhattacharjee, and Aravind Srinivasan. “Scalable Resilient Multicast”, Proceedings of ACM SIGMETRICS, pages 102–113, 2003.
- Suman Banerjee, Seungjoon Lee, Ryan Braud, Bobby Bhattacharjee, and Aravind Srinivasan. “Scalable Resilient Media Streaming”, In Proceedings of ACM NOSSDAV’04, pages 4–9, 2004.
- Suman Banerjee, Christopher Kommareddy, Koushik Kar, Bobby Bhattacharjee, and Samir Khuller. “OMNI: An Efficient Overlay Multicast Infrastructure for Real-time Applications”, Special Issue of Computer Networks on Overlay Distribution Structures and their Applications, Vol 50:6, pages 826-842, 2005.
- Suman Banerjee, Seungjoon Lee, Bobby Bhattacharjee, and Aravind Srinivasan. “Resilient Multicast using Overlays”, IEEE/ACM Transactions of Networking, Vol. 14, No. 2, pages 237–248, April 2006.

72. Thus, the fact that Hayhurst uses peer-to-peer / distributed storage and distribution certainly does not take Hayhurst outside the field of securing data and,

if anything, places it squarely within that field because, as I’ve discussed, Hayhurst’s distributed storage is done for the purpose of adding security (among other benefits).

73. I understand that SFI’s argument is that notwithstanding Hayhurst’s (repeated) “disclosures about the security of the media files” those disclosures should be ignored because those disclosures allegedly are only “tangentially relate[d]” to securing data in Hayhurst. POR, 42. In my opinion, POSAs would disagree with SFI’s argument. As I discussed above, Hayhurst **explicitly** says that its “Technical Field” includes “the **secure** handling and storage of [] media.” Hayhurst, [0002]. As I also discussed above, Hayhurst provides numerous disclosures explaining how Hayhurst’s technique provides “**security** and [] deterrence to piracy” of stored data. Hayhurst, [0008]. In my opinion, POSAs would find that SFI’s allegation that data security is “tangential” to Hayhurst (POR, 42) is irreconcilable with what Hayhurst itself says is its “Technical Field” and what Hayhurst says about how the security of storing data is enhanced by Hayhurst’s distributed storage techniques. Again, POSAs would recognize that Hayhurst’s “segmented and distributed storage” of data is not tangential to securing data—it is **how** Hayhurst secures data to “**provid[e] a high level of security**.” Hayhurst, [0021].

74. My opinion that Hayhurst’s distributed storage and distribution technique is in the field of securing data is consistent with testimony that Dr. Rubin provided at deposition. Dr. Rubin testified that to secure data from unauthorized access it “means that parties that shouldn’t – that haven’t been given access or haven’t purchased access and are not parties that should have the access can’t get access to the data.” Rubin-Depo., 108-109. He further testified that “[s]ecur[ing] data” means “protect[ing] data” and to “[p]rotect the data” means “[t]o provide security for the data such that it isn’t available to people who are not supposed to have it and that they can’t be changed.” Rubin-Depo., 108-109.

75. The same is true in Hayhurst as a result of Hayhurst’s distributed-storage technique. For example, Hayhurst’s technique provides “**security** and [] deterrence to **piracy**” (i.e., preventing parties that haven’t been given access or purchased access from accessing the secured data). Hayhurst, [0008].

76. SFI further argues Hayhurst that does not teach that security was a problem in prior-art systems “or that its invention addresses a need for more security.” POR, 42-43. As I discussed above (§VI.A), I have been informed that under the field-of-end endeavor inquiry, the problems being addressed are irrelevant because the field-of-end endeavor test does not look to the problem that the patent purports to address.

77. But even if the problem(s) being addresses mattered, in my opinion, POSAs would understand that SFI is incorrect that Hayhurst did not recognize a need for more security in the prior art. Hayhurst explicitly says its “approach” to storing and delivering data “provides **increased security**.” Hayhurst, [0021]. By recognizing that its technique provides increased security, Hayhurst was clearly recognizing and telling POSAs that Hayhurst believed there was a **need** for that “increased security” that Hayhurst sought to provide. As I’ve discussed above, Hayhurst explicitly teaches (several times) that its storage technique is designed to meet that need for increased security (including for already encrypted data).

78. As I discussed above, the ’116 patent and Hayhurst describe overlapping techniques for securing data, which the ’116 patent itself recognizes is a broad field.

The secure data parser of the present invention **may be integrated into any suitable existing system** in order to provide the use of its functionality in each system's respective environment. FIG. 27 shows a block diagram of an illustrative system 2600, which may include software, hardware, or both **for implementing any suitable application**. System 2600 may be an existing system in which secure data parser 2602 **may be retrofitted as an integrated component**. Alternatively, secure data parser 2602 **may be integrated into any suitable system** 2600 from, for example, its earliest design stage. Secure data parser 2600 **may be integrated at any suitable level of system** 2600.... Secure data parser 2602 may be used for parsing and

splitting data among one or more storage devices 2604 in accordance with the present invention. Some illustrative examples of systems having the secure data parser integrated therein are discussed below.

EX1001, 67:48-67.

79. I note again that Hayhurst **expressly** says its Technical Field “relates to” the “**secure** handling and storage” of data ([0002]) and Hayhurst achieves that secure handling of data using a technique the ’116 patent **itself** says secures data (including already-encrypted data), *i.e.*, distributed data storage.

80. Thus, in my opinion, POSAs would find that Hayhurst is analogous art to the ’116 patent at least because it is plainly within the field of securing data, which is the field of endeavor that SFI has identified for its ’116 patent.

2. POSAs would find that Hayhurst is reasonably pertinent to at least one of the problems to which the ’116 patent relates.

81. Even if Hayhurst were (somehow) found not to be in the same field of endeavor as the ’116 patent, in my opinion POSAs would find that Hayhurst is also reasonably pertinent to one or more of the problems to which the ’116 patent relates.

a. Hayhurst is reasonably pertinent to the problem of user-independent security.

82. SFI argues that the problem to which the ’116 patent relates is providing a “cryptographic system whose security is user independent.” POR, 44-45 (quoting EX2033, ¶112). SFI does not explain what it means by “user

independent” security. At deposition, Dr. Rubin opined that user-independent security means “[t]he data is secured in a way that is not dependent on a particular user; it’s dependent on the data.” Rubin-Depo., 115-116. Dr. Rubin identified an example of user-independent security in the patent is its disclosure of “encrypting the data, splitting it into shares,” which Dr. Rubin said “is not going to depend on a particular user who’s going to be using the data.” Rubin-Depo., 116. Dr. Rubin also said, for example, user-independent security includes when “no user will have enough of the data to be able to obtain the original data.” Rubin-Depo., 116. Dr. Rubin further testified that an example of that includes when “the data is stored such that in no user, at any particular location where a piece of the data is, can reconstruct the whole data.” Rubin-Depo., 115-120. As I discuss below, POSAs would recognize that all of those examples of user-independent security are found in Hayhurst.

83. In particular, Hayhurst’s distributed-storage technique secures data in a user-independent manner because Hayhurst segments and distributes data without requiring user action or user knowledge for either the storing or recovering of the secured data. For example, Hayhurst discloses a “main server 26” that “distributes [] the movie segments...to all of the participating subscribers units [].” Hayhurst, [0027]. Hayhurst continues that “[s]uch distribution and storage of the segments on the units may be performed in a manner that is **automatic and**

transparent to the users of the units.” Hayhurst, [0027]. That is, the users play no part in the segmenting, distribution, or storage of Hayhurst’s segments (beyond having a storage device that is connected to the necessary network). Users do not know what segments are being stored, what other users have segments for the same data set (or what segments any individual has), or how any segment has been encrypted. Users do not select the encryption keys or encryption methods, nor is an encryption key ever revealed to a user. Users do not know how a file was segmented or how many segments exist for any given file. Users also cannot undermine the security that Hayhurst’s system provides (and thus the security is not dependent upon users) because no one subscriber unit has more than one segment, and a single segment has little value on its own. Hayhurst, [0021] (disclosing that “each subscriber...will store only very small portions of the movie on the[ir] storage device” and thus “decoding of any individual bundle will be virtually impossible”), [0029] (explaining that the main server has to “authoriz[e]” users to recover the data).

84. Hayhurst further teaches that additional security is provided because “access to the correct method for recombining the segments to recreate the original media file could be stored in a password-protected file, linked to an encryption key, or locked to all devices **except the specific hardware device that created the bundled segments.**” Hayhurst, [0021]. That further reinforces that the

security provided by Hayhurst's technique is user-independent because users have no idea how to recombine segments even if they were given all of the segments (and even then they would need to be given the encryption key(s) that are not dependent on, selected by, or revealed to users).

85. Thus, in my opinion, POSAs would have considered Hayhurst when considering the problem of providing a cryptographic system whose security is user independent, which is the only problem SFI identifies in alleging Hayhurst is not analogous art.

86. In discussing the above problem that the inventors were allegedly involved with, SFI's POR quotes Dr. Rubin's declaration, which, in turn, quotes the patent alleging that its disclosure includes "provid[ing] a cryptographic system whose security is user-independent while still supporting mobile users." POR, 44-45 (quoting EX2033, ¶112 (quoting EX1001, 1:31-40, 1:54-2:38). If SFI and Dr. Rubin are arguing that the '116 patent's claims are limited to or require supporting mobile users, then that is a new claim construction that no party had previously identified. In my opinion, POSAs would also disagree with that construction because there is no indication in the plain language of the claims or intrinsic evidence that would limit and/or require mobile devices.

87. I note too that at deposition Dr. Rubin admitted that the independent claims of the '854 patent do not refer to a mobile user, that it was not his opinion

that the disclosed system or independent claims required mobile users (because he said he “ha[s]n’t studied that” and “it’s not an opinion that’s in [his] report”), and he further testified that he “ha[d] not looked at th[e] question” of whether a system without mobile users is outside the scope of claim 1 of the ’854 patent (which SFI similarly alleges addresses the same problem as the ’116 patent). Rubin-Depo., 115-120. Indeed, Dr. Rubin testified that he was just “quoting from the summary of the invention, but [he] didn’t elaborate or give it further thought as to what the supporting mobile user – how that might relate to the claims or anything like that.” Rubin-Depo., 119-120. I have given that some thought, and I disagree that POSAs would think that supporting mobile users was relevant to a problem the patent was allegedly trying to address. At a minimum, to the extent SFI has correctly identified user-independent security as a problem that the ’116 patent concerns itself with, POSAs would recognize that that would be a consideration with or without the need to support mobile users.

88. In any event, in my opinion, POSAs would understand that Hayhurst **does** support mobile users because POSAs knew that distributed/peer-to-peer storage techniques (like Hayhurst’s) also supported and were beneficial to mobile users. In response to SFI’s and Dr. Rubin’s arguments otherwise, my opinion that Hayhurst’s technique would support mobile users is corroborated by EX1091, [0010], which discloses a “peer-to-peer (p2p) computing model [that] lends itself

to the dynamic environments of **wireless devices** including, but not limited to, **cell phones, pagers, PDAs, and other consumer devices.**”

89. My opinion is further corroborated by prior art papers in the field of securing data that I co-authored myself that discussed using distributed storage and/or distribution networks like Hayhurst’s that supported mobile users. *See, e.g.:*

- Bobby Bhattacharjee, Mostafa Ammar, Ellen Zegura, Viren Shah, and Zongming Fei. “Application-Layer Anycasting”, In Proceedings of INFOCOM’97, pages 1388–1396, Kobe, Japan, 1997.
- Zongming Fei, Bobby Bhattacharjee, Ellen W. Zegura, and Mostafa Ammar. “A Novel Server Selection Technique for Improving the Response Time of a Replicated Service”, In Proceedings of INFOCOM’98, pages 783–791, San Francisco, CA, 1998.
- Ellen W. Zegura, Mostafa Ammar, Zongming Fei, and Samrat Bhattacharjee. “Application-Layer Anycasting: A Server Selection Architecture and Use in Replicated Web Service”, Transactions on Networking, Vol. 8, Issue 4, pages 455-466, August 2000.

90. Moreover, as I discussed above (§VI.A), I understand that a prior art reference can be analogous art if it is reasonably pertinent to any **one** of the problems addressed by the patent’s inventors. Though SFI’s allegation that Hayhurst is not analogous art only identifies one problem allegedly addressed by the ’116 patent, in my opinion POSAs would understand that the ’116 patent

identifies other alleged problems that were allegedly being addressed by the patent. In my opinion, POSAs would have found that Hayhurst is also reasonably pertinent to several of those other problems the '116 patent discusses.

b. Hayhurst is reasonably pertinent to the problem of computer-network security.

91. The '116 patent expresses concern with the problem of how to “ensur[e] [that] data stored and transmitted over” “computer networks” “cannot be read or otherwise compromised.” EX1001, 1:26-33. *See also* Rubin-Depo., 109-110 (testifying that “ensuring data stored and transmitted over these systems cannot be read or otherwise compromised” was “part of the problem that the invention solves”).

92. In my opinion, Hayhurst is reasonably pertinent to that problem because Hayhurst too “relates to the field of **electronic media delivery**...including the **secure handling and storage** of” data ([0002]) and “deter[ing] [] piracy” (i.e., unauthorized access) ([0008]). *See also* Hayhurst, [0016] (describing various computer networks in which Hayhurst’s technique can be integrated including a “cable television (CATV) network,” or “any other communications network of users or subscribers, such as those linked by a local area network (LAN) or a wide area network (WAN)”). That is, Hayhurst is also addressing the problem of how to ensure that data stored and transmitted over a computer network maintains

“increased security from piracy” (*i.e.*, unauthorized reading or comprising of security).

