

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

---

**BEFORE THE PATENT TRIAL AND APPEAL BOARD**

---

AMAZON.COM, INC., and  
AMAZON WEB SERVICES, INC.,

Petitioners,

v.

DIVX, LLC,

Patent Owner.

---

Case: IPR2025 01222

U.S. Patent No. 10,412,141

---

**DECLARATION OF HARI KALVA**

## TABLE OF CONTENTS

|                                                       | Page |
|-------------------------------------------------------|------|
| I. INTRODUCTION .....                                 | 14   |
| II. BACKGROUND AND QUALIFICATIONS .....               | 14   |
| III. COMPENSATION AND RELATIONSHIP WITH PARTIES.....  | 19   |
| IV. Materials Considered .....                        | 19   |
| V. Basis of Opinion .....                             | 27   |
| A. Legal Standard for Anticipation.....               | 27   |
| B. Legal Standard for Obviousness .....               | 27   |
| C. Claim Construction.....                            | 29   |
| VI. RELATED LITIGATION AND RELATED PATENTS .....      | 30   |
| VII. Overview of the 141 Patent .....                 | 30   |
| A. Technical Background.....                          | 30   |
| 1. Progressive Playback Systems.....                  | 30   |
| a. Architecture of Progressive Playback Systems ..... | 30   |
| b. Media Files.....                                   | 32   |
| c. Requests .....                                     | 34   |
| d. Trick Play and Indexing.....                       | 35   |
| e. Encryption and Digital Rights Management .....     | 37   |
| B. Patent Description .....                           | 39   |
| C. Priority Date .....                                | 40   |
| D. Level of Ordinary Skill in the Art.....            | 40   |
| VIII. Prior Art Overview .....                        | 41   |

# TABLE OF CONTENTS (Continued)

|                                                                                                                                                                  | Page |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|
| A. Hagai.....                                                                                                                                                    | 41   |
| B. Li.....                                                                                                                                                       | 48   |
| C. Park.....                                                                                                                                                     | 50   |
| D. Schmitz.....                                                                                                                                                  | 51   |
| IX. GROUND 1: Claims 1, 3-8, 10, 20, 22-27, and 29 are obvious over Hagai, Li, and General Knowledge of a POSITA.....                                            | 53   |
| A. A POSITA would have combined Hagai and Li.....                                                                                                                | 53   |
| B. Claim 1 .....                                                                                                                                                 | 55   |
| 1. [1.pre] “A playback device, comprising:” .....                                                                                                                | 55   |
| 2. [1.a] “a processor; and”.....                                                                                                                                 | 57   |
| 3. [1.b] “a non-volatile storage containing an application for causing the processor to perform the steps of:” .....                                             | 58   |
| 4. [1.b(i)] “establishing at least one connection for communicating with a remote server system;” .....                                                          | 59   |
| 5. [1.b(ii)] “obtaining information from a remote server system describing at least one video track, multiple audio tracks, and multiple subtitle tracks;” ..... | 61   |
| 6. [1.b(iii)] “selecting a video track from the at least one video track;” .....                                                                                 | 69   |
| 7. [1.b(iv)] “requesting a header describing the selected video track;” .....                                                                                    | 70   |
| 8. [1.b(v)] “selecting an audio track from the multiple audio tracks;” .....                                                                                     | 72   |
| 9. [1.b(vi)] “obtaining index information indicating the locations of audio and video data within the selected audio and video tracks;” .....                    | 74   |

**TABLE OF CONTENTS**  
**(Continued)**

|                                                                                                                                                                                | <b>Page</b> |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| 10. [1.b(vii)] “determining byte ranges to request from the selected audio and video tracks using the index information;” .....                                                | 78          |
| 11. [1.b(viii)] “requesting byte ranges from the selected video track and the selected audio track from the remote server system;” .....                                       | 79          |
| 12. [1.b(ix)] “buffering received bytes of information comprising audio and video data;” .....                                                                                 | 81          |
| 13. [1.b(x)] “checking that sufficient data is buffered to commence playback and playing back the buffered audio and video data;” .....                                        | 82          |
| 14. [1.b(xi)] “responding to a received seek instruction by;” .....                                                                                                            | 83          |
| 15. [1.b(xi)(1)] “pausing playback;” .....                                                                                                                                     | 84          |
| 16. [1.b(xi)(2)] “determining byte ranges to request from the selected audio and video tracks based upon a new playback location using the index information;” .....           | 85          |
| 17. [1.b(xi)(3)] “requesting byte ranges required to play the selected audio and video tracks from the new playback location from the remote server;” .....                    | 85          |
| 18. [1.b(xi)(4)] “buffering received bytes of information comprising audio and video data pending resumption of playback; and” .....                                           | 85          |
| 19. [1. b(xi)(5)] “checking that sufficient data is buffered to commence playback and playing back the buffered audio and video data.” .....                                   | 85          |
| C. Claim 3: “The playback device of claim 1, wherein the step of requesting a header describing the selected video track comprises the step of requesting a DRM header.” ..... | 85          |

**TABLE OF CONTENTS**  
**(Continued)**

|                                                                                                                                                                                                                                                               | <b>Page</b> |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| D. Claim 4: “The playback device of claim 3, wherein the application is further capable of causing the processor to perform the step of decrypting the DRM header.” .....                                                                                     | 91          |
| E. Claim 5 .....                                                                                                                                                                                                                                              | 93          |
| 1. [5.pre], [5.a] “The playback device of claim 3, wherein: the DRM header identifies at least one key; and” .....                                                                                                                                            | 93          |
| 2. [5.b] “the application is further capable of causing the processor to perform the steps of decrypting encrypted frames of video using the at least one key prior to decoding.” .....                                                                       | 93          |
| F. Claim 6 .....                                                                                                                                                                                                                                              | 96          |
| 1. [6.pre], [6.a] “The playback device of claim 5, wherein: encrypted frames of video are only partially encrypted;” .....                                                                                                                                    | 96          |
| 2. [6.b] “the received bytes of information contain references to portions of video frames that are encrypted; and” .....                                                                                                                                     | 98          |
| 3. [6.c] the step of decrypting encrypted frames of video using the at least one key prior to decoding further comprises decrypting encrypted portions of frames of video using the references to the portions of the video frames that are encrypted.” ..... | 99          |
| G. Claim 7: “The playback device of claim 1, wherein playing back the buffered video data from the requested byte ranges provides fast forward trick play.” .....                                                                                             | 101         |
| H. Claim 8: “The playback device of claim 1, wherein playing back the buffered video data from the requested byte ranges provides fast rewind trick play.” .....                                                                                              | 102         |
| I. Claim 10 .....                                                                                                                                                                                                                                             | 103         |

**TABLE OF CONTENTS**  
**(Continued)**

|                                                                                                                                                                                                                         | <b>Page</b> |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| 1. [10.pre], [10.a]: “The playback device of claim 1, wherein the application is further capable of causing the processor to perform the steps of: selecting a subtitle track from the multiple subtitle tracks;” ..... | 103         |
| 2. [10.b]: “requesting data from the selected subtitle track from the remote server system, where the data contains subtitle text; and” .....                                                                           | 106         |
| 3. [10.c]: “superimposing the subtitle text over video during playback.” .....                                                                                                                                          | 107         |
| J. Claim 20: .....                                                                                                                                                                                                      | 108         |
| 1. [20.pre] A method of playing back content on a playback device, comprising:.....                                                                                                                                     | 108         |
| 2. [20.a] establishing at least one connection for communicating with a remote server system using a playback device;.....                                                                                              | 108         |
| 3. [20.b] obtaining information from a remote server system using the playback device, where the obtained information describes at least one video track, multiple audio tracks, and multiple subtitle tracks; .....    | 108         |
| 4. [20.c] selecting a video track from the at least one video track;.....                                                                                                                                               | 108         |
| 5. [20.d] requesting a header describing the at least one video track using the playback device; .....                                                                                                                  | 108         |
| 6. [20.e] selecting an audio track from the multiple audio tracks using the playback device; .....                                                                                                                      | 108         |
| 7. [20.f] obtaining index information indicating the locations of audio and video data within the selected audio and video tracks; .....                                                                                | 108         |