93. Thus, in my opinion, POSAs would have considered Hayhurst as reasonably pertinent to the problem the ’116 patent identifies of ensuring that data stored and transmitted over computer networks cannot be read or otherwise compromised.

c. Hayhurst is reasonably pertinent to problems addressed by distributed-storage (e.g., the problem of data remaining whole and subject to attack).

94. The ’116 patent also discusses problems that occur with data “remain[ing] whole and intact and subject to attack” (EX1001, 57:10-13) and how to “protect[...against compromise of any individual data storage facility” (EX1001, 5:18-27).

95. While the section of SFI’s POR that alleges Hayhurst is not analogous art focuses only on the alleged problem of providing user-independent security, elsewhere in its POR, SFI quotes Dr. Rubin’s opinion that the patent explains that “when ‘data ... remains whole and intact,’ it is ‘subject to attack’” and that “[t]he ’116 addresses this concern, in part, by ‘performing a cryptographic parsing and splitting of the encrypted file’” and thus “teaches that splitting the data set into shares helps secure the data.” POR, 21-22 (quoting EX2033, ¶¶69-70 (quoting EX1001, 57:10-24)). Thus, SFI recognizes that the ’116 patent also alleges there

were problems arising from secured data remaining whole and intact and subject to attack. SFI and Dr. Rubin do not discuss whether Hayhurst is reasonably pertinent to that problem, let alone allege and demonstrate that Hayhurst is not reasonably pertinent to that problem. As I discuss further below, POSAs would find that Hayhurst is reasonably pertinent to that problem as well.

96. At deposition, Dr. Rubin testified that SFI's patents address a "concern" of "when data remains whole or intact it's subject to attack" because, in that case, when "an adversary breaks into the site where the data is, then all the data's there." Rubin-Depo., 124-126. The '116 addresses those concerns / problems by splitting and "storing cryptographic data in geographically remote secure data storage facilities" (i.e., distributed storage facilities). EX1001, 5:48-6:9. That is, the '116 patent addresses an alleged problem that arose in prior art techniques that kept encrypted data together in one piece and stored in one place. The patent addresses that problem by breaking the data apart (before or after encrypting) and storing the pieces (with or without individually encrypting the pieces) on different storage devices.

97. I note that Dr. Rubin provided deposition testimony that is consistent with my opinion regarding the patent's purported solution to that problem. Rubin-Depo., 126-128 (testifying that the patent addresses the concern of data remaining whole and intact by "put[ting] one of those shares, for example, in one place and

another share in another place” “[s]o now the data is not whole or intact, so if an attacker comprises [sic: compromises] one of the sites, but not the other sites, then they haven’t gotten the data”); Rubin-Depo., 129-130 (by having one piece in one location and the other piece in the other location “[t]hat makes it harder for an attacker to get the data”); Rubin-Depo, 130-132 (“I’m not offering an opinion at this point the case on th[e] question” of whether Claim 1 requires cryptographic splitting of the data); Rubin-Depo., 133-135 (testifying that the objective of preventing an employee from obtaining more than one portion of a data set is met by having “geographically remote” storage devices so that “a physical compromise of a system would only yield the portion that’s there, but then you would need the other portion at the other remote location to actually get any useful data”).

98. In my opinion, POASs would find that Hayhurst is not only reasonably pertinent to those same problems discussed above but solves them the same way the ’116 patent alleges it addresses those problems: via segmenting encrypted data and storing the segments on distributed storage devices, as I discussed above (§VI.A.1). Hayhurst, [0021].

99. Indeed, claim 1 requires “*storing*” portions of data “*on a plurality of different storage devices*,” which is one of the teachings from Hayhurst that I opined POSAs would have used in the Foster-Hayhurst combination. By claiming as a limitation, the very thing that Hayhurst teaches, it reinforces to POSAs that

Hayhurst is reasonably pertinent to one of the problems discussed in the '116 patent and, in fact, solves that problem in the same way. POSAs would plainly find that a prior art reference that teaches segmenting data and storing it on distributed storage devices for the purposes of increasing security, like Hayhurst (*e.g.*, [0008]), was reasonably pertinent to a patent whose claims require storing segments of data on multiple storage devices and whose specification alleges that that segmenting and distributed storage was done for the purposes of increasing security.

100. Thus, in my opinion, POSAs would have also considered Hayhurst reasonably pertinent to the problem the '116 patent identifies of security concerns when data remains whole and intact and subject to attack and the related problem of preventing an overall failure of security for a data file even if one (or some subset) of the data storage facilities become insecure and/or fail.

101. Thus, for the reasons I discussed above in this section, it is my opinion that Hayhurst is also analogous art because POSAs would understand that Hayhurst is reasonably pertinent to at least one (and in my opinion several) of the problems to which the '116 patent's inventors were involved.

B. Response to Argument that POSAs Would Not Have Had Motivation For the Foster-Hayhurst Combination

102. My First Declaration provided my opinions (backed by citations to Foster, Hayhurst, and several corroborating references) that POSAs would have been motivated to pursue the Foster-Hayhurst combination that I described in my

First Declaration with a reasonable expectation of success because POSAs would have recognized that implementing Foster’s system using Hayhurst’s distributed-storage technique would have provided **numerous** benefits for Foster’s system.

First Declaration, ¶¶241-246; *see also id.* ¶246 (explaining my opinion that POSAs would have found that Foster-Hayhurst “would have also been nothing more than the obvious application of Hayhurst’s known distributed storage technique to Foster’s system, which was ready for the improvement”).

103. Any one of the benefits that I identified (summarized again below) would have motivated POSAs to pursue the Foster-Hayhurst combination, as I corroborated with evidence from Hayhurst itself and from POSAs’ background knowledge about the benefits of a distributed storage and delivery system, like Hayhurst’s:

- “shifting the bandwidth requirement in delivery digital content aware from the content provider” (citing Hayhurst, [0005]);
- “provid[ing] a high level of security and a deterrence to piracy” (quoting Hayhurst, [0008]; citing also, EX1030, [0005]-[0010]);
- “support[ing] redundancy in storage” (citing Hayhurst, [0009]); and
- “benefits in bandwidth and cost, security, and redundancy (also referred to as ‘fault tolerance’),” “aggregate speed,” “efficiency / ‘intelligence’”.

EX1003, ¶¶241-246 (citing corroboration).

104. In its POR, SFI argues that POSAs would not have sought to achieve any of the benefits that I identified because, according to SFI and Dr. Rubin, using Hayhurst’s distributed-storage technique in Foster’s system would have allegedly been incompatible with Foster’s goal of “avoiding delays” to improve “streaming live television.” POR, 47-48. Dr. Rubin similarly asserts that the combination “would introduce enormous delays...rendering Foster’s system unsuitable for streaming live television.” EX2033, ¶149. Dr. Rubin does not explain what he meant by “enormous delays,” nor how he (let alone a POSA) would quantify “enormous delays,” nor explain why there would have been “enormous delays,” nor did Dr. Rubin cite any evidence in support of that assertion that there would have been enormous delays. (Dr. Rubin’s opinion in ¶149 is also undermined directly by his immediately preceding paragraph ¶148 because in ¶148 he recognizes that Foster **also** includes “content **stored** on a DVD” and thus includes embodiments that are not about streaming live television, as I discuss further below.)

105. In my opinion, based on my experience and research in this space, I do not agree and POSAs would not agree that there would be “enormous delays” over Foster. *See supra* §VI.B.2 (discussing how Foster-Hayhurst would reduce delays over Foster alone).

106. For one, I note that the '116 patent itself recognizes that sending data via “multiple parallel streams” (as would have occurred in Hayhurst-Foster but not Foster alone) provides for “**increasing speed and efficiency.**” EX1001, 70:10-12. Benefits of increased speed and efficiency are **exactly** two of the benefits that I opined would have motivated POSAs to pursue the Foster-Hayhurst combination, which I corroborated with independent prior art (*i.e.*, my opinion was not based on hindsight from the '116 patent's teachings).

107. Below is a list of several, peer-reviewed articles and/or papers relating to presentations given that relate to downloading data from replicated sources and the benefits thereof. Each of these papers were published and publicly available in the prior art that I was a co-author on that are in the field of securing data and which POSAs would have been familiar with:

- Bobby Bhattacharjee, Mostafa Ammar, Ellen Zegura, Viren Shah, and Zongming Fei. “Application-Layer Anycasting”, In Proceedings of INFOCOM'97, pages 1388–1396, Kobe, Japan, 1997.
- Zongming Fei, Bobby Bhattacharjee, Ellen W. Zegura, and Mostafa Ammar. “A Novel Server Selection Technique for Improving the Response Time of a Replicated Service”, In Proceedings of INFOCOM'98, pages 783–791, San Francisco, CA, 1998.
- Ellen W. Zegura, Mostafa Ammar, Zongming Fei, and Samrat Bhattacharjee. “Application-Layer Anycasting: A Server Selection

Architecture and Use in Replicated Web Service”, Transactions on Networking, Vol. 8, Issue 4, pages 455-466, August 2000.

- Suman Banerjee, Seungjoon Lee, Bobby Bhattacharjee, and Aravind Srinivasan. “Scalable Resilient Multicast”, Proceedings of ACM SIGMETRICS, pages 102–113, 2003.

108. Below is a list of several, peer-reviewed articles and/or papers relating to presentations given that relate to large-scale streaming of data (including live streaming thereof) from multiple sources and the benefits thereof. Each of these papers were published and publicly available in the prior art that I was a co-author on that are in the field of securing data and which POSAs would have been familiar with:

- Suman Banerjee and Samrat Bhattacharjee. “Scalable Application-Layer Multicast for Content Distribution”, Computer Communications Workshop, 2001.
- Suman Banerjee, Bobby Bhattacharjee, and Christopher Kommareddy. “Scalable Application- Layer Multicast”, Proceedings of ACM SIGCOMM, pages 205–217, 2002.
- Suman Banerjee, Christopher Kommareddy, Koushik Kar, Bobby Bhattacharjee, and Samir Khuller. “Construction of an Efficient Overlay Multicast Infrastructure for Real-time Applications”, Proceedings of IEEE Infocom, pages 1521–1531, 2003.

- Suman Banerjee, Seungjoon Lee, Bobby Bhattacharjee, and Aravind Srinivasan. “Scalable Resilient Multicast”, Proceedings of ACM SIGMETRICS, pages 102–113, 2003.
- Suman Banerjee, Seungjoon Lee, Ryan Braud, Bobby Bhattacharjee, and Aravind Srinivasan. “Scalable Resilient Media Streaming”, In Proceedings of ACM NOSSDAV’04, pages 4–9, 2004.
- Suman Banerjee, Christopher Kommareddy, Koushik Kar, Bobby Bhattacharjee, and Samir Khuller. “OMNI: An Efficient Overlay Multicast Infrastructure for Real-time Applications”, Special Issue of Computer Networks on Overlay Distribution Structures and their Applications, Vol 50:6, pages 826-842, 2005.
- Suman Banerjee, Seungjoon Lee, Bobby Bhattacharjee, and Aravind Srinivasan. “Resilient Multicast using Overlays”, IEEE/ACM Transactions of Networking, Vol. 14, No. 2, pages 237–248, April 2006.

109. As I discuss further below (§§VI.B.1-VI.B.2), POSAs would not agree with SFI’s argument (at least) because SFI’s argument is premised on an interpretation of Foster’s disclosure being limited to “streaming live television.” However, POSAs understand that Foster’s disclosure is relevant to much more than that, including the **storage** of Foster’s encrypted content (both before and after delivering to end-users, whether via streaming or otherwise).

110. POSAs would have understood that reducing latency (and thus improving the delivery of streaming of live television) is **one** of the benefits Foster

says its system and encryption technique can provide, in some use cases. But as I discuss further below (§VI.B.1), Foster’s teachings are not limited to that exemplary benefit of reducing latency, let alone limited to the context of live streaming of television.

111. POSAs would read Foster for all that it teaches. As I discussed in my First Declaration (¶¶221-224, ¶¶252-253, ¶264) and further below (§§VI.B.1-VI.B.2), Foster teaches embodiments and implementations where Foster’s encrypted content is not streamed live but rather **stored** for and by recipient(s) for later playback (*e.g.*, a movie downloaded from a video-on-demand server). That is a context explicitly taught by Foster and one that is relied upon in my First Declaration and reinforced by POSAs’ background knowledge about the systems and content discussed in Foster. It is in (at least) that storage context that POSAs would have recognized that Hayhurst’s distributed-storage technique would have offered benefits to both the storage and delivery of Foster’s encrypted content. In my opinion, once Foster’s teachings are correctly read as not being limited to live streaming, SFI’s arguments against POSAs’ motivation-to-combine Foster and Hayhurst fall apart because they depend on an incorrectly confining Foster to streaming live broadcasts, which directly contradicts Foster’s broader teachings that include storing. (In fact, as I discuss further below, §VI.B.1, POSAs would

understand that even Foster’s streaming context typically includes storage, both before and after delivery to end users.)

1. It is clear to POSAs from Foster’s disclosure that Foster stores encrypted content.

112. As I discussed in my First Declaration (¶¶213-214), Foster describes benefits to attaching an encryption key to the data that that encryption key encrypts, which Foster describes that as “synchronization” (because the data and keys can be stored and delivered at the same time). Foster, [0002]-[0007]. Foster explains that attaching an encryption key to the data that that encryption key encrypts provides several benefits. For example, by keeping the encryption key with the *share* of Foster’s data that it encrypted, the risk of applying the wrong key to the wrong share of data is reduced (because the system would not have to “search” for the correct key, such as in systems where keys are stored and/or delivered separately from the data they encrypt). Foster, [0002]-[0007]. Placing the encryption key with the *share* of Foster’s data that it encrypts **also** reduces the time it takes to locate the correct key for decrypting a portion of the dataset (i.e., reduces latency from having to locate the correct key), thereby facilitating “random access” to portions of “protected content.” Foster, [0002]-[0007]. Foster is clear (and POSAs would have recognized) that the benefit of reducing latency for random access to content is **not** limited to streaming content, let alone streaming “live” content. Instead, as I explained in my First Declaration (¶¶213-214),

Foster's technique allows for random access to stored content (e.g., movie) at a particular ("random") *chunk*: for example, starting a stored movie at any point (say the middle where your favorite scene is), rather than have to start the movie at the beginning every time. Foster, [0002]-[0007]. Foster calls the technique of attaching an encryption key with the encrypted data "synchroniz[ation]." Foster, [0002].

113. In its POR, SFI argues (and Dr. Rubin opines) that Foster's "fundamental goal" is to use "synchronization" to "reduc[e] delays in viewer access to content **as it is transmitted**" in real-time. POR, 49-50 (citing EX2033, ¶120). In my opinion, POSAs would disagree that reducing delay in accessing real-time content as it is transmitted is the **fundamental** goal of synchronization in Foster and to the extent SFI is arguing that that is the **only** goal of synchronization in Foster, POSAs would disagree in my opinion. Using synchronization for the benefit of streaming live content is **a** benefit in Foster, when Foster is applied to streaming live content (but that is just one of the applications Foster discusses). For example, Foster certainly teaches "synchronized transmission to...a receiver." Foster, [0008]. But in that very same sentence, Foster **also** teaches attaching an encryption key to encrypted data to beneficially provide "for synchronized...**storage** by a receiver." Foster, [0008].

114. That is, Foster is **explicit** that synchronization can be used to benefit transmission **or storage** of content (or both in combination). Synchronization

benefits apply to applies to **stored** data as well because, for example, as I explained above, it supports random access to a movie at a random point in the movie, rather than have to start the movie at the beginning and wait for the movie to reach the portion you're interested in viewing.

115. My opinion that Foster's synchronization is not limited to live-streaming applications is reinforced by Foster repeatedly saying synchronization is useful for either data "transmission to, **or storage by**, a recipient." Foster, Abstract; *see also id.* [0007]-[0008], [0048], [0064], [0079], claim 11. Indeed, I understand that in the IPR of the '609 patent (which I understand is IPR2024-00214 and for which I am also submitting a reply declaration), SFI **admits** that Foster "**teaches**...the 'recipient 430 can....**store** the encrypted content.'" 609-POR, 44 (quoting Foster, [0079]), 51 (similar). At deposition, Dr. Rubin acknowledged that, in Foster, "[t]he recipient...could decide to **store** it [i.e., received content]." Rubin-Depo., 191. As I noted above, in EX2033 ¶148, even Dr. Rubin admits that "Foster's system focuses upon streamed content [] or **content stored** on a DVD."

116. Thus, it is readily apparent to POSAs and made clear by multiple disclosures in Foster that Foster includes processing a data set for the purposes of **storing** a Foster-encrypted version of the data set.

117. As demonstrated by the following quote from Foster, Foster is even explicit that synchronization reduces latency for streaming purposes **and** storage-based purposes, as would be used for the Digital Rights Management Systems (“DRMS”) implementations discussed in Foster:

[A] recipient will receive or store **encrypted content** along with all information necessary to decrypt the content. This not only **prevents the synchronization problems** associated with previous systems, **but also provides random access of content**. As indicated above, both DRMS and CAS have inherent latencies (e.g., delays) in providing random access of content. Specifically, because each content packet transmitted under the present invention is “packaged” with the title key that was used to encrypt it, a recipient can access and decrypt the content stream at any packet.

Foster, [0064].

118. As Foster explains, in Foster’s DRMS implementations, “content is packaged and loaded [i.e., stored] onto a web server” and subsequently “**downloaded to** [i.e., stored at a] personal computer” for on-demand accessing at “rendering time” (i.e., when the user wants to view stored content at some indefinite time after downloading/storing). Foster, [0005], [0049] (explaining how users access “content on media server,” which POSAs understood is **storing** the content before it has been purchased so it could be downloaded by multiple users at various times), FIG. 1 (content is accessed for “streaming **or stored**”).

119. Foster at [0050] says that “the content **can** be delivered through streaming delivery (e.g., if the file is large).” Foster says it “can” be delivered via

streaming “**if**” the file is large because the file would not **have** to be delivered via streaming (e.g., if the file were not too large or when the storage devices had sufficient memory to store the file). POSAs would understand that one of the things that Foster at [0050] is suggesting is that if someone is downloading a large file (e.g., a lengthy, high-definition movie) it may take a long time to download and store the entire file. Thus, rather than wait for the entire file to be downloaded before viewing it, a user could start viewing the content via streaming while it is also being downloaded and stored (e.g., because it was purchased, or could be viewed again over the course of several days, or to support interactive television, like pausing and rewinding, etc.).

120. In response to SFI’s and Dr. Rubin’s arguments that Foster’s synchronization is only relevant to streaming live television, my opinion that POSAs would have understood Foster’s DRMS embodiments to be for the purposes of **storing** content is also corroborated by SFI’s EX2039, which recognizes that DRMS “safeguards content when **at rest** [i.e., stored] in addition to the transport phase,” and extends to content “**stored** on a receiving device” and for “offline playback,” *i.e.*, not just for live streaming (i.e., “offline”). EX2039, 6-10. POSAs would have had the same understanding of DRMS in the prior art.

121. Consistent with Foster’s DRMS disclosures, Foster also teaches that its payloads can be processed according to the MPEG-standard’s so-called

“program stream,” which Foster says (and POSAs knew) is intended for use “**in fixed storage.**” Foster, [0073]. My opinion that POSAs knew as part of their background knowledge that an MPEG program stream is designed and intended for storage is supported by the following corroborating references.

122. SFI’s EX2014, for example, acknowledges that the MPEG “[p]rogram streams are **designed for...storage.**” EX2014, 1.

123. As another example I cite in response to SFI’s arguments and Dr. Rubin’s opinions, EX1108, teaches that **both “[t]he program and transport streams** may be recorded on [i.e., stored on] or played back from a digital video disc (DVD), video tape, magnetic or optical disk drive or other suitable storage device” (EX1108, 1:65-2:31), which are all examples of storage devices in the ’116 patent (EX1001, 67:7-12, 67:30-47, 70:36-48). EX1108 further discloses that it was known that even with the MPEG transport stream it was conventional that “a prior art video server may interact with a peripheral **such as a storage device** in a television set-top box **to download a transport stream corresponding to a video-on-demand selection.**” EX1108, 1:65-2:31. EX1108 is thus consistent with my opinion in my First Declaration (and discussed further below) that Foster is consistent with a video-on-demand (VOD) system, where encrypted content is both stored at the provider (e.g., a cable company and/or movie studio) **and** at the recipient (e.g., in a cable customer’s set-

top box). *See also* Foster, [0052] (disclosing that content is “delivered to receiver 28 (e.g., a set-top box)”).

124. As another example I cite in response to SFI’s arguments and Dr. Rubin’s opinions, EX1109 (an IEEE article from 1995) teaches that the “MPEG-2 System layer describes how elementary streams comprising one or more programmes are multiplexed together to form a single data stream suitable for digital transmission **or storage**.” EX1109, 1.² EX1109 further taught that “[t]he programme stream ... is **intended for the storage** and retrieval of programme materials from **digital storage media**.” EX1109, 2; *id.*, 3-9 (explaining how “data bytes [are] taken sequentially from the original elementary stream” to form a

² The IEEE Xplore website indicates that EX1109 was published in “IEE Colloquium on MPEG-2 - What it is and What it isn’t,” in connection with a conference on “24 January 1995”; IEEE Xplore also indicates that EX1109 was added to IEEE Xplore on “06 August 2002.”

<https://ieeexplore.ieee.org/document/478032>. Based on the above evidence and my review of the indicia appearing on EX1109 and my understanding of the IEEE and IEEE Xplore, it is my opinion that EX1109 was publicly accessible in the prior art by being disseminated or otherwise made available to POSAs exercising reasonable diligence by February 1995, or at least by August 2002.

“PES-packet”), 13 (“The ‘programme stream’ is biased **towards the storage** and replay of a single programme **from a digital storage device....**”).

125. As another example I cite in response to SFI’s arguments and Dr. Rubin’s opinions, EX1072 discloses as background information that content distributors (e.g., cable and satellite companies) had been developing and releasing “[i]nteractive television (iTV)” that provided, among other features, “2) personal video recorders (PVRs) which enable users to record, store, pause, and playback programming using hard-drive storage mechanisms, as opposed to conventional video tape, 3) video on demand (VOD) which enables viewers to order any type of content on demand, such as movies, television shows, special events, sports programs, and other content, and to control its delivery on their receiving devices by pausing, rewinding, or fast-forwarding them.” EX1072, [0002]-[0003].

EX1072 further recognized as background information, that the “content distributor’s access control system, often referred to as a Conditional Access system” would interact with “a user’s set-top box” having “the appropriate middleware software that is compatible with the iTV applications being deployed at the content distributor's head-end.” EX1072, [0004].

126. EX1072 further discusses as background information that it was “desirable to have methods and systems that allow **a service provider to devise, deploy and manage ... video-on-demand service** packages.” EX1072, [0011]. In

EX1072's example, prior-art system, included numerous places where content would be stored – both before delivery and after delivery to customers:

[0042] ... The system operates with a **built-in storage device in the client device for storing the programs, for instance, a hard disk drive or any other storage media or by storing programs on a server at the head-end**. Exemplary personal video recorders are designed by companies such as TiVo, Inc. and manufactured by companies such as Sony Corporation and U.S. Phillips, Inc.

[0043] Video on demand (VOD) 345 enables the subscribers to select a particular content type from a list of **available content stored on a plurality of servers at the head-end to be viewed by the subscribers**. The selected content type could be television programs, movies, events, sporting events, music or **some other form of stored media** or real-time encoded broadcast. The VOD servers send the requested content to the subscriber over a delivery mechanism such as cable, satellite, or other transmission infrastructure. The client device may have **some storage media in which the delivered content can be stored** and viewed by the users. The user may also be provided with the transport control functions such as controlling the delivered video. This may include pause, fast forward, and so forth. **The transport control functions are typically supported at the headend where the content is located/stored**.

127. Accordingly, that Foster's teachings include storage-based embodiments is readily apparent from Foster's text, as explicitly stated in Foster

and POSAs would independently understand from their background knowledge of the DRMS implementations Foster explicitly identifies for loading (storing) encrypted content to web servers for download and storing by recipients.

128. Finally, even in the implementations of Foster that are for streaming content for use in a conditional access system (“CAS”), those systems do not only deliver “live” broadcasts. And to the extent SFI meant real-time broadcasts – like a live sporting event – POSAs would know that a percentage (if not a majority) of television at the time was not streaming live events, but rather streamed from content that has been pre-recorded and stored by the content provider (e.g., a television sitcom, a movie, a documentary, etc.).

129. In response to SFI’s arguments and Dr. Rubin’s opinions regarding CAS in Foster, POSAs knew that CAS is used in video-on-demand (VOD) systems, where “content **is stored pre-encrypted** on hard drives in the VOD server.” EX1110, [0008]. POSAs similarly knew that the content delivered by VOD was commonly stored again at a “client device[’s]” or in “**storage** media in which the **delivered content can be stored**” (EX1072, [0043]).

130. In response to SFI’s arguments and Dr. Rubin’s opinions regarding the use of a CAS in Foster, my opinion that POSAs understood that Foster’s CAS embodiments were broader than “live” streaming and also relevant to the **storing**

of encrypted content is corroborated by the following references (along with the references I discussed above).