# TABLE OF CONTENTS (Continued)

|                                                                                                                                                                                    | Page |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|
| 8. [20.g] determining byte ranges to request from the selected audio and video tracks using the index information;.....                                                            | 108  |
| 9. [20.h] requesting byte ranges from the selected video track and the selected audio track from the remote server system using the playback device; .....                         | 109  |
| 10. [20.i] buffering received bytes of information comprising audio and video data on the playback device; .....                                                                   | 109  |
| 11. [20.j] checking that sufficient data is buffered to commence playback and playing back the buffered audio and video data using the playback device;.....                       | 109  |
| 12. [20.k] responding to receipt of a seek instruction at the playback device by:.....                                                                                             | 109  |
| 13. [20.k(i)] pausing playback on the playback device; .....                                                                                                                       | 109  |
| 14. [20.k(ii)] determining byte ranges to request from the selected audio and video tracks based upon a new playback location using the index information;.....                    | 109  |
| 15. [20.k(iii)] requesting byte ranges required to play the selected audio and video tracks from the new playback location from the remote server using the playback device; ..... | 109  |
| 16. [20.k(iv)] buffering received bytes of information comprising audio and video data pending resumption of playback using the playback device; and .....                         | 110  |
| 17. [20.k(v)] checking that sufficient data is buffered to commence playback and playing back the buffered audio and video data using the playback device.....                     | 110  |
| K. Claim 22: “The method of claim 20, wherein requesting a header describing the selected video track further comprises requesting a DRM header using the playback device.” .....  | 110  |

# TABLE OF CONTENTS

## (Continued)

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | Page |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|
| L. Claim 23: “The method of claim 22, further comprising decrypting the DRM header.” .....                                                                                                                                                                                                                                                                                                                                                                                      | 111  |
| M. Claim 24: “The method of claim 23, wherein: the DRM header identifies at least one key; and the method further comprises decrypting encrypted frames of video using the at least one key prior to decoding using the playback device.” .....                                                                                                                                                                                                                                 | 111  |
| N. Claim 25: “The method of claim 24, wherein: encrypted frames of video are only partially encrypted; the received bytes of information contain references to portions of video frames that are encrypted; and decrypting encrypted frames of video using the at least one key prior to decoding further comprises decrypting encrypted portions of frames of video using the references to the portions of the video frames that are encrypted using the playback device..... | 112  |
| O. Claim 26: “The method of claim 20, wherein playing back the buffered video data from the requested byte ranges provides fast forward trick play.”.....                                                                                                                                                                                                                                                                                                                       | 112  |
| P. Claim 27: “The method of claim 20, wherein playing back the buffered video data from the requested byte ranges provides fast rewind trick play.” .....                                                                                                                                                                                                                                                                                                                       | 112  |
| Q. Claim 29: “The method of claim 20, further comprising: selecting a subtitle track from the multiple subtitle tracks using the playback device; requesting data from the selected subtitle track from the remote server system using the playback device, where the data contains subtitle text; and superimposing the subtitle text over video during playback using the playback device.”.....                                                                              | 113  |
| X. GROUND 2: Claims 9, 12-14, 16-18 and 28 are obvious over Hagai, Li Park, and General Knowledge of a POSITA.....                                                                                                                                                                                                                                                                                                                                                              | 113  |
| A. Motivation to Combine .....                                                                                                                                                                                                                                                                                                                                                                                                                                                  | 113  |
| B. Claims 9 and 28.....                                                                                                                                                                                                                                                                                                                                                                                                                                                         | 115  |

# TABLE OF CONTENTS

## (Continued)

|                                                                                                                                                                                                                                                        | Page |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|
| Claim 9: “The playback device of claim 1, wherein the application is further capable of causing the processor to perform the step of discarding audio and video data contained within the buffer in response to a received seek instruction”; and..... | 115  |
| Claim 28: “The method of claim 20, further comprising discarding audio and video data contained within the buffer in response to a received seek instruction.” .....                                                                                   | 115  |
| C. Claim 12 .....                                                                                                                                                                                                                                      | 117  |
| 1. [12.pre] “A playback device, comprising:” .....                                                                                                                                                                                                     | 117  |
| 2. [12.a] “a processor; and” .....                                                                                                                                                                                                                     | 117  |
| 3. [12.b] “a non-volatile storage containing an application for causing the processor to perform the steps of:” .....                                                                                                                                  | 118  |
| 4. [12.b(i)] “establishing at least one connection for communicating with a remote server system;” .....                                                                                                                                               | 118  |
| 5. [12.b(ii)] “obtaining information from a remote server system describing at least one video track, and at least one audio track;” .....                                                                                                             | 118  |
| 6. [12.b(iii)] “selecting a video track from the at least one video track;” .....                                                                                                                                                                      | 118  |
| 7. [12.b(iv)] “requesting a header describing the selected video track, where the requested header comprises a DRM header;” .....                                                                                                                      | 118  |
| 8. [12.b(v)] “decrypting the DRM header;” .....                                                                                                                                                                                                        | 119  |
| 9. [12.b(vi)] “selecting an audio track from the at least one audio track;” .....                                                                                                                                                                      | 119  |

**TABLE OF CONTENTS**  
**(Continued)**

|                                                                                                                                                                       | <b>Page</b> |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| 10. [12.b(vii)] “obtaining index information indicating the locations of audio and video data within the selected audio and video tracks;” .....                      | 119         |
| 11. [12.b(viii)] “determining byte ranges to request from the selected audio and video tracks using the index information;” .....                                     | 119         |
| 12. [12.b(ix)] “creating a buffer;” .....                                                                                                                             | 120         |
| 13. [12.b(x)] “requesting byte ranges from the selected video track and the selected audio track from the remote server system;” .....                                | 120         |
| 14. [12.b(xi)] “buffering received bytes of information comprising audio and video data;” .....                                                                       | 120         |
| 15. [12.b(xii)] “checking that sufficient data is buffered to commence playback;” .....                                                                               | 121         |
| 16. [12.b(xiii)] “decrypting encrypted frames of video using information from the decrypted DRM header;” .....                                                        | 121         |
| 17. [12.b(xiv)] “playing back the buffered audio and the decrypted video data;” .....                                                                                 | 121         |
| 18. [12.b(xv)] “responding to a received seek instruction by;” .....                                                                                                  | 121         |
| 19. [12.b(xv)(1)] “pausing playback;” .....                                                                                                                           | 121         |
| 20. [12.b(xv)(2)] “discarding buffered audio and video data;” .....                                                                                                   | 121         |
| 21. [12.b(xv)(3)] “determining byte ranges to request from the selected audio and video tracks based upon a new playback location using the index information;” ..... | 122         |
| 22. [12.b(xv)(4)] “requesting byte ranges required to play the selected audio and video tracks from the new playback location from the remote server;” .....          | 122         |

**TABLE OF CONTENTS**  
**(Continued)**

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | <b>Page</b> |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| 23. [12.b(xv)(5)] “buffering received bytes of information comprising audio and video data pending resumption of playback; and” .....                                                                                                                                                                                                                                                                                                                                         | 122         |
| 24. [12.b(xv)(6)] “checking that sufficient data is buffered to commence playback;” .....                                                                                                                                                                                                                                                                                                                                                                                     | 122         |
| 25. [12.b(xv)(7)] “decrypting encrypted frames of video using information from the decrypted DRM header; and” .....                                                                                                                                                                                                                                                                                                                                                           | 122         |
| 26. [12.b(xv)(8)] “playing back the buffered audio and decrypted video data.” .....                                                                                                                                                                                                                                                                                                                                                                                           | 122         |
| D. Claim 13: “The playback device of claim 12, wherein: the DRM header identifies at least one key; and the step of decrypting encrypted frames of video using information from the decrypted DRM header comprises decrypting encrypted frames of video using the at least one key prior to decoding.” .....                                                                                                                                                                  | 123         |
| E. Claim 14: “The playback device of claim 13, wherein: encrypted frames of video are only partially encrypted; the received bytes of information contain references to portions of video frames that are encrypted; and the step of decrypting encrypted frames of video using the at least one key prior to decoding further comprises decrypting encrypted portions of frames of video using the references to the portions of the video frames that are encrypted.” ..... | 123         |
| F. Claim 16: “The playback device of claim 12, wherein playing back the buffered video data from the requested byte ranges provides fast forward trick play.” .....                                                                                                                                                                                                                                                                                                           | 123         |
| G. Claim 17: “The playback device of claim 12, wherein playing back the buffered video data from the requested byte ranges provides fast rewind trick play.” .....                                                                                                                                                                                                                                                                                                            | 123         |