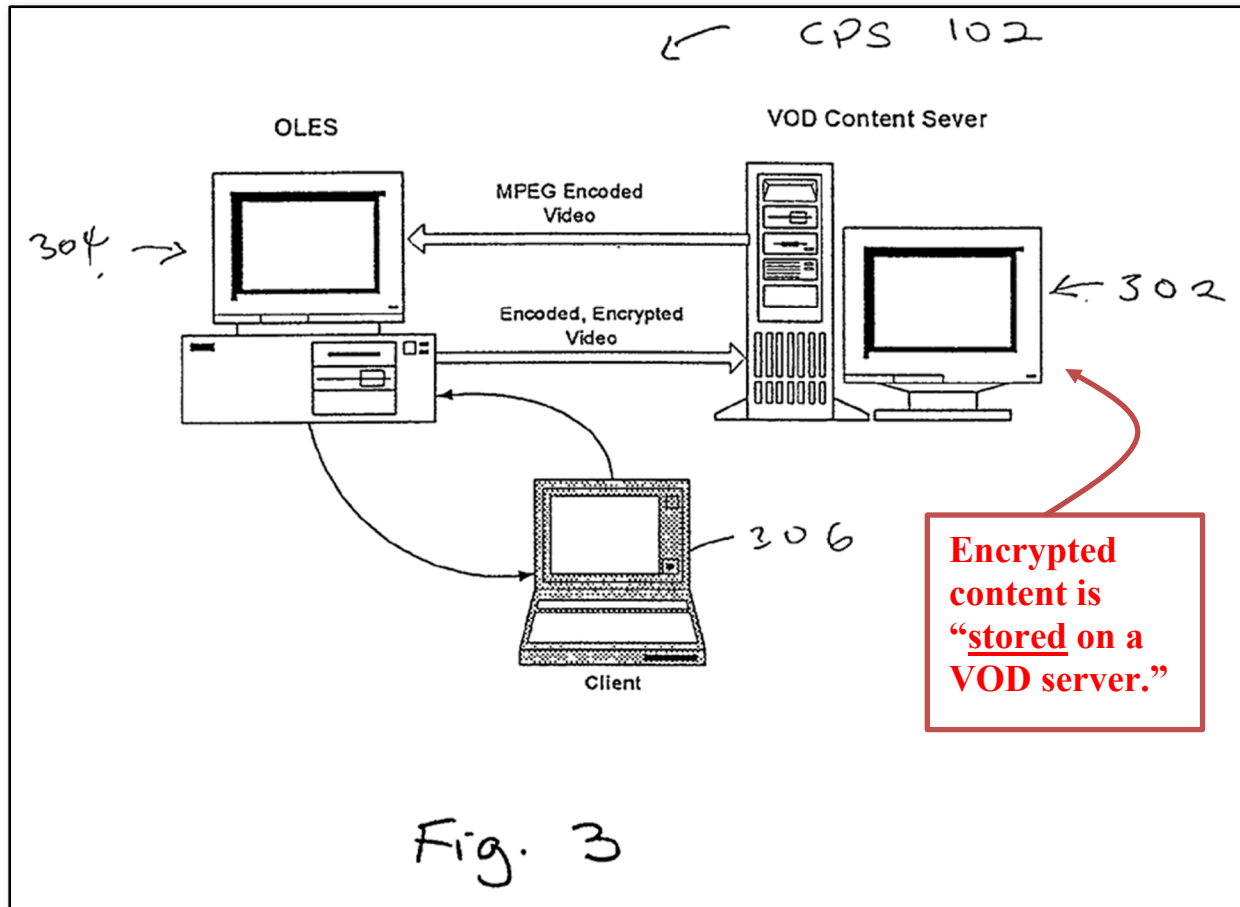
131. For one, Hayhurst itself recognizes that in “deliver-on-demand,” such as for “cable television (CATV) system (or for other content provider[s]),” it was desirable for the end-user consumer (e.g., the user that has purchased and received an encrypted movie file), to “assemble[the segments] into the complete media file (movie) for display, **or stored for later display**, on the requesting subscriber's unit.” Hayhurst, [0004]-[0006].

132. As another example in response to SFI’s arguments and Dr. Rubin’s opinions, I cite EX1028 (cited First Declaration, ¶243). EX1028 discloses as background information that “[v]ideo on demand, such as through cable and satellite providers” include “head end hardware” ([0002]), which is where content is stored from VOD. EX1028 shows “a traditional client/server type of media-on-demand network,” where the central server “may be a regional server or local hub” that “delivers the content to the requesting user.” EX1028, [0015]. EX1028 further notes that the “content [is] available from peers and centralized storage.” EX1028, [0018]. *See also* EX1028, [0043] (noting that “[t]he user may store the content on the device for any period of time the user desires”).

133. As another example in response to SFI’s arguments and Dr. Rubin’s opinions, I cite EX1031 (cited First Declaration, ¶243). EX1031 notes as

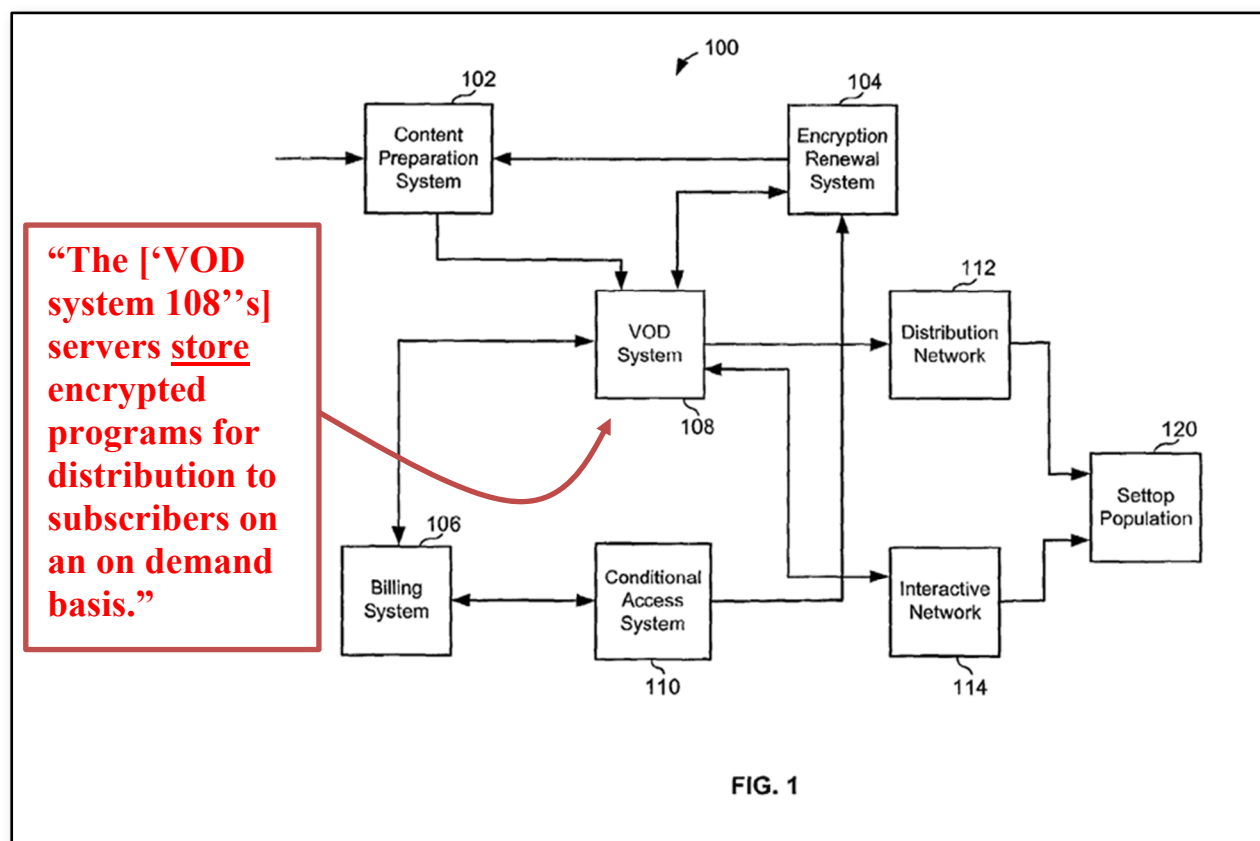
background information that the prior art includes “Content Delivery Networks,” that POSAs knew where systems having servers that stored content to support the delivery of content-service-provider content. EX1031, [0005]. EX1031 at [0018]-[0025] notes benefit of “transfer[ring] by downloading data from multiple sources or channels in parallel” (as in Hayhurst). One of the (several) benefits of those delivery systems include the support for “Streaming” where “one can actually watch a video as it is being downloaded from multiple sources in parallel,” such as for “Video-on-Demand.” EX1031, [0022].

134. As another example I cite in response to SFI’s arguments and Dr. Rubin’s opinions, EX1073 at [0001]-[0004] corroborates by opinion that POSAs knew that “conditional access” systems were used for VOD system. EX1073 at [0043] further corroborates that it was known that VOD systems had “pre-encrypt[ed]” content and “**stores** the content for forwarding...when authorized” for delivery to a storage device. *See also* EX1073, [0098] (“The encoded file is encrypted and verified prior to writing the encrypted material to VOD content server 302.... A streaming mode of pre-encryption is also possible in which content is ‘streamed’ from a source of raw content such as a video (possibly analog tape) through an MPEG encoder, sent to the OLES to **perform encryption and finally stored on a VOD server.**”), FIG. 3 (annotated below).



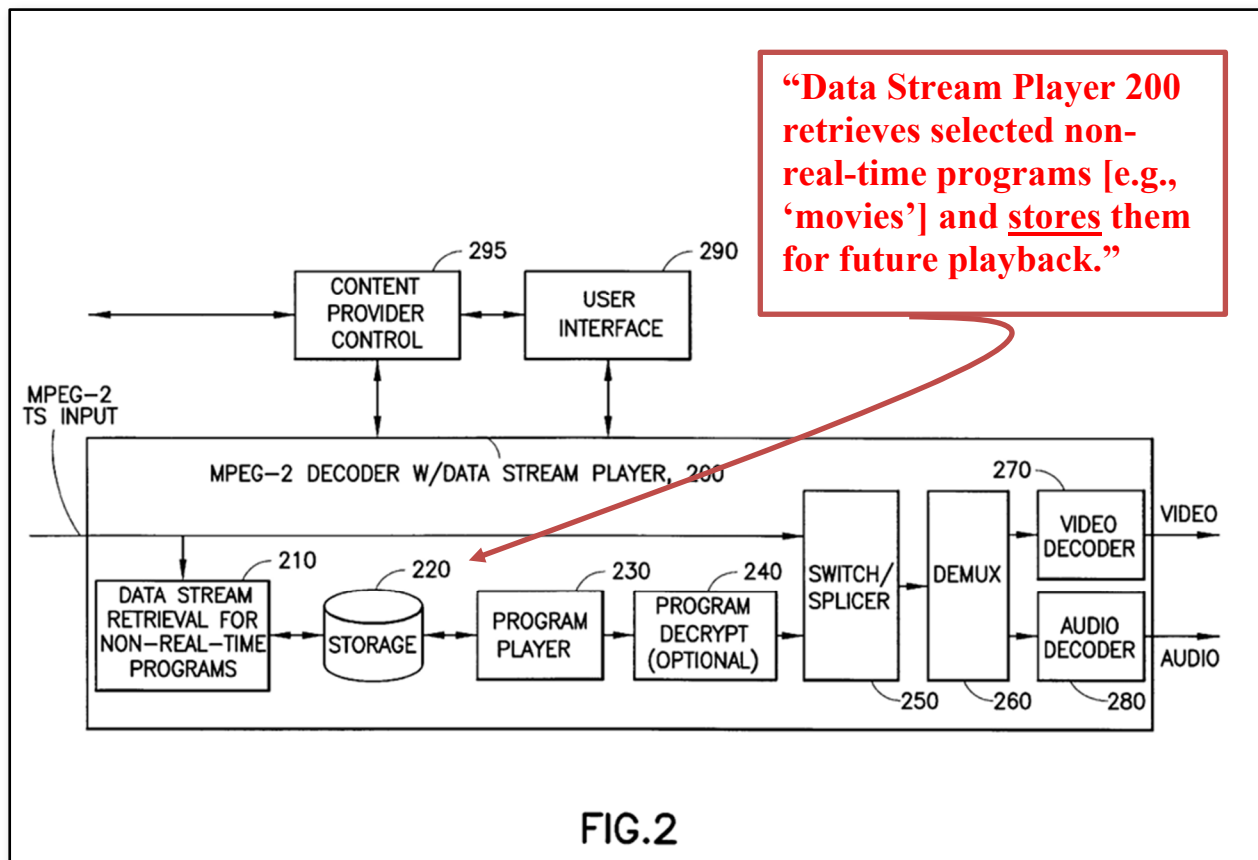
135. As another example I cite in response to SFI's arguments and Dr. Rubin's opinions, EX1097 at 2:31-53 describes as background knowledge that using "real-time encryption" in VOD system is problematic and "poses much greater cost and space issues." Thus, rather than use real-time encryption, it was known to implement content-delivery systems (like Foster and Hayhurst) that "that **encrypts content offline (typically before the content is requested by the user)** before it is distributed to point-to-point systems such as cable systems. The system allows content to be encrypted once, **at a centralized facility**, and to be useable at different point-to-point systems." EX1097, 2:31-53, FIG. 1

(annotated below). That is, the content is encrypted and **stored** at the “centralized facility” so it could be delivered (again and again if popular content). Thus, EX1097 is also consistent with my opinion in my First Declaration that POSAs would have wanted to pursue Foster-Hayhurst for the various benefits that POSAs would have recognized, such as improving the aggregate speed of content delivery and fault tolerance. First Declaration, ¶¶241-246.