**TABLE OF CONTENTS**  
**(Continued)**

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | <b>Page</b> |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| H. Claim 18: “The playback device of claim 12, wherein the application is further capable of causing the processor to perform the steps of: obtaining information from a remote server system describing at least one subtitle track; selecting a subtitle track from the at least one subtitle track; requesting data from the selected subtitle track from the remote server system, where the data contains subtitle text; and superimposing the subtitle text over video during playback.” ..... | 124         |
| XI. GROUND 3: Claims 2, 11, 21, and 30 are obvious over Hagai, Li, Schmitz, and General Knowledge of a POSITA .....                                                                                                                                                                                                                                                                                                                                                                                  | 124         |
| A. Motivation to Combine .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | 124         |
| B. Claims 11 and 30 .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                            | 128         |
| Claim 11: “The playback device of claim [1], wherein the application is further capable of causing the processor to perform the step of flushing previous byte range requests in response to the received seek instruction.” .....                                                                                                                                                                                                                                                                   | 128         |
| Claim 30: “The method of claim 20, further comprising flushing previous byte range requests in response to the received seek instruction.” .....                                                                                                                                                                                                                                                                                                                                                     | 128         |
| C. Claims 2 and 21 .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | 130         |
| Claim 2: “The playback device of claim [1], wherein the application is further capable of causing the processor to asynchronously request byte ranges from the selected video track and the selected audio track from the remote server.” .....                                                                                                                                                                                                                                                      | 130         |
| Claim 21: “The method of claim 20, further comprising asynchronously requesting byte ranges from the selected video track and the selected audio track from the remote server using the playback device.” .....                                                                                                                                                                                                                                                                                      | 130         |
| XII. GROUND 4: Claims 15 and 19 are obvious over Hagai, Li, Park, Schmitz, and General Knowledge of a POSITA .....                                                                                                                                                                                                                                                                                                                                                                                   | 133         |

**TABLE OF CONTENTS**  
**(Continued)**

|                                                                                                                                                                                                                                                 | <b>Page</b> |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| A. Motivation to Combine .....                                                                                                                                                                                                                  | 133         |
| B. Claim 15 .....                                                                                                                                                                                                                               | 134         |
| Claim 15: “The playback device of claim 12, wherein the application is further capable of causing the processor to asynchronously request byte ranges from the selected video track and the selected audio track from the remote server.” ..... | 134         |
| C. Claim 19 .....                                                                                                                                                                                                                               | 134         |
| Claim 19: “The playback device of claim 12, wherein the application is further capable of causing the processor to perform the step of flushing previous byte range requests in response to the received seek instruction.” .....               | 134         |
| XIII. CONCLUSION.....                                                                                                                                                                                                                           | 135         |

I, Hari Kalva, Ph.D., declare as follows:

## **I. INTRODUCTION**

1. I have been asked by the parties requesting this review, Amazon.com, Inc. and Amazon Web Services, Inc. (“Petitioner”), to provide my expert opinions in support of the petition for Inter Partes review of U.S. Patent No. 10,412,141 (the “141 patent” or “challenged patent”) challenging the patentability of claims 1-30 (the “challenged claims”).

2. In summary, it is my opinion that the challenged claims are disclosed and rendered obvious by the prior art. I set forth below my detailed opinions.

## **II. BACKGROUND AND QUALIFICATIONS**

3. I am Professor and Department Chair in the Department of Electrical Engineering and Computer Science at Florida Atlantic University. I regularly teach courses on Engineering Design, Video Communication, and Cloud Computing. I am also Director of Florida Atlantic University’s Multimedia Lab, in which capacity I conduct research in the broad area of visual computing that include video compression, communication, and analysis. Part of the research relates to development and optimization of video compression standards.

4. I received my Doctorate in Electrical Engineering from Columbia University in 2000. Prior to that, I received a Master of Philosophy in Electrical Engineering from Columbia University in 1999, a Master of Science in Computer

Engineering from Florida Atlantic University in 1994, and a Bachelor of Technology in Electronics and Communications Engineering from the N.B.K.R. Institute of Science and Technology at Sri Venkateswara University in Tirupati, Andhra Pradesh, India.

5. I have taught in the Department of Electrical Engineering and Computer Science at Florida Atlantic University since 2003. Before that, I was a Consultant in the Video Processing Group for Mitsubishi Electric Research Labs from 2001 to 2003; Co-Founder and Vice President of Engineering at Flavor Software, a New York based startup developing MPEG-4 standard based multimedia solutions, from 2000 to 2001; a Research Staff Associate in the Image and Advanced Television Laboratory at Columbia University from 1995 to 1996; and a Teaching Assistant in the Department of Electrical Engineering and Computer Science at Florida Atlantic University from 1993 to 1994.

6. I have extensive experience in multimedia data compression technologies. In addition to my education and nearly three decades of professional experience in data compression technologies, at both academic and commercial levels, I have been granted numerous patents for inventions relating to such technologies; I have authored over 100 peer refereed publications regarding such technologies; and I have contributed to several international video compression standards, including contributions of standard essential patents.

7. I am a 2025 Inductee to the Florida Inventors Hall of Fame and recognized for “Significant contributions to international standards for video compression, communications, and analytics that have enabled video streaming technologies, fundamentally transforming education, entertainment, and the workplace.” I am a Fellow of the National Academy of Inventors (NAI), an honor bestowed upon individuals whom the NAI’s Selection Committee deems to “have demonstrated a highly prolific spirit of innovation in creating or facilitating outstanding inventions that have made a tangible impact on the quality of life, economic development and the welfare of society.” I am also a senior member of the Institute of Electrical and Electronics Engineers (IEEE); a member of the IEEE’s Consumer Electronics, Circuits and Systems, Communications, and Signal Processing Societies, including the Multimedia Technical Committee of the IEEE’s Communications Society; Chair of the IEEE’s Palm Beach Section; and a member of the Association for Computer Machinery.

8. I have been granted 70 U.S. patents, covering inventions including systems and methods for encoding multimedia content; systems and methods for processing object-based audiovisual architecture; systems and methods for video and multimedia rendering, composition, and user-interactivity; methods for encrypting and compressing video, object-based audiovisual terminal and bitstream structures. I have also been granted over a dozen foreign patents covering similar

types of inventions. In addition, I have sold patent applications to Mitsubishi Electric Company covering multi-level bi-directional prediction and methods for encoding and decoding video.

9. In particular, I am an inventor of patented technology essential to the implementation of three video compression standards: H.264, also known as MPEG-AVC (for “Advanced Video Coding”) or MPEG-4 Part 10; H.265, also known as HEVC (for “High Efficiency Video Coding”); and H.266, also known as VVC (for “Versatile Video Coding”) or MPEG-I Part 3.

10. I was a key member of the team that initiated the MPEG-4 file format standards activity and wrote the software for the proposed file format. Based on the ideas and experiences with implementing the initial MPEG-4 file format, MPEG-4 video, and MPEG-4 Systems technologies, I developed requirements for the MPEG-4 file format that is used today. I have experience writing software for video compression and streaming systems including MPEG-1, MPEG-2, H.263, MPEG-4 part 2, and MPEG-4 part 10.

11. I have made over three dozen contributions to the development of international video compression standards from the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC). Among other video compression standards, I made a variety of contributions to the MPEG-4 group of international standards for compressing digital audio and visual

data, multimedia systems, and file storage formats. A list of my contributions to international standards is provided in my curriculum vitae.