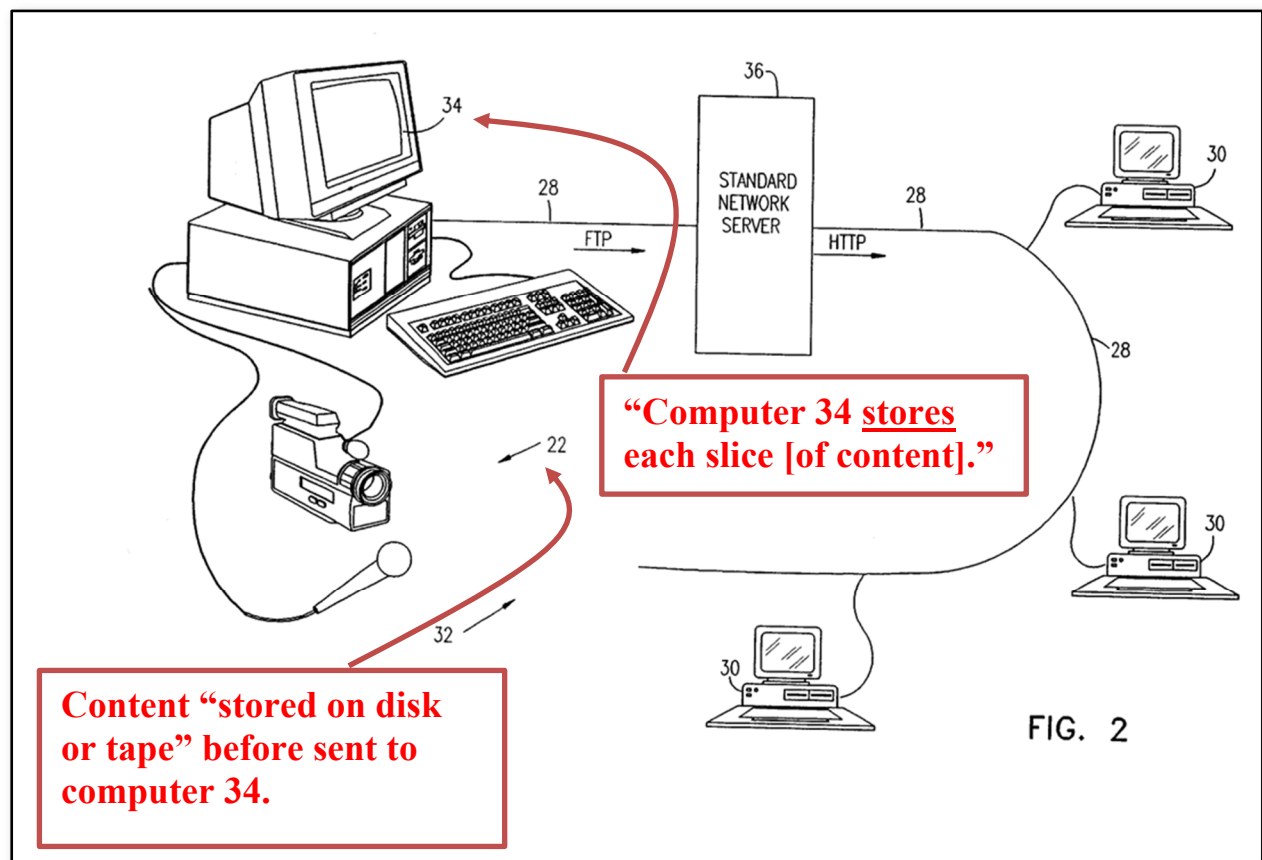


136. As another example I cite in response to SFI’s arguments and Dr. Rubin’s opinions, EX1075 at 1:51-67 teaches that it was known to implement the “MPEG-2 transport” stream in a “system [that] should allow the programs and other content to be provided *on a non-real-time basis for later use by a recipient*.”

Such content should be able to be easily received, *recorded on a digital VCR or otherwise stored*, and played back at the user's convenience." EX1075 at 1:51-67 further confirms that it was desirable to have "the delivery of *pre-encrypted content* ... in *non-real-time*." See also EX1075, FIG. 2 (annotated below). Thus, EX1075 further corroborates my opinion that POSAs not only understood Foster's program stream to be beneficially and primarily directed to storage, but that it was beneficial to *non-real-time* storage and delivery of encrypted content (i.e., *not* live streaming).

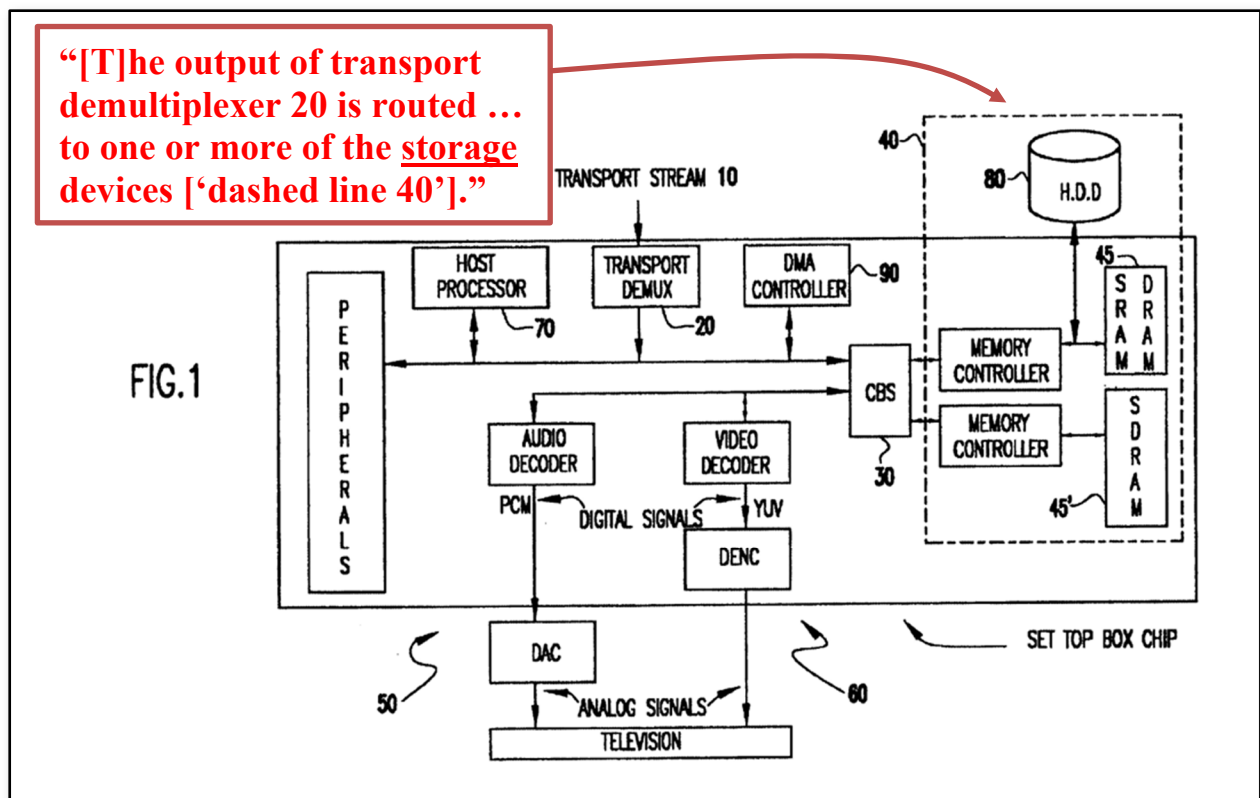


137. As another example I cite in response to SFI's arguments and Dr. Rubin's opinions, EX1076 at 9:53-10:5 discloses a prior art system where even "Broadcast data are [] input to the computer, for example, from input devices 22, or from a video, audio or animation sequence *stored on disk or tape*." That is, even "live" broadcasting of content is rarely real-time, but instead often retrieved from a *stored device* that stores the content for some later use.



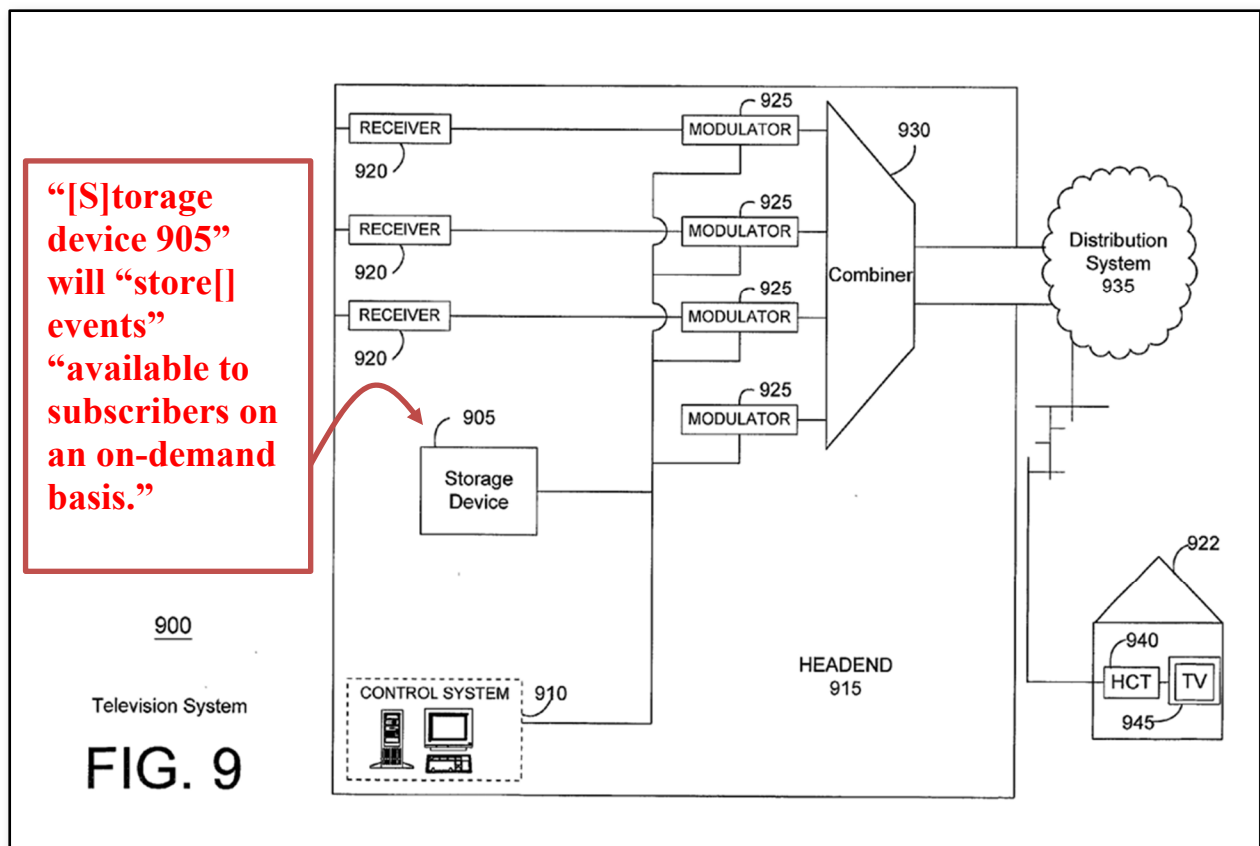
138. As another example I cite in response to SFI's arguments and Dr. Rubin's opinions, EX1077 (which is another patent by several of the Foster inventors), which discloses that it was known in the prior art to have "pay-per-view

movies” that were viewed “at a time convenient” to the user. EX1077, 1:12-21. In those system, users typically had a “set-top box (STB)” (as in Foster, [0040], [0052]), that was expected to have “substantial storage within the STB” and/or paired with “further storage on an onboard mass storage device such as a hard disk drive or compact disk recorder/playback device.” EX1077, 1:48-65. EX1077 at 3:56-5:8 goes on to describe processing of an MPEG “packetized elementary stream” (as in Foster) for “stor[ing] to mass storage.”



139. In response to SFI’s arguments and Dr. Rubin’s opinions that Foster’s CAS implementation would only be for use in streaming live television, I cite EX1092. EX1092’s Figure 9 (annotated below) illustrates “a storage device 905,

such as a video-on-demand server.” EX1092, [0079]. “A control system 910 within the headend 915 **is programmed to store all the streams** included within popular events that are received from one of the receivers 920” (where the receivers are receiving content from content providers). That is, even in cable / satellite company systems, a conventional video-on-demand server for CAS (like those discussed in Foster) will have a processor that instructs storage devices to store content for later downloading and storage to another storage device.



Thus, it is clear from Foster's disclosure (and reinforced POSAs' background knowledge) that Foster's technique was designed for use in CAS-based systems that included the storage of content both before and after delivery of the encrypted content to end-user recipients (both for streaming and for non-streaming applications).

140. I note that at his deposition, Dr. Rubin provided various testimony about the wide-spread use of CAS in the prior art, such as for use in the security of "[f]ile systems" and "video-on-demand" or "basically any place that you have permissions and you want to prevent people who aren't authorized from getting to something." Rubin-Depo., 85-86; *see also* Rubin-Depo., 87-89 (testifying that "content producers [] would license the content to the [VOD] hosting sites" and that it was known to have "service providers that would keep the content" that "would be stored," for example, "[o]n a server"), 90-94 (testifying that VOD system included "intermediaries that would house the actual content" for distribution to end users). I agree that POSAs would have known that CAS was widely used including VOD / VOD-like systems that are discussed in Foster and Hayhurst.

141. Foster's compatibility and suggestion of CAS-based and/or VOD-based system thus corroborates the opinions I expressed in my First Declaration that POSAs had motivation to combine Foster and Hayhurst because "[l]ike Foster,

Hayhurst concerns ‘electronic media delivery, such as video-on-demand.’ First Declaration, ¶228 (quoting Hayhurst, [0002]). Indeed, Hayhurst explicitly says its technique is beneficial for “cable television (CATV) system[s]” and “a content provider in **delivering video-on-demand.**” *E.g.*, Hayhurst, [0002] (“This application relates to the field of electronic media delivery, **such as video-on-demand, including the secure handling and storage of such media.**”), [0005]-[0006] (“The present invention addresses the problem of prohibitive bandwidth requirements for a content provider in delivering video-on-demand or other large media files....In response to such a request, the other subscriber units direct to the requesting subscriber all of the segments, which are then assembled into the complete media file (movie) for display, **or stored for later display**, on the requesting subscriber's unit.”).

142. Thus, as I explained in my First Declaration (¶¶240-253), POSAs had motivation to pursue the Foster-Hayhurst combination for purposes of storing encrypted content in Hayhurst distributed storage devices, including for streaming applications (e.g., streamed VOD) and non-streaming applications (e.g., VOD for downloading and storage).

2. The Foster-Hayhurst combination discussed in my First Declaration is consistent with Foster’s goals.

143. Based on SFI’s narrow interpretation of Foster’s teachings being limited to streaming of live broadcasts, SFI argues Hayhurst’s distributed-storage

technique would “thwart Foster’s goals, make Foster inoperable for its intended purpose, and compromise any security benefits otherwise provided by Foster alone.” POR, 57-68. In my opinion, none of these three critiques would have had merit to POSAs, let alone discouraged POSAs from pursuing the Foster-Hayhurst combination.

144. The first two critiques – based on Foster’s alleged “goals” and “intended purpose” – depend on narrowly interpreting Foster as being concerned **only** with transmission of synchronized data for real-time video streaming purposes, *e.g.*, live video of a sporting event. POR, 51-58 (“Foster seeks to securely stream content, *e.g.*, live television, without delay...Foster’s system is able to deliver live or real-time content...”). However, as I discussed above (§VI.B.1), Foster (like Hayhurst) is also concerned with **storage** of data for later/repeated playback. *See also, e.g.*, Hayhurst, [0002]-[0006].

145. SFI argues that POSAs would have been discouraged from the potential for excessive delays in delivering content to users in the Foster-Hayhurst system because a movie file would purportedly be distributed amongst “thousands” of recipients. POR, 53-54. In my opinion, POSAs would disagree with SFI’s argument for multiple reasons.

146. First, SFI ignores my opinion, supported by multiple corroborating references, that POSAs knew distributed distribution systems like Hayhurst’s

would have **reduced** delays in Foster’s system because it would have increased the “aggregate speed” and the “efficiency/intelligence” of content delivery. First Declaration, ¶¶242-244 (citing Hayhurst, EX1028, EX1029, EX1030, EX1031). For example, in Foster alone, content is delivered from one server, e.g., at the content service provider’s system. Foster, [0005] (In DRMS, “content is packaged and loaded onto a web server.”), [0049] (“Referring now to FIG. 1, a depiction of a Digital Rights Management System (DRMS) is shown. As depicted, content 10 is packaged with access rights/content usage conditions 12 and loaded onto media server 14. The keys used to access the content are downloaded to PC 18 as a separate component from the content.”), [0052] (“[A]s depicted in FIG. 2, content 22 is provided to a head-end server 24 and encrypted using a key.”).

147. Thus, as I explained in my First Declaration (with corroboration), that could sometimes create a potential bottleneck (e.g., if the server is overloaded or its network connection (or the server itself) fails). First Declaration, ¶¶242-243.

148. In contrast, in Foster-Hayhurst, based on Hayhurst’s teachings, the content service provider will no longer “require[] a[n] extensive amount of bandwidth” (Hayhurst, [0004]) because the overall data set is delivered from multiple storage devices (each effectively a server of the content that it stores). Thus, as I explained in my First Declaration (with corroboration), POSAs understood that a delivery network like Hayhurst would, on average, allow for

faster delivery of content to a user. First Declaration, ¶243 (citing, e.g., EX1028, EX1031). For example, EX1031 at [0019]-[0025] (cited First Declaration, ¶243), notes benefit of “transfer[ring] by downloading data from multiple sources or channels in parallel” (as in Hayhurst). One of those benefits is “**Speed**” (optimizing to “download the data as absolutely fast as possible”) and “**Intelligence**” (“download[ing] from the most desirable mirrors or sources”). EX1031, [0019]-[0020].

149. Again, I note that the ’116 patent itself recognizes (as POSAs already knew) that sending data via “multiple parallel streams” (as would have occurred in Hayhurst-Foster but not Foster alone) provides for “**increasing speed and efficiency.**” EX1001, 70:10-12.

150. Thus, Hayhurst’s technique would not only **not** add excessive delays, as SFI alleges, but POSAs would have reasonably expected Foster-Hayhurst to **reduce** delays over Foster alone. Similarly, by having redundant copies of encrypted segments as Hayhurst teaches, it would also be reasonably expected to **reduce** delays overall (and provided added protection against corrupted and/or lost data) because the system could intelligently determine which storage devices were better suited to deliver the content (e.g., selecting storage devices that were closest to the end-user and/or less busy).

151. In his declaration, Dr. Rubin opines that “there is no indication that Foster’s system suffers from the bandwidth constraints confronted by Hayhurst’s video-on-demand system.” Rubin-Decl., ¶148. Even if that were true, I disagree that that would have dissuaded POSAs from pursuing the Foster-Hayhurst combination. For one, Hayhurst teaches that bandwidth constraints were a concern for content service providers (Hayhurst, [0004]-[0005]), whether or not Foster itself recognized it. Limited bandwidth would have also been recognized as a universal consideration for any content storage and/or content delivery system. Indeed, as I discuss below, at deposition, Dr. Rubin recognized that the prior art comprised content delivery networks. The very purpose of which was to address, among other considerations, bandwidth limitations of a content service provider: as the number of users in the system increases, and/or the size of the data being stored and delivered increases, and/or the speed by which you want to stored and/or retrieve stored content, the available bandwidth can become a limiting factor that POSAs would have known was a consideration in the design of systems like Foster’s. For example, beyond Hayhurst, EX1031 similarly recognizes as background information that “for many applications and file transfers taking many hours [e.g., for video files], even on a broadband connection, there exists **a need to optimize the delivery of such files and make that delivery robust in the face of network changing** conditions.” EX1031, [0004]. EX1031’s solution to that

known need and problem is similar to Hayhurst's: "transferring data over multiple channels in parallel, possibly from multiple servers." EX1031, [0004]. Moreover, Foster does not discourage the use of techniques that reduce bandwidth considerations for content service providers, and SFI and Dr. Rubin did not identify such disclosure in Foster. Foster also does not expressly identify any concerns with the potential for power failures and the need for data backup, but Foster did not need to identify those concerns because those were considerations well-known to POSAs in designing systems like Foster's. Thus, even if it were true that Foster itself did not recognize a need to improve bandwidth constraints at a content service provider, POSAs would have recognized that that would have been a potential concern that would have been addressed by the Foster-Hayhurst combination.

152. Dr. Rubin offered testimony at his deposition that is consistent with my opinion that POSAs would have recognized that Hayhurst would have provided benefits to Foster (including in terms of speed and intelligence). For example, Dr. Rubin testified about prior art content delivery networks, where intermediaries would store content on servers that "would be closer to the consumer than the servers of either the service provider or the content owner" and would delivery content over networks using intelligence, such as by having "servers based on population density." Rubin-Depo., 90-93. That testimony is consistent with the

testimony I offered in my First Declaration (with corroborating evidence) that POSAs had motivation for Foster-Hayhurst because it would have beneficially allowed for “efficiency / ‘intelligence’” in the delivery of Foster’s encrypted content, such as “by being able to select storage device(s) that would provide the best delivery, based on availability, **location**, etc., [which] would have improved the reliability and speed of retrieval.” First Declaration, ¶243 (citing EX1031).

153. Second, in my opinion POSAs would disagree with the premise of SFI’s argument that Foster-Hayhurst necessarily has “thousands” of users delivering any single piece of content. POR, 46-47. Hayhurst says a cable system “**may**” have “thousands of subscribers” (Hayhurst, [0016]), but my opinions regarding the Foster-Hayhurst combination did not require thousands of storage devices because the challenged claims only require a plurality of storage devices, not thousands. Indeed, Hayhurst’s exemplary embodiment has only **four** storage nodes.

154. Moreover, Hayhurst also says “not all of the subscriber units” must participate “at all times” in storing/distributing data, which teaches that even when content is stored at end-users, fewer than all the users (i.e., fewer than “thousands”) can store chunks of any given dataset. Hayhurst, [0023]. The number of users that would store *data chunks* of any given *dataset* would be a matter of design choice, balancing the benefits of having more storage devices (e.g., additional security)

against considerations like the complexity of managing storage and delivery of each data set increases as the number of storage devices increases. For example, POSAs would have recognized that there would have not been a need to distribute a single data set to one million storage devices. In the same way, POSAs would have recognized that even in Foster-Hayhurst systems with “thousands” of storage devices, there would have been reasons to distribute each data set to far fewer than a thousand storage devices (under some design considerations, a single-digit number of storage devices could be sufficient and/or beneficial).

155. Regardless, even in implementations of Foster-Hayhurst that were using “thousands” of storage devices, POSAs would have still recognized the combination had the same benefits over Foster alone that I identified in my First Declaration (e.g., using smaller segments across more storage devices increased security and flexibility in the intelligent delivery of the content).

156. Furthermore, Hayhurst says rather than use a cable network or even a wide-area network for the storage and delivery of encrypted content, Hayhurst explicitly teaches that “any other communications network,” like “a local area network (LAN),” could be used. Hayhurst, [0016]. POSAs knew that LANs typically had far fewer than thousands of nodes. In other words, Hayhurst is teaching that its storage and delivery system is beneficial even in systems that

distribute the encrypted data segments to only a handful of storage devices (e.g., as a conventional tradeoff in speed and complexity).

157. In addition to the evidence noted above that I cited in my First Declaration (§243), in response to SFI's arguments and Dr. Rubin's opinions that POASs would not have pursued Foster-Hayhurst, my opinion about Foster-Hayhurst increasing the speed of delivery over Foster alone is further corroborated by EX1078. EX1078 discloses that peer-to-peer like delivery network will reduce the load on the server that originates the content. EX1078 says that its "present invention is

to improve an information delivery rate on a network such as Internet, reduce load occurring in a server and the network and shorten a data delivery path, by **enabling information to be delivered in parallel by means of use of a distributed computing technology** utilizing idle resources including CPUs and hard discs of general user computers (hereinafter, referred to as 'peers') such as PCs and a bandwidth of the network, and a peer-to-peer technology for enabling data to be delivered between general user computers without passage through the central server.

EX1078, 4:5-12.

158. EX1078 further teaches using a peer-to-peer delivery system:

to enable clients to make **use of rapid delivery, to reduce load** in a central server and a network and **to shorten a data delivery path** by means of delivery of data from adjacent peers rather than the central

server, **by dividing information to be serviced and redundantly storing the divided data in a plurality of peers** on behalf of the central server for entirely storing and delivering the information to be serviced, searching peers storing the divided data corresponding to information requested by the clients and **selecting peers capable of providing the most rapid delivery rate** when the clients request the information, and **causing the data to be delivered in parallel to the clients requesting the information.**”

EX1078, 4:13-28.

159. In my opinion, POSAs would have also disagreed with SFI’s argument that the Foster-Hayhurst combination is less secure than Foster’s unmodified system. POR, 64-66. That argument is premised on the idea that (i) changing or replacing Foster’s key-encryption-key (“KEK”) would have been more difficult in Foster-Hayhurst because the KEK would have (according to SFI) been located in more than one storage location, and (ii) Foster-Hayhurst would have stored the in more locations, which SFI alleges would have made it more vulnerable to attack. POR, 64-66.

160. As to SFI’s first allegation regarding the need to update or replace the KEK, I note that the claims do not have any limitations directed to updating or replacing any of the encryption keys, in particular the *external key* that is met by Foster’s KEK under SFI’s claim constructions. The Foster-Hayhurst combination that I described in my First Declaration did not require updating or replacing the

KEK as part of the combination, and thus the combination would not have had to update or replace the KEK. Thus, even if it were true that Foster by itself preferred to have the ability to update / change the KEK, that was not a feature that would have needed to have been brought over to Foster-Hayhurst for the combination's system to have worked properly and thus it would not have discouraged POSAs from pursuing the combination, which had other benefits over Foster alone that would have motivated the combination (e.g., efficiency, intelligent content delivery, enhanced security from distributed storage, etc.).

161. Moreover, Foster itself does not **require** updating or replacing the KEK. Instead, Foster explicitly says the KEK “**can** be” (i.e., does not **have** to be) changed in Foster's system. Foster, [0054]. Thus, to the extent SFI is arguing Foster (by itself) required the ability to change its KEK, POSAs would disagree as Foster says otherwise.

162. Foster also recognizes that KEKs “are inherently more secure than title keys, and **do not need to be changed as frequently.**” Foster, [0055]. That disclosure in Foster further reinforces my opinion that the ability to change / update the KEK would not have dissuaded POSAs from pursuing Foster-Hayhurst because the fact that the KEK did not have to be changed frequently meant it would have been an easier problem to solve compared to an encryption key that had to be changed more frequently.

163. But even though changing and updating the KEK is not required by Foster, Foster-Hayhurst, **or** the challenged claims, SFI's argument still fails because POSAs would have known how to update and/or replace KEKs that have been distributed across many devices/users. In addition, POSAs would have recognized that whether the encrypted content was stored in one location (as in Foster alone) or across distributed storage devices (as in Foster-Hayhurst), changing the KEK would have to be propagated across the same number of segments. Accordingly, there would be no meaningful difference (let alone insurmountable different requiring more than ordinary skill) to POSAs in complexity between Foster's updating of the KEK and Foster-Hayhurst's updating the KEK. (I note, however, that as I discuss below, Foster-Hayhurst would not have stored the KEK on every storage device in the system anyway.)

164. In response to SFI's POR and Dr. Rubin's opinions regarding updating KEKs in distributed networks, my opinion regarding POSAs' background knowledge that KEKs could be changed and/or updated even when distributed to multiple storage device is corroborated by the following references.