12. I served as U.S. delegate to the ISO standards committees during the development of the MPEG-4, MPEG-7, VVC, and VCM international video compression standards. Over the past three decades, I have also served on the technical program committees of a variety of conferences of the International Conference on Consumer Electronics, the International Conference on Multimedia and Expo, the International Workshop on Multimedia Signal Processing, the IEEE's International Symposium on Multimedia, and other organizations.

13. I have published numerous peer refereed articles, chapters, and books pertaining to video compression technology over the past three decades. Many of these publications have focused on various international video compression standards, including MPEG-2, H.264, and H.265, also known as HEVC or MPEG-H Part 2. I have also authored many publications on video compression technologies such as intra-frame coding, a technique that enables smaller files sizes and lower bit rates with little to no loss in quality. A list of my peer reviewed publications, including all publications authored in the previous ten years, is provided in my curriculum vitae (EX1035).

14. I have previously been retained as an expert witness in cases involving patents before the Patent Trial and Appeal Board of the U.S. Patent and Trademark

Office to assess the relevance of prior art to patents related to HEVC video decoding. A list of cases in which I have been retained as an expert witness, including all cases at which I testified as an expert at trial or by deposition during the previous four years, is provided in my curriculum vitae (EX1035).

### **III. COMPENSATION AND RELATIONSHIP WITH PARTIES**

15. I am being compensated for my time at a rate of \$650 per hour. This compensation is not contingent upon my performance, the conclusions I reach in my analysis, the outcome of this matter, or any issues involved in or related to this matter.

16. I have no financial interest in Petitioner or any related parties.

### **IV. MATERIALS CONSIDERED**

17. I have reviewed and considered, in preparing this declaration, the 141 patent (EX1001), excerpts of its prosecution history (EX1002), and the following prior art references:

- U.S. Patent No. 7,051,110 to Makoto Hagai et al., titled “Data reception/playback method and apparatus and data transmission method and apparatus for providing playback control functions,” was filed December 20, 2000 and issued on May 23, 2006 (“Hagai”) (EX1004);

- U.S. Patent No. 8,731,369 to Adam H. Li et al., titled “Multimedia distribution system for multimedia files having subtitle information,” was filed on December 17, 2004 and issued on May 20, 2014 (“Li”) (EX1005);
- U.S. Patent Application Publication No. 2005/0198681 to Daniel John Park, titled “Playout buffer management to minimize startup delay,” was filed on March 8, 2004 and published on September 8, 2005 (“Park”) (EX1006).
- U.S. Patent No. 5,544,318 to Patrick L Schmitz et al., titled “Asynchronous media server request processing system for servicing reprioritizing request from a client determines whether or not to delay executing said reprioritizing request,” was filed on April 16, 1993, and issued on August 6, 1996 (“Schmitz”) (EX1007).

18. I am also aware of, and considered in forming my opinions presented in this declaration, other information available to a person of ordinary skill in the art at the time of the effective filing date of the challenged patent (“POSITA”), such as

other patents, reference materials, articles, and standards, and knowledge of a POSITA.<sup>1</sup> These include:

- Poole, C. & Bradley, J., *Developer's Digital Media Reference*, Focal Press (2003) ("Poole") (EX1008);
- U.S. Patent Application Publication No. 2005/0102371 to Emre Aksu, titled "Streaming from a server to a client," was filed on November 7, 2003 and published on May 12, 2005 ("Aksu") (EX1009);
- U.S. Patent No. 7,962,942 to Michael L. Craner, titled "Systems and methods for enhanced trick-play functions," was filed on February 28, 2006 and granted on June 14, 2011 ("Craner") (EX1010);
- U.S. Patent No. 5,874,986 to David Gibbon et al., titled "Method for communicating audiovisual programs over a communications network," was filed on June 26, 1996 and granted on February 23, 1999 ("Gibbon") (EX1011);

---

<sup>1</sup> I understand that the challenged patent has an effective filing date of January 5, 2007 for the purposes of this IPR, as I discuss in more detail below.

- Apple Computer, Inc., *QuickTime 7 User's Guide* (2005) ("QuickTime 7 User's Guide") (EX1012);
- Farber, N. et al., *Adaptive progressive download based on the MPEG-4 file format*, J. Zhejiang Univ. 2006 7 (Suppl. I) 106-111 ("Farber") (EX1013);
- U.S. Patent Application Publication No. 2006/0037057 to Shugong Xu, titled "Method and system of enabling trick play modes using HTTP GET," was filed on May 24, 2004 and published on February 16, 2006 ("Xu") (EX1014);
- U.S. Patent Application Publication No. 2005/0108320 to Robert D. Lord et al., titled "Method and apparatus for assisting with playback of remotely stored media files," was filed on November 18, 2003 and published on May 19, 2005 ("Lord") (EX1015);
- Sun, H., Chen, X., & Chiang, T., *Digital Video Transcoding for Transmission and Storage*, CRC Press, 2005 ("Sun") (EX1016);
- U.S. Patent No. 6,754,715 to Anthony Cannon et al., titled "Methods and apparatus for implementing control functions in a streamed video display system," was filed on November 9, 1999 and granted on June 22, 2004 ("Cannon") (EX1017);

- U.S. Patent No. 6,965,646 to Scott S. Firestone, titled “MPEG file format optimization for streaming,” was filed on June 28, 2000 and granted on November 15, 2005 (“Firestone”) (EX1018);
- U.S. Patent No. 7,826,712 to Dan Minnick et al., titled “Method and apparatus for receiving, storing, and presenting multimedia programming without indexing prior to storage,” was filed on August 29, 2006 and granted on November 2, 2010 (“Minnick”) (EX1019);
- U.S. Patent No. 5,566,208 to Mahesh Balakrishnan, titled “Encoder buffer having an effective size which varies automatically with the channel bit-rate,” was filed on March 17, 1994 and granted on October 15, 1996 (“Balakrishnan”) (EX1020);
- U.S. Patent No. 7,073,063 to Marcus Peinado, titled “Binding a digital license to a portable device or the like in a digital rights management (DRM) system and checking out/checking in the digital license to/from the portable device or the like,” was filed on June 27, 2001 and granted on July 4, 2006 (“Peinado”) (EX1021);

- U.S. Patent No. 7,376,233 to Brant L. Candelore et al., titled “Video slice and active region based multiple partial encryption,” was filed on October 18, 2002 and granted on May 20, 2008 (“Candelore”) (EX1022);
- U.S. Patent No. 5,477,263 to Daniel O’Callaghan et al., titled “Method and apparatus for video on demand with fast forward, reverse and channel pause,” was filed on May 26, 1994 and granted on December 19, 1995 (“O’Callaghan”) (EX1023);
- U.S. Patent No. 5,878,135 to Harold Blatter et al., titled “Decoding system for processing encrypted broadcast, cable or satellite video data,” was filed on November 27, 1996 and granted on March 2, 1999 (“Blatter”) (EX1024);
- U.S. Patent No. 7,647,614 to Jason Krikorian et al., titled “Fast-start streaming and buffering of streaming content for personal media player,” was filed on June 7, 2005 and granted on January 12, 2010 (“Krikorian”) (EX1025);
- U.S. Patent Application Publication No. 2001/0030710 to William B. Werner, titled “System and method for associating subtitle data with cinematic material,” was filed on December 1, 2000 and published on October 18, 2001 (“Werner”) (EX1026);

- U.S. Patent Application Publication No. 2005/0078947 to Hyun-Kwon Chung et al., titled “Information storage medium for storing subtitle and video mapping information, and method and apparatus for reproducing thereof,” was filed on July 29, 2004 and published on April 14, 2005 (“Chung”) (EX1027);
- Mahemoff, M., *Ajax Design Patterns: Creating Web 2.0 Sites with Programming and Usability Patterns*, O’Reilly (2006) (“Mahemoff”) (EX1028);
- International Telecommunication Union, *ITU-T Rec. H.262 – Information technology – Generic coding of moving pictures and associated audio information: Video*, (Feb. 2000) (EX1029).
- Fielding, R. et al., *RFC 2616 – Hypertext Transfer Protocol – HTTP/1.1*, Network Working Group (June, 1999) (“RFC2616”) (EX1030);
- U.S. Patent No. 7,739,351 to Taras Shkvarchuk et al., titled “Synchronous interface to asynchronous processes,” was filed on March 23, 2004 and granted on June 15, 2010 (“Shkvarchuk”) (EX1031);
- U.S. Patent No. 8,238,725 to Jason Demas et al., titled “System and method for providing personal video recording trick modes,”

was filed on June 18, 2002 and granted on August 7, 2012 (“Demas”) (EX1032);

- European Patent Application No. 1 158 799 to Karsten Rittner et al., titled “Method and receiver for providing subtitle data in several languages on demand,” was filed on May 18, 2000 and published on November 28, 2001 (“Rittner”) (EX1033); and
- European Patent Application No. 1 343 321 to Paul England et al., titled “Methods and systems for cryptographically protecting secure content in video memory,” was filed on November 26, 2002 and published on September 10, 2003 (“England”) (EX1034).