165. First, I note that I am co-author on several prior art, peer-reviewed journal articles that were intended for POSAs in the field of securing data that addressed techniques for updating encryption keys in large and/or distributed systems. Below is a list of several, peer-reviewed articles and/or papers relating to

presentations given that relate to key management with large userbases and the benefits thereof. Each of these papers were published and publicly available in the prior art that I was a co-author on that are in the field of securing data and which POSAs would have been familiar with:

- Suman Banerjee and Bobby Bhattacharjee. “Scalable Group Communication over IP Multicast”, In Proceedings of the International Conference on Network Protocols, pages 261–271, 2001.
- Suman Banerjee and Samrat Bhattacharjee. “Scalable Secure Group Communications over IP-multicast”, IEEE Journal of Selected Areas in Communications, JSAC, Vol. 20, No. 8, pages 1511 - 1527, October 2002.

166. In response to SFI’s POR and Dr. Rubin’s opinions about the alleged complexity in changing a KEK across multiple devices, I further cite, EX1079, which is a 2003 IEEE article entitled “Key Management and Distribution for Secure Multimedia Multicast.”³ EX1079 describes using “[m]ulticast

³ IEEE Xplore’s website lists EX1079 as having been published on “31 December 2003” in “IEEE Transactions on Multimedia (Volume: 5, Issue: 4, December 2003).” *See* <https://ieeexplore.ieee.org/document/1244812>. Based on the above evidence and my review of the indicia appearing on EX1079 and my

communications” because it “is efficient since it reduces the demands on the network and bandwidth resources.” EX1079, 544 (identifying potential applications in “pay-per-view” and “interactive television”). EX1079 (at 545) noted that “Due to the dynamic nature of the group, and the possible expiration of keying material, **it is necessary to update both the SK and KEKs using rekeying messages.**” EX1079 (at 544-546) then goes on to provide a prior art procedure for updating / changing KEKs that have been distributed to multiple users, which corroborates my opinion that POSAs were aware of techniques for updating KEKs across many devices.

167. In response to SFI’s POR and Dr. Rubin’s opinions about the alleged complexity in changing a KEK across multiple devices, I cite EX10180. EX1080 is another journal article from 2003 obtainable from IEEE Xplore, entitled “Pushing the internet direct to the user - security issues.”⁴ It describes a system where

understanding of the IEEE and IEEE Xplore, it is my opinion that EX1079 was publicly accessible in the prior art by being disseminated or otherwise made available to POSAs exercising reasonable diligence by 2004.

⁴ IEEE Xplore’s website lists EX1080 as having been added to IEEE Xplore on “24 November 2003” and published in “IEE Seminar on IP Over Satellite - The next Generation: MPLS, VPN and DRM Delivered Services, 2003.” *See*

“[d]ata is encrypted at the source by the TEK” and the system “distribute[s] the key to each receiver,” and notes that the TEK “must itself be encrypted” using a KEK. EX1080, 6. EX1080 further notes that “[i]n multicast sessions users who leave a session and are no longer entitled to take part force the changing of the TEK, and a multicast with a flat Key Management system and a large number of subscribers” there can be “a huge **rekey problem**,” but techniques for addressing that problem already had solutions that were known to POSAs, such as “described in a number of IETF MSEC Internet Drafts.” EX1080, 6-7. Thus, EX1080 further corroborates my opinion that POASs would not have been dissuaded from Foster-Hayhurst because of a purported challenge in changing the KEK(s).

168. As to SFI’s argument that Foster-Hayhurst would make the KEK less secure (POR, 64-66), it is my opinion that POSAs would disagree. SFI’s argument fails (at least) because Foster-Hayhurst does not have to give every device a copy of a KEK and I did not opine otherwise in my First Declaration. Rather, as in

<https://ieeexplore.ieee.org/abstract/document/1246598>. Based on the above evidence and my review of the indicia appearing on EX1080 and my understanding of the IEEE and IEEE Xplore, it is my opinion that EX1080 was publicly accessible in the prior art by being disseminated or otherwise made available to POSAs exercising reasonable diligence by 2004.

Foster alone, Foster-Hayhurst would only give users access to a KEK to a user after that user has been authorized to receive and decrypt the encrypted content. Foster, [0052] (“Upon receiving encrypted content, **if the entitlement management message indicates that a subscriber is authorized** to access a certain channel, [then] the control words in the entitlement control message **will be converted into a key** that is passed to descrambler 36.”), [0055] (“If, however, the DVD player was a known non-compliant or circumvention device, its internal device keys **would be unable to recover the correct key encrypting key 28**, because the KMB would have been modified to exclude the device”).

169. Similarly, in Foster-Hayhurst, the plurality of storage devices would not each store a KEK; instead, the KEK would only be available to the end-user(s) that have been authorized to receive and decrypt an entire data set. For example, based on Hayhurst’s teachings, the “encryption key to unlock the movie” (the KEK in Foster-Hayhurst) would have been stored on the “main server,” not at each subscriber. Hayhurst, [0007], [0021] (cited First Declaration, ¶¶32, 251). As I explicitly opined in my First Declaration (¶251), POSAs would have found it obvious “[i]n the Foster-Hayhurst combination, [that] the KEK **would be maintained in a storage device** (e.g., Foster’s database 426) until required by a subscriber....consistent with Foster’s providing the KEK to authorized recipients.”

In Foster-Hayhurst, Foster’s database is part of the main server coordinating the storage and delivery of content by the storage devices.

170. That obvious implementation of Foster-Hayhurst is also consistent with Foster’s teachings to decouple the transferring of information needed to obtain the KEK from the transferring of the encrypted data and encryption keys. *See, e.g.*, Foster, [0055] (“It should be understood 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....”), [0056] (“the KMB should be delivered synchronously with, or prior to, the encrypted content”).

171. Finally, I also disagree with SFI that POSAs would have found Foster-Hayhurst to be less secure than Foster alone. POR, 58. Rather, as I explained in my First Declaration (with corroborating evidence, including from Hayhurst’s own disclosure) that Foster-Hayhurst would **increase** the overall security of Foster’s system. First Declaration, ¶¶242-243. Again, Hayhurst explicitly says it “provides a **high level of security**” via its distributed storage technique. Hayhurst, [0008]; *See also supra* §VI.A.

172. In my opinion, POSAs would have agreed that Foster-Hayhurst was more secure than Foster alone in several ways. As I discussed in my First Declaration (¶¶242-243) and above (§VI.A.1, §VI.A.2.c), Hayhurst would have

reduced the risk of unauthorized access to content in Foster because an attacker of Foster-Hayhurst would have to collect multiple encrypted *data chunks* from multiple storage devices, rather than from one storage device, as in Foster alone. Hayhurst further teaches that security is increased by its system because an attacked would have to know how to reassemble the Foster-Hayhurst *data chunks* because “access to the correct method for recombining the segments to recreate the original media file could be stored in a password-protected file, linked to an encryption key, or locked to all devices except the specific hardware device that created the bundled segments.” Hayhurst, [0021].

173. Moreover, even if SFI were arguing (or at suggesting) that Foster taught distributing its KEK to end users even before users were authorized to access the encrypted content, Foster-Hayhurst would not have done that because (as I discuss above) Hayhurst teaches to keep the KEK at the main server that coordinates the storage devices, rather than at one of the distributed storage devices.

174. But even if Foster-Hayhurst were less secure than Foster in some of the ways SFI asserts, in my opinion POSAs would have been motivated to pursue Foster-Hayhurst for the reasons I identified in my First Declaration because POSAs would have recognized that the combination would have added various

security techniques that Foster lacks, as my First Declaration discussed (and I reiterated above and further below).

175. Separate from improvements in security that I reiterated above, Foster-Hayhurst would have **also** provided other (undisputed) benefits in terms of increased speed, intelligence, efficiency, bandwidth requirements, as I discussed in my First Declaration (¶¶242-246) and above (§VI.B.1). Those benefits would have also motivated POSAs to pursue Foster-Hayhurst, with a reasonable expectation of success. Indeed, POSAs knew that designing an encryption and storage system for encrypted content was a matter of tradeoffs:

Data protection in storage management systems is a **tradeoff** between user's convenience and security, speed of operation and capabilities of the encryption algorithm, length of the encryption keys, government restrictions, and other elements known in the art. There are many encryption algorithms available that vary by strength, speed and other parameters.

EX1081, 15:6-25.

176. As I discussed above, I note that at deposition, Dr. Rubin testified that it was known in the prior art that content owners would use intermediaries like content providers and content delivery networks (working with content providers) to improve the delivery of content “because they could optimize for sharing content in many different locations” such as having more servers in higher “population density” locations. Rubin-Depo., 91-93. Dr. Rubin further testified that by having storage devices that are “remote from each other” and storing different

portions of data “a physical compromise of the system would only yield the portion that’s there [i.e., at the storage device that was compromised].” Rubin-Depo., 134-137 (further testifying that “redundancy” could be helpful “[i]n the event that there would be problems at some of the sites”). All of that testimony is consistent with my opinions in my First Declaration identifying several benefits POSAs would have recognized from Foster-Hayhurst, compared to Foster alone, which I corroborated by several prior art references.

177. That deposition testimony also directly contradicts Dr. Rubin’s declaration testimony that “Foster’s system has no need for redundancy as Foster’s system can simply resend dropped packets” (EX2033, ¶148): POSAs understood that the one of the reasons why redundancy would have been beneficial to Foster is that it would have provided a way to store and/or deliver content in more robust ways than was been possible when the single source of content in Foster’s “one-on-one transmission” (EX2033, ¶148) failed. POSAs would have recognized that designs with a single point for storing and/or delivering data could be fragile, *e.g.*, due to component and/or network disruptions. Unlike Foster alone, Foster-Hayhurst would have been obvious to POSAs as a more robust solution that was less susceptible to those problems because it provided redundant sources for storing and delivering data.

178. As with Dr. Rubin’s testimony alleging that Foster itself did not express a need for the benefits that Hayhurst’s technique would have provided, POSAs knew that systems like Foster’s were not designed for operation under the assumption that optimal conditions would always persist. Rather, POSAs designed those system to account for the real-world where problems could (and often did) arise, such as when bandwidth becomes limited (e.g., at times of high demand, like holiday periods), where data storage and/or delivery systems failed, where data could be lost or subject to attack if maintained in a single location and without backup, etc. Indeed, at deposition, Dr. Rubin admitted that POSAs knew that redundancy was helpful when “one of the sites went down, like a power failure.” Rubin-Depo., 134-137.

179. Dr. Rubin opines in his declaration that “Foster’s system has no ... need for efficiency/intelligence when sending on a one-to-one basis.” EX2033, ¶148. As an initial matter, I am unsure exactly what Dr. Rubin meant by “when sending on a one-to-one basis.” By prefacing it with “when,” Dr. Rubin appears to admit that there may be cases **when** a one-to-one basis is not used (or at least it was known to have systems that were not on a one-to-one basis). In any event, Foster only discusses an example system where content is delivered from one server – e.g., Foster’s media server – to recipients. *See, e.g.*, Foster, FIG. 1. That appears to be what Dr. Rubin means by a “one-to-one basis.” As I demonstrated

with corroboration in my First Declaration, POSAs would have recognized that having **more than one** source to deliver content (i.e., what Hayhurst teaches), allows for several benefits in terms of speed, intelligence, and efficiency (in addition to security and fault tolerance benefits) that are not possible with a single source for the delivery of content. First Declaration, ¶243 (citing Hayhurst and additional corroboration).

180. As just one of the examples I identified in detail in my First Declaration, by having multiple sources of storage devices storing and delivering the encrypted content, Foster-Hayhurst's system could select the storage devices that were best equipped to delivery content to a particular user compared to other storage devices in the network **and** that, in many cases, would have been better than Foster's system having only one server of the content. For example, consider a system where one recipient was in New York and another recipient for the same content was in California. In Foster alone, both recipients would be served by the same media server that might be equidistant from both recipients or (more likely) much closer to one recipient over the other. In Foster-Hayhurst, storage devices in (or closer to) New York could deliver the content to the New York recipient, whereas storage device in (or closer to) California could deliver the content to the California recipient. That is a form of intelligence and efficiency that I identified in my First Declaration, with support from the prior art.

181. I note too that at deposition Dr. Rubin recognized that POSAs were aware of situations similar to the one I just described. Dr. Rubin testified that so-called content delivery networks were known and were used by content owners and/or content service providers to deliver content “because they could optimize for sharing content with customers in many different locations.” Rubin-Depo., 91. For example, he testified that “instead of AT&T having to build warehouses all over the place where their customers lives, they could rely on the CDN,” who would “have servers based on population density” so “they could service [a] city **optimally**.” Rubin-Depo., 89-92. That is precisely an example of efficient / intelligent storage and delivery of content that POSAs would have recognized that Foster-Hayhurst provided that Foster’s one-to-one delivery could not.

182. In summary, it is my opinion that POSAs would have disagreed with SFI’s argument that POSAs would not have had reason to pursue Foster-Hayhurst with a reasonable expectation of success. That is because SFI’s arguments are largely (if not exclusively) premised on incorrectly limiting Foster to streaming of live broadcasts. SFI’s argument fails because POSAs would have recognized that Hayhurst would have **improved** Foster in several ways (including in terms of security but also intelligence, speed, redundancy, etc.). SFI’s argument also fails because even if Foster-Hayhurst would have led to any decrease in security over Foster alone (which POSAs would have disagreed with), POSAs would have

recognized that Hayhurst would have provided other security and non-security benefits that Foster by itself did not achieve.