19. I understand that Petitioner and Patent Owner are involved in a parallel litigation involving seven patents, including the challenged patent.

20. Should new information be disclosed or discovered during this proceeding, I may supplement and further elaborate on the opinions I express in this declaration as necessary. Nothing in this declaration should be interpreted as an opinion that no grounds of invalidity exist other than the ones in the petition. I do not express such an opinion and have not reached such an opinion.

## **V. BASIS OF OPINION**

### **A. Legal Standard for Anticipation**

21. I understand that a claimed invention is unpatentable as “anticipated” by a single prior art reference if it discloses each and every element of the claim. I also understand that, even if a claim is not explicitly disclosed in the prior art reference, the prior art reference may nevertheless anticipate the claim if the subject matter of the claim is inherently present from the perspective of a POSITA. Moreover, I understand that even if the prior art reference does not disclose the limitations of a claim in the same arrangement or combination of the limitations of the claim, it may nevertheless anticipate if a POSITA would envisage the claimed arrangement or combination.

### **B. Legal Standard for Obviousness**

22. I understand that a claimed invention is unpatentable as “obvious” if the differences between the invention and the prior art are such that the subject matter of the claim as a whole would have been obvious to a POSITA.

23. It is my understanding that “obviousness” is a question of law based on underlying factual issues including (1) the scope and content of the prior art, (2) the differences between the prior art and the asserted claims, (3) the level of ordinary skill in the pertinent art, and (4) the existence of secondary considerations such as commercial success, long-felt but unresolved needs, and failure of others.

24. I understand that a single reference can render a patent claim obvious if any differences between that reference and the claims would have been obvious to a POSITA. Alternatively, the teachings of two or more references may be combined in the same way as disclosed in the claims, if such a combination would have been obvious to a POSITA. In determining whether a combination based on either a single reference or multiple references would have been obvious, I understand that it is appropriate to consider the following factors:

- Whether the teachings of the prior art references disclose known concepts combined in familiar ways, and when combined, would yield predictable results;
- Whether a POSITA could implement a predictable variation, and would see the benefit of doing so;
- Whether a POSITA would have recognized a reason to combine known elements in the manner described in the claim;
- Whether there is some teaching or suggestion in the prior art to make the modification or combination of elements claimed in the patent; and
- Whether the innovation applies a known technique that had been used to improve a similar device or method in a similar way.

25. In an obviousness evaluation based on a combination of multiple prior art references, I understand that the prior art references themselves may provide a suggestion, motivation, or reason to combine, but other times the nexus linking two or more prior art references is simple common sense. I further understand that obviousness analysis recognizes that market demand, rather than scientific literature, often drives innovation, and that a motivation to combine references may be supplied by the direction of the marketplace.

**C. Claim Construction**

26. I understand that, in *Inter Partes* Review (“IPR”) proceedings, a claim of a patent shall be construed using the same claim construction standard that would be used to construe the claim in a civil action filed in a U.S. District Court (which I understand is called the “*Phillips*” claim construction standard), including construing the claim in accordance with the plain-and-ordinary meaning of such claim as understood by one of ordinary skill in the art and the prosecution history pertaining to the patent.

27. I applied plain-and-ordinary meaning to all terms of the challenged claims for my analysis in this proceeding.

28. I am not an attorney, and my opinions are limited to what I believe a POSITA would have understood the meaning of certain claim terms to be, based on the patent specification and prosecution history in light of the POSITA’s knowledge.

## **VI. RELATED LITIGATION AND RELATED PATENTS**

29. I understand that the 141 patent is the subject of pending litigation involving Petitioner and Patent Owner: *DivX, LLC v. Amazon.com, Inc., and Amazon.com Web Services, Inc.*, No. 1:24-cv-02061 (Eastern District of Virginia, November 15, 2024) (the “parallel litigation”). I understand that Patent Owner sued Petitioner for alleged infringement of the 141 patent.

## **VII. OVERVIEW OF THE 141 PATENT**

### **A. Technical Background**

#### **1. *Progressive Playback Systems***

30. The 141 patent is directed to performing trick play operations (e.g., fast-forward, rewind, and seek) in a progressive playback video system. *See* EX1001, 2:23-39. Progressive playback was a well-known technique for playing video content as it was downloaded by the playback device, rather than waiting for the entire file to be downloaded. EX1008, 6. It works by having a server deliver video over a network to a playback device in small sequential chunks (i.e., portions of the video file). The playback device buffers each chunk as it arrives and plays it back once sufficient data is buffered.

#### **a. *Architecture of Progressive Playback Systems***

31. Progressive download and trick play were well known in the art before 2007. Progressive download and trick play require at least a remote server and a playback device, which requests and receives data from the server. EX1009, [0015];

*infra* §§ VII.A.1.b, VII.A.1.c. Known playback devices at the time included personal computers, mobile phones, and other devices that could communicate with a server over a network and play back received data. EX1010, 3:20-37; EX1004, 1:17-20; EX1001, 6:10-20. The playback device includes at least a processor for executing code, memory (e.g., a buffer) for storing received stream data, as well as mechanisms for implementing trick play functionality during playback. EX1011, 5:51-56 (explaining that the client device stores buffered data in memory); EX1012, 8 (describing floating controls enabling “functions like pause, play, fast-forward, and rewind”), 9 (describing system requirements, including processors); EX1005, 2:53-59 (describing a processor configured to decode and play back data); *infra* § VII.A.1.c.

32. To access content, a client playback device must establish a connection with a server via, for example, HTTP (Hypertext Transfer Protocol). EX1001, 6:38-46; EX1004, 1:46-52; EX1009, [0030]. HTTP is an application-layer protocol that facilitates communication across Internet-based systems. EX1030, 7. Because progressive download typically relied on connecting to traditional HTTP servers, it was also frequently referred to as HTTP streaming. Playback devices frequently use HTTP GET requests to retrieve multimedia content from the HTTP server. EX1030, 35. These requests can specify retrieval of an entire file, a specific byte range, or multiple byte ranges. *Id.*, 76, 85. HTTP also allows “extension of its request

methods” to enable additional features, like request cancellation (e.g. the AJAX “abort” function). EX1030, 1; EX1028, 118 (explaining that the abort command “stops the request”).

33. The use of HTTP enabled streaming systems to leverage existing Internet infrastructure, allowing multimedia streaming without the need for specialized hardware. EX1013, 4; EX1009, [0030]; EX1014, [0014]. Because remote servers may store numerous unrelated or undesired files, the playback device relies on the server to provide information about the available media content. EX1004, 6:59-62, 18:30-34; EX1015, [0033]. Thus, once the playback device establishes a connection with the server, the playback device may request from the server information describing the content stored on the server, and based on the received information, request from the server media content for playback on the playback device. EX1009, [0008]; EX1014, [0016].

**b. *Media Files***

34. Media may be segmented for storage on the server. Segmentation is a technique for dividing a file into smaller pieces, allowing it to be sent in manageable portions. EX1013, 4. For example, files using the RIFF (Resource Interchange File Format) format, like Microsoft’s AVI format, stored pieces of audio, video, and other information in separate “chunks” that could be retrieved and processed individually. EX1005, 1:54-65. Segmenting media content enables the playback

device to request byte ranges corresponding to specific chunks without needing to download the entire file. EX1013, 4.

35. To support efficient transmission from the server to the playback device, multimedia files are generally compressed through encoding. Raw, uncompressed multimedia files are often too large for rapid delivery over the Internet. EX1004, 1:46-52; EX1016, 64-65. To balance compression efficiency and playback fidelity, video data was often encoded in formats using I-, P-, and B-frames. EX1016, 64-65; EX1017, 10:66-11:6. A “frame” contains lines of video data and typically represents a single picture in the video. EX1029, 17.

36. Encoding generally involved removing redundancy by storing only the differences between consecutive frames. EX1016, 32, 62. In this method, most frames, including P- and B-frames, could only be decoded using information from surrounding frames. EX1016, 64-65; EX1017, 10:66-11:6. Only I-frames, or “key frames,” could be decoded independently. *Id.* Playback cannot begin from B- or P-frames because they can only be decoded using information from surrounding key frames. *Id.* Instead, playback must begin at a key frame, which serves as a valid entry point into the data stream. *Id.* As discussed below, index information requested by the playback device can identify the locations of these key frames, enabling almost instantaneous decoding and playback from the new position. EX1009, [0008]; EX1014, [0018]; *infra* IX.G, IX.H.

**c.     *Requests***

37.     A playback device may send a request to the server for information about the stored media content, as discussed above, or for a byte range of the stored media content. EX1009, [0010] (requesting byte ranges), [0020] (requesting metadata describing a multimedia file); EX1013, 4 (requesting byte ranges). The playback device can send synchronous requests or asynchronous requests to the server. EX1028, 92; EX1009, [0030]; EX1007, 7:60-64; EX1031, 2:23-42, 8:9-18.

38.     Asynchronous requests are requests that allow flexible communication between the client and the server. EX1007, 8:58-64. When a client issues an asynchronous request, the client can perform additional operations without waiting for a response from the server. EX1028, 112; EX1031, 2:32-42, 8:13-18. By contrast, synchronous requests require the client to wait for a response to each request before making additional requests or performing further operations. EX1028, 111; EX1031, 2:26-31.

39.     In streaming, asynchronous requests enable smooth playback that allows for user interaction and control. EX1007, 5:55-60. By removing delays, asynchronous requests enable progressive playback by allowing the client to obtain sufficient data to support smooth playback without relying on a sequential back-and-forth with the server. EX1007, 8:58-64, 2:15-23. Asynchronous requests also enable user control, like trick play, by providing a way for the playback device to

receive and process data as quickly as necessary to accommodate user feedback. EX1007, 6:18-22. The device may also cancel or flush previous requests to ensure that resources are allocated only to the most recent user command, thereby reducing delays. EX1007, 5:60-62, 6:52-61, 8:19-21.

**d. *Trick Play and Indexing***

40. As discussed above, trick play operations are user-controlled playback features, such as pause, fast-forward, skip, and fast-rewind, that replicate the interactivity of physical media players. EX1014, [0002]. Trick play functionality was well known prior to 2007. For example, Apple supported trick play features in its QuickTime 7, which supported “floating DVD-like controls for easy access to functions like pause, play, fast-forward, and rewind.” EX1012, 8.

41. To implement trick play operations, playback devices request index information from the server to identify and access relevant portions of a multimedia file. EX1014, [0016]; EX1009, [0010]; EX1018, 10:50-53, 13:35-37, 14:42-45. For example, the index information may include details such as byte offsets identifying positions where playback can resume after a trick play command, like the locations of key frames. EX1014, [0016]. Informing the playback device of key frame locations also enables operations like fast-forward or rewind because it allows the playback device to request and display only a series of key frames between the current position and a desired position in the video. EX1014, [0016]. Here, the user

would see a series of individual, intermediate pictures, thereby replicating what a VHS or DVD user might see while fast-forwarding or rewinding.

42. The playback device stores received data in a buffer, from which it may be played back. EX1004, 10:4-10; EX1010, 7:26-30; EX1019, 4:26-29. By managing the buffer to keep it full and progressively playing back each buffered segment, a playback device can cushion against potential transmission delays and maintain a smooth streaming experience. EX1020, 1:46-52. When the buffer contains too little data, the device may briefly pause playback to allow the buffer to refill before resuming. EX1004, 6:15-28, Figs. 10, 13. The playback device may also discard information in the buffer to prevent outdated data from being displayed after a trick play command. EX1032, 2:1-4, 4:6-9; EX1017, 13:49-61; EX1025, 17:1-9; EX1006, [0027].

43. The downloaded multimedia file includes video, audio, and subtitle data that would be synchronized during the download process. EX1033, [0005]. Once the desired audio or subtitle track is received, the playback device synchronizes it with the associated video content. EX1033, [0005]; EX1027, [0005], [0050]. Synchronization may involve the use of timestamps or time offset information to align playback accurately. EX1033, [0016], [0018]; EX1027, [0005]. Audio data may be interleaved with video data within the file, such that audio chunks are positioned adjacent to their corresponding video chunks. EX1005, 1:63-65;

EX1009, [0005]. Subtitle data may also be interleaved and can be presented either as text or as bitmap images. EX1026, [0031]; EX1005, 27:12-14. These subtitles are stored in a manner that allows them to be superimposed directly onto corresponding video frames. EX1026, [0040].

**e.     *Encryption and Digital Rights Management***

44. Many playback systems use digital rights management (DRM) to protect digital content by restricting access to authorized users. EX1021, 1:36-58. DRM commonly includes encryption to convert media content into secure, unreadable data that can only be accessed using a decryption key. EX1021, 2:47-51. In many encryption schemes, the same key is used to both encrypt and decrypt data, rendering encryption key and decryption key synonymous. EX1021, 6:51-54. By restricting access to those with the appropriate key, encryption protects files from unauthorized access. EX1024, 1:27-30; EX1034, [0008]; EX1005, 51:14-20.

45. The media content of the multimedia files may be either fully or partially encrypted. EX1022, 9:16-20, 9:26-27, 9:41-44. For example, the video data of the multimedia file may be partially encrypted such that only portions of certain frames, such as portions of key frames, are encrypted. EX1022, 13:21-30 (describing multiple methods of partial-frame encryption). Partially encrypting a frame, or portions of a key frame, prevents access to all dependent frames that

require the key frame for decoding, thereby reducing the processing overhead required for decryption. EX1022, 9:34-36; 9:45-49; 13:21-30.

46. One method of implementing DRM involves the use of DRM metadata, like DRM headers. EX1005, 27:64-28:2; EX1021, 7:58-64. The DRM header, which is stored alongside the multimedia file in a specialized metadata chunk, may contain information related to the key necessary to decrypt portions of video data. EX1005, 28:1-15. For example, the DRM header might identify the key required to decrypt encrypted portions of frames. EX1005, 28:2-4. Additional metadata, like certain chunks in RIFF files, may contain information identifying the encrypted portions of video frames. EX1005, 27:64-67.

47. As an additional layer of protection, the DRM header itself may be encrypted, requiring a separate decryption key to access the information it contains. EX1005, 52:44-49. When the DRM header is encrypted, the playback device must first obtain from the server a decryption key to decrypt the header. EX1005, 52:44-46. Once decrypted, the header provides the information necessary to identify and decrypt the protected portions of the multimedia file. EX1005, 52:49-51, Fig. 4. This process may be repeated as needed until all encrypted segments have been decrypted. EX1005, 53:9-14.

48. Because encrypted data cannot be decoded for playback, the device must decrypt the multimedia before decoding it. EX1023, 3:62-65. Accordingly,

when a multimedia file is encrypted, decryption must occur before playback can begin.

## **B. Patent Description**

49. The 141 patent is directed to a progressive playback system that uses a client application to support “trick play” functions. As explained in the 141 patent, prior progressive playback systems could play back content as it is downloaded from a remote server. EX1001, 1:32-35.

50. The 141 patent explains, however, that prior progressive playback systems typically only support short video clips because they typically download files linearly. EX1001, 1:40-45, 2:1-3. Because prior art systems download files linearly, they cannot request data out of order, which means they typically do not support “random seeking, trick-play, and playback of remotely stored longer content such as feature length movies.” EX1001, 1:51-55, 2:10-19 (explaining that trick play functions could include rewinding, fast forwarding, and skipping between scenes that may not have been downloaded from the remote server).