C. Response to Argument that Foster’s Encrypted Payloads Are Not “*Data Chunks*”

183. Claims 1 and 9 of the ’116 patent require “*distributing the data set into a plurality of data chunks*” and then encrypting those *data chunks*. As I explained in my First Declaration (¶¶254-257), Foster’s payloads meet the claimed *data chunks* and Foster’s content meets the claimed *data set*. Foster’s payloads are each only a portion of Foster’s content and thus “*none of* [Foster’s payloads] *are, by themselves, sufficient to reconstruct* [Foster’s content].” That is all claims 1 and 9 require of the claimed *data chunks*. See First Declaration, ¶¶254-257.

184. SFI does not dispute that each of Foster’s payloads are insufficient by themselves to reconstruct Foster’s content. Rather, in its POR, SFI argues “Foster does not disclose generated the claimed ‘data chunks’” chunks for other reasons that SFI says were explained earlier in the POR’s Ground 2 discussion of Foster’s payloads. POR, 68.⁵ In that discussion, SFI alleges only that “[t]here is no reason

⁵ SFI primarily discusses its challenge to whether Foster discloses *data chunks* in its response to Ground 2 (Tajima-Foster). See POR, 19-23. However, the *data chunks* in Tajima-Foster are **Tajima’s** data fragments, not Foster’s packets / content units.

to assume that a POSITA would understand that Foster’s packet payload corresponds to the data chunk of the ’116 patent.” POR, 20. As support, SFI cites Dr. Rubin’s testimony that the ’116 patent explains that “partitioning the data set into chunks increases security because when ‘data...remains whole and intact,’ it is ‘subject to attack,’” and that the ’116 address this concern “in part, by ‘performing a cryptographic parsing and splitting of the encrypted file.’” EX2033, ¶69 (citing EX1001 57:10-24). But the ’116 specification makes clear that “the invention is not intended to be limited” to any particular data splitting process, but “[r]ather, a skilled artisan will recognize from the disclosure herein, a wide number of alternatives for the data splitting process.” EX1001, 21:44-48.

185. Indeed, the specification has multiple disclosures that inform POSAs that the pieces of data that are split from the data set (i.e., *data chunks* in the ’116 patent’s claims) can be of any size of data from the data set or can each be the same size or different sizes of data from the data set. For example, the ’116 patent says that “data may be split **into any desired, selected, predetermined, or**

SFI’s POR attempts to rely on those arguments regarding the Tajima-Foster ground for its arguments for the Foster-Hayhurst ground. POR, 68 (“As explained earlier, Foster does not disclose generating the claimed ‘data chunks.’ *See supra* Section III.A.2.a.i.”).

randomly assigned size unit.” EX1001, 21:55-59. The ’116 patent similarly says that a data set may be split into data units that are “**all of the same size**, a fixed set of different sizes, a combination of sizes, or randomly generates [sic] sizes.” EX1001, 21:65-22:1.

186. Elsewhere, the patent says “[i]t is readily apparent to those of ordinary skill in the art that **the split data unit sizes may be a wide variety of data unit sizes** or patterns of sizes or combinations of sizes.” EX1001, 21:62-65. Similarly, the patent says “[i]t will be understood that any other suitable parsing and splitting approach may be used to generate portions of data from an original data set in accordance with the present invention.” EX1001, 75:15-19.

187. Thus, the ’116 patent explicitly (and repeatedly) allows for a data set to be split in any way desired using contiguous or non-contiguous data units of any size. EX1001, 21:44-22:4, 58:58-62, 75:15-19, 77:51-82:2; First Declaration, ¶¶87-88, 254-257. In my opinion, a POSA understood from the ’116 specification that a *data chunk* is simply a portion (or part) of the *data set*—i.e., a subset of the data set—and that nothing in the ’116 patent supports a narrow interpretation of *data chunk* that excludes Foster’s payloads.

188. At deposition, Dr. Rubin testified that in his opinion “the share is the thing that something gets split into” (Rubin-Depo., 17) and that the terms share and chunk “refer[] to the pieces that have been split from the original data set” (Rubin-

Depo., 157). POSAs would have recognized that Foster's payloads (content units) meet Dr. Rubin's view of what the *shares / chunks* are in SFI's patent because each of Foster's payload's data is a split portion of the complete *data set* being secured by Foster.

189. To the extent Dr. Rubin's declaration testimony (EX2033, ¶69) cited by SFI (POR, 20-22) is meant to suggest that a *data chunk* must be cryptographically generated (*e.g.*, randomly split), the '116 patent simply provides no support for narrowing how data chunks are generated. The '116 patent is clear that a data set can be divided in any way, including simply using “**fixed** or predetermined data unit size...**or** a randomly generated data unit size,” as an **alternative**. EX1001, 22:1-4. The '116 patent places no limitation on how a data set is divided.

190. The dependent claims of the '116 patent evidence that the independent claims are likewise not limited to a particular way of distributing the data set into data chunks. Dependent claims 7/9 limit claim 1's and claim 9's *data chunks*, respectively, to ones that are generated using cryptographic techniques like a “random technique.” I understand that a dependent claim is narrower than the independent claim from which it depends. Thus, the fact that a dependent claim from claim 1 and a dependent claim from claim 9 require use a random technique, claims 1 and 9 do not have any such limitation and cover data chunks resulting

from non-random techniques. SFI's suggestion that claim 1's or claim 9's *data chunks* must result from "cryptographic parsing and splitting" is unsupported by the '116 patent and contradicted by the claims.

191. I note that Dr. Rubin nowhere provides his opinion that Foster's payloads do not meet the claimed *data chunks*. Dr. Rubins testimony regarding the Foster-Hayhurst combination are limited to motivation-to-combine arguments. See EX2033, ¶¶104-150. As discussed above, the section of the POR that SFI alleges explained that "Foster does not disclose generating the claimed 'data chunks'" (POR, 68)—which it does not—presents arguments against the combination of Tajima-Foster. In that section, SFI cites Dr. Rubin's testimony that (1) "packet payload (as in Foster), however, is not created to protect data." POR, 22 (citing EX2033, ¶70).

192. But POSA understand that generating Foster's packet payloads from Foster's content is done specifically to protect data (and not merely as a function of the MPEG standard). Foster splits the data set into *data chunks* for the purpose of protecting the individual pieces of data and overall *data set*. Foster individually encrypts each *data chunk* (packet payload) with an encryption key (potentially a different encryption key). First Declaration, ¶¶213-224, 258-259 (citing Foster, [0004], [0064]-[0068], [0071]). Thus, Foster's disclosure that dividing the content into content units so that each payload can be encrypted and that using a different

encryption key “provide[s] enhanced security by preventing an entire content stream from being accessed with a single key” (Foster, [0004], [0071]) demonstrates that Foster’s splitting was performed to protect the data.

193. Contrary to SFI, Foster’s splitting and encryption is **not** “just a by-product of the MPEG streaming standard.” POR, 21. The MPEG standard, for example, does not require encrypting each of the packet payloads (let alone encrypting them with respective, different encryption keys as taught for one embodiment of Foster). It is Foster’s encryption technique, not a requirement of the MPEG standard, that encrypts each of the individual *data chunks* for the purpose of protecting the *data set*.

194. Furthermore, based on Hayhurst’s teachings, in the Foster-Hayhurst **combination**, the data set is split into smaller portions (*data chunks*) for purposes of storing the encrypted segments on distributed storage devices. As I discussed in my First Declaration (¶¶228-232, ¶¶241-243) and again above (§VI.B), splitting Foster-Hayhurst’s *data set* was done for several benefits, including to address the problem when “data...remains whole and intact’ [and] it is ‘subject to attack’” (POR, 32, quoting EX1001, 57:10-24). *See supra* §V.A.2.c (discussing how Hayhurst is reasonably pertinent to this problem). That is, in Foster-Hayhurst, the splitting of the *data set* into *chunks* (packet payloads) is done for the purposes of

adding security by, among other things, encrypting the payloads and separately storing the content units on geographically dispersed storage devices.

195. SFI's POR identified several exhibits, several of which do not appear to be prior art, that allegedly teach that a "payload" or "packet" can be "divided into 'chunks' or 'fragments.'" POR, 20-21 (citing EX2012-EX2013, EX2041-EX2043, EX2036-EX2038). I have reviewed these exhibits. They generally concern generic "packets" (e.g., an IP packet), and are not about packets produced according to the MPEG standard, let alone according to Foster's or Hayhurst's disclosures (or the combination thereof).⁶

⁶ I note too that Foster itself is not limited to using MPEG data, as "Content" is defined as "**any** data such as digital image, sound or binary data deliverable from a source to a recipient." Foster, [0037]. Thus, Foster (let alone the Foster-Hayhurst combination) would not be limited to dividing its *data set* into MPEG "packets" (or any other packet). Regardless, as I discuss in this section of my reply declaration, I do not agree and a POSA would not agree that a packet of MPEG-processed data cannot be a *data chunk*; and a POSA would find that Foster's processing in particular generates *data chunks* for the reasons I discussed in my First Declaration (¶¶254-257) and here in response to SFI's and Dr. Rubin's arguments.

196. In my opinion, the exhibits SFI cites are not inconsistent with my opinion that POSAs' would have understood that Foster-Hayhurst meets the claimed *data chunks*. In fact, SFI's cited Exhibits merely show that a "chunk" is a data unit that is smaller than the data on which "chunking" is performed. POR, 20-21; EX2033, ¶71. The fact that a payload can also be divided into chunks does not refute, and in fact supports, that when Foster's content is divided into smaller payloads, those payloads are *data chunks*. This is reinforced by other literature that demonstrates that POSAs knew that something referred to as a "packet" could also be called a "chunk" / "data chunk" of a data set if (as in Foster, Hayhurst, and Foster-Hayhurst) the packet comprises a less-than-entire portion of the data set.

197. In response to SFI's and Dr. Rubin's argument regarding POSAs' understanding of Foster's content units, I cite EX1082, which unlike SFI's exhibits **is** in the context MPEG-encoded data.⁷ EX1082 says that with MPEG, each

⁷ I note that a document sharing the same author, title, and publisher as EX1082 was cited on page 2 of U.S. Patent 7,898,951 (filed Aug. 6, 2004; issued Mar. 1, 2011). See <http://bit.ly/49OIc3o> (PDF of US7,898,951 on USPTO's website). The '951 patent lists as a reference titled "Mike Knee, 'MPEG Video'" and indicates it was published "Mar. 2002" by "Snell & Wilcox," and was available from <http://www.bretl.com/mpeghtml/codecdial.HTM>. I have also reviewed the Internet

packet's "payload [] is a **chunk** of a particular elementary stream." EX1082, 8. The "elementary [content] stream" is Foster's *data set*. Foster, [0064]-[0068], [0071], FIG. 8; First Declaration, ¶¶216-217, ¶254.

198. And even as to generic computer "packets," prior art recognizes a "packet" can be a chunk (or a "share") of a dataset. For example, EX1083 at 7:33-40 gives an example of "[a] packet [that] is a chunk of data provided by the application program." Thus, EX1084 contradicts SFI's argument that something called a "packet" cannot also be "chunk of data."

Archive's Wayback Machine and identified archived copies of Snell & Wilcox's website, including an archive from May 15, 2006 of a "White Papers" web page. <https://bit.ly/4fuXZWb>. That archived web page provides a link to download "MPEG Video: Mike Knee," which links to a November 9, 2006 archived PDF of EX1082. <https://bit.ly/3ZRVAiu>. See also the following link to a 2006 archived copy of Snell & Wilcox's home page, which provides a link to a "Knowledge Center" web page that links to the "White Papers" web page I identified above. <https://bit.ly/4iPP58B>. See also <https://bit.ly/49P0xxe> (link to IEEE Xplore's page for an article added to IEEE Xplore on January 3, 2006, which lists "Mike Knee, 'MPEG Video,' Snell and Wilcox" as a reference cited in that article, which I have confirmed by reviewing a copy of that January 2006 article).

199. Similarly, EX1084 at [0026] teaches that an “RTP **packet**” can be “A **chunk or portion of multimedia data** as sent over an IP network....” Thus, EX1084 contradicts SFI’s argument that something called a “packet” cannot also be “chunk or portion of” a data set.

200. As another example, EX1085 (at 4:7-25) describes a system that performs “disassembling a data set into a plurality of data blocks...transforming each of the data blocks into L *data chunks*,” where “[a] size of *each data chunk can correspond to packet size*” (i.e., there is only one chunk-per-packet). Thus, EX1085 contradicts SFI’s argument that something called a “packet” cannot also be “data chunk” of a data set or that a packet is only divisible into data chunks rather than itself being the data chunk.

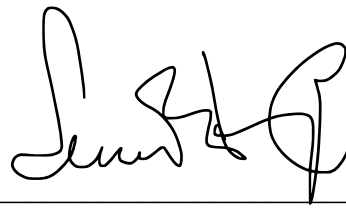
201. Thus, the above references corroborate my opinion that SFI is incorrect in its argument that POSAs would understand that packets cannot be *chunks* merely because packets could, in some systems, **also** be “divided into ‘chunks’” (POR, 20)). The above references demonstrate that POSAs would disagree, as do I.

* * *

I declare that all statements made herein of my knowledge are true, that all statements made on information and belief are believed to be true, and that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code.

I declare under penalty of perjury that the foregoing is true and correct.

Dated: December 23, 2024

A handwritten signature in black ink, appearing to read 'Samrat Bhattacharjee', written over a horizontal line.

Samrat Bhattacharjee, Ph.D.