51. The 141 patent attempts to overcome these limitations by using index information referencing data within the stored file that allows the playback system to request specific byte ranges within the file from a remote server. EX1001, 2:48-52; EX1002, 95 (Response to OA dated February 20, 2019). However, it was well-known to a POSITA that playback systems could support trick play

functionality on partially downloaded media files using index information and requests for specific byte ranges of data. *Supra* § VII.A.1.d.

**C. Priority Date**

52. The 141 patent was filed as a continuation of application No. 15/682,379 filed on August 21, 2017, which is a continuation of application No. 14/632,670 filed on February 26, 2015, now Pat. No. 9,794,318, which is a continuation of application No. 12/982,413 filed on December 30, 2010, now Pat. No. 8,977,768, which is a continuation of application No. 11/970,493 filed on January 7, 2008, now Pat. No. 7,886,069. EX1001, Related U.S. Application Data. The 141 patent also claims priority to Provisional Application No. 60/883,659, filed on January 5, 2007. EX1001, Cover. I understand that the Petitioner does not dispute that the priority date is January 5, 2007 for the purposes of this IPR.

**D. Level of Ordinary Skill in the Art**

53. I understand that the 141 patent should be interpreted from the perspective of a person having “ordinary skill in the art” as of the effective filing date of the patent. For the purposes of this declaration, I assume that the effective filing date of the patent application leading to the 141 patent is no earlier than January 5, 2007, and I understand that Patent Owner has not claimed an earlier priority date in the parallel litigation. I am familiar with the technological field at issue and the state of the art as of that date. I understand that the education level of

those in the technological field, sophistication of the technology, types of problems encountered, prior art solutions to those problems, and the speed at which innovations are made in the field may factor into establishing what the appropriate level of ordinary skill should be.

A POSITA would have a bachelor's or master's degree in electrical engineering, computer engineering, or a related field, with specialized training or coursework in digital video transmission and playback and related technologies. The individual would have at least four years of practical experience in digital video transmission and playback, including software and/or hardware architectures used to perform digital video transmission and playback. A person could also have qualified as a POSITA with more formal education and less technical experience, or vice versa. I had at least the requisite level of skill with respect to the subject matter of the challenged patent at the time of the effective filing date as demonstrated by my qualifications listed above and my CV (EX1035).

## **VIII. PRIOR ART OVERVIEW<sup>2</sup>**

### **A. Hagai**

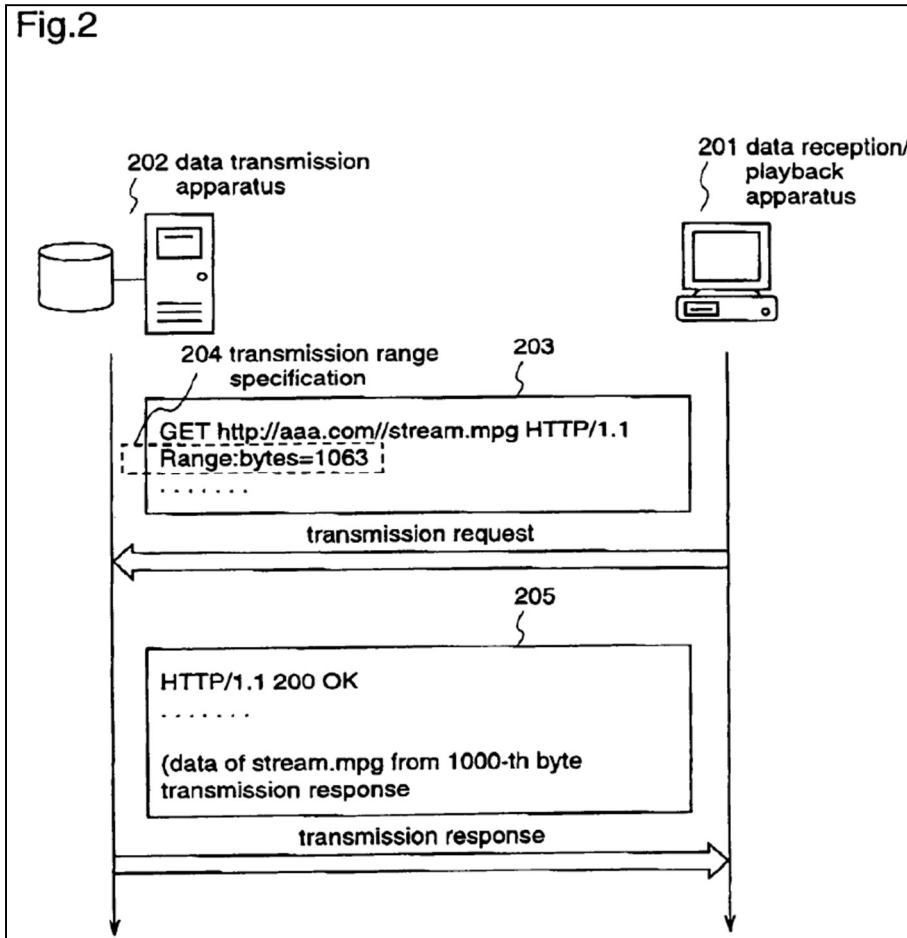
54. U.S. Patent No. 7,051,110 to Makoto Hagai et al., titled "Data Reception/Playback Method and Apparatus and Data Transmission Method and

---

<sup>2</sup> I have been informed that pre-AIA 35 U.S.C. §102 applies to the 141 patent.

Apparatus for Providing Playback Control Functions” was filed on December 20, 2000, published on August 9, 2001, and issued on May 23, 2006 (“Hagai”) (EX1004). EX1004, Cover. I understand that Hagai is prior art because its filing, publication, and issuance dates predate the priority date of the 141 patent. Hagai was not cited by the applicant or examiner during prosecution of the 141 patent. EX1001, References Cited.

55. Hagai is directed to “methods and apparatuses for transmitting, receiving, and playing audio data or video data (moving pictures, still pictures, graphics, etc.) through a network.” EX1004, 1:9-12. Specifically, Hagai discloses a data reception/playback apparatus (“playback device”) that communicates with a data transmission apparatus (“server”) over the Internet using HTTP. EX1004, 1:25-28, 3:62-4:5, Fig. 2. Hagai discloses examples of devices that a POSITA would have known as a playback device, such as a personal computer, mobile terminal, or handy phone. EX1004, 1:17-20.



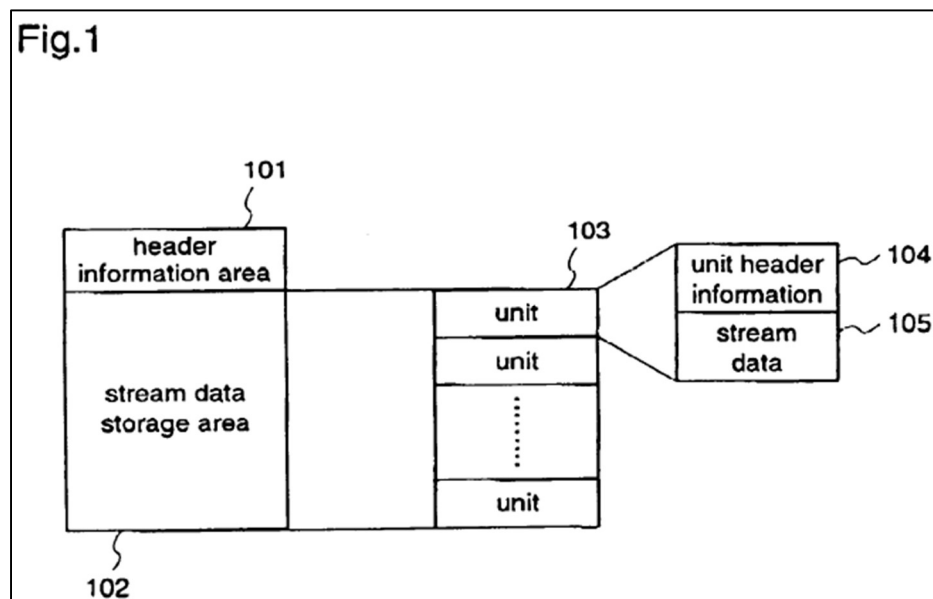
56. The playback device uses HTTP to send a transmission request 1804 identifying the desired media file or portions of the media file. EX1004, 9:17-23, Fig. 2. In response, the server retrieves the requested data and sends a transmission response 1805 with the requested stream data. EX1004, 10:4-7. The playback device then plays back the received data. EX1004, 10:8-10.

57. Hagai's playback device includes a control means 1801 configured to instruct the transmission request sending means 1803 to send requests to the server for data related to the media files stored on the server and stream data, including a playback start position table 404. EX1004, 9:17-19. After the playback device

receives a response from the server, the playback start position table extraction means 403 extracts the table from the received data and stores it in the playback start position table buffer 405. EX1004, 9:37-41. Subsequent requests made by the transmission request sending means 1803, such as for byte ranges of selected tracks, are informed by the playback start position selection means 407, which selects a start position from the playback start position table 404. EX1004, 9:41-46. When multiple playback start positions are stored in the playback start position table 406, the playback start position selection means 407 can select a start position using any suitable method, such as selecting the position closest to that specified by the user. EX1004, 9:47-54.

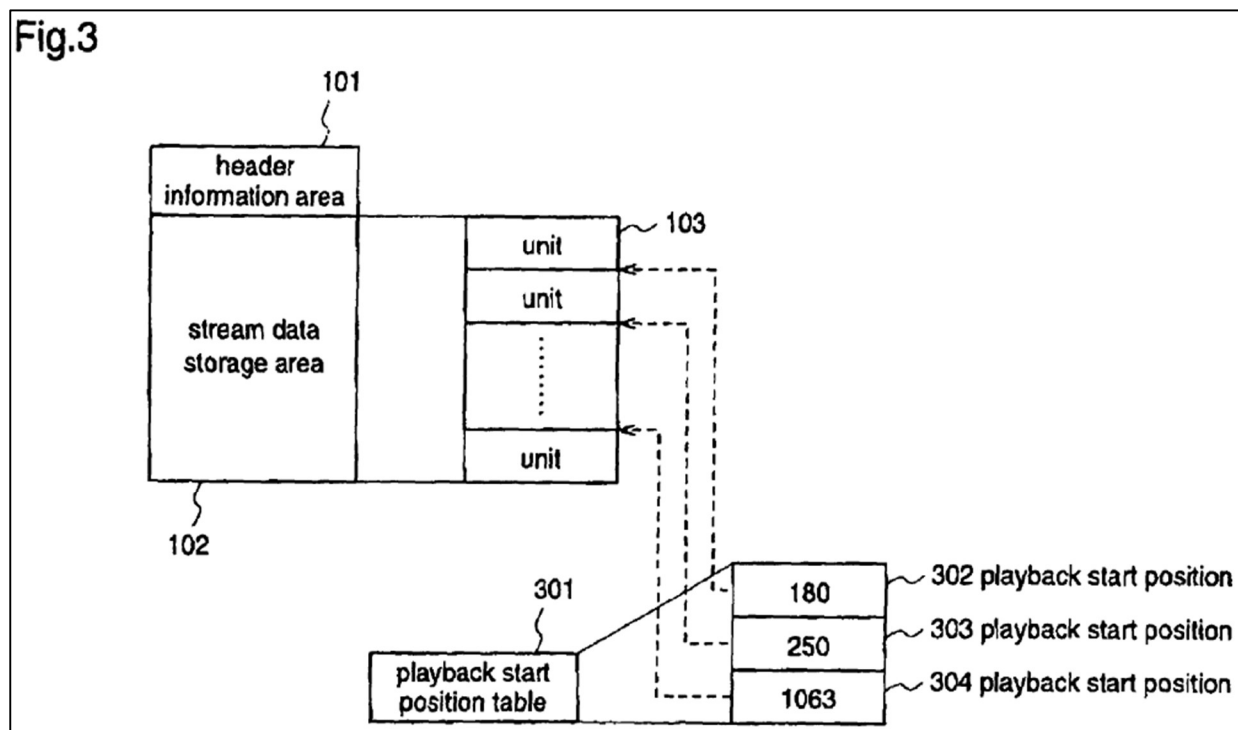
58. The playback device also includes a reception buffer 1808 that is configured to store received stream data, and a buffer monitor means for monitoring the quantity of buffered data to minimize latency during playback. EX1004, 6:7-14. For example, when the playback device receives requested stream data, it stores the stream data in reception buffer 1808 in preparation for playback on the playback device. EX1004, 10:4-10. Prior to playback, the buffer monitor means monitors the quantity of data stored in the buffer. EX1004, 6:10-14, 16:48-51. When the buffer drops below a threshold level of data, the playback device will halt playback until the data exceeds a predetermined value. EX1004, 6:21-27.

59. As noted above, the server stores media files and metadata related to the audio and video content of the stored media files for retrieval by the playback device. EX1004, 6:49-65, Fig. 15. Hagai discloses that these media files contain both audio and video. EX1004, 1:53-58, 2:56-58, 4:13-16. The stored media files include a header information area 101 and a stream data storage area 102. EX1004, 8:11-18, Fig. 1.



60. The header information area 101 is positioned at the beginning of the file and indicates the types of media content stored in the units of the storage area 102. EX1004, 8:16-18, Fig. 1. The stream data storage area 102 is divided into a plurality of units 103, and each unit includes unit header information 104 that identifies information about the stream data 105 for playback (e.g., data size information and playback time information). EX1004, 2:11-18, Fig. 1.

61. According to one embodiment, the playback start position table 301 is also stored on the server, in a separate file from the stream data, and identifies playback start positions, such as playback start positions 302, 303, and 304. EX1004, 8:41-45, 8:52-59. Each playback start position 302, 303, and 304 indicates the number of bytes from the head position of the stream data storage area 102 to the respective start position. EX1004, 8:55-59, Fig. 3.



62. Hagai further discloses that the server stores a playback-speed/file mapping table that identifies available files and the playback speeds of the tracks contained within. EX1004, 18:34-44. Column 1601 of Fig. 16 identifies the playback speeds of the file and column 1602 identifies the available files stored on the server. EX1004, 18:34-44, Fig. 16.

**Fig.16**

| 1601<br>{<br>playback speed | 1602<br>{<br>file name |
|-----------------------------|------------------------|
| 1.0                         | stream.mpg             |
| 2.0                         | stream_s2.mpg          |
| 3.0                         | stream_s3.mpg          |

63. For example, the file name “stream\_s2.mpg” in column 1602 of Fig. 16 refers to a file containing audio and video content that has a playback speed double that of “normal playback speed.” EX1004, 18:14-20, 5:8-11, Fig. 16.

64. Hagai discloses five embodiments generally directed to a method for receiving and playing audio and video stream data through a network. EX1004, 1:9-12. For example, the first and second embodiments both relate to a foundational method of multimedia playback from arbitrary points in the stream. EX1004, 8:1-4. However, the first embodiment is directed to situations where the playback start position table and stream data are stored in separate tracks, whereas the second embodiment relates to situations where the playback start position table and stream data are stored in the same track. EX1004, 10:28-34. The third embodiment

describes how the playback device can request to halt and restart the transmission of playback data from the server. EX1004, 15:11-16:2. The fourth embodiment describes that the playback device monitors the levels of stored data in the reception buffer. EX1004, 16:24-30, 16:59-17:46. The fifth embodiment describes that the server stores a playback-speed/file mapping table, and that the playback device uses the table for video track selection. EX1004, 17:55-59. A POSITA would have understood that the third, fourth, and fifth embodiments describe additional operations, and that each of them can be combined with the first or second embodiments to add additional functionality. EX1004, 4:6-12.

**B. Li**

65. U.S. Patent No. 8,731,369 to Adam H. Li et al., titled “Multimedia Distribution System for Multimedia Files Having Subtitle Information” was filed on December 17, 2004, published on September 22, 2005, and issued on May 20, 2014 (“Li,” EX1005). I understand that Li is prior art because its filing and publication dates predate the priority date of the 141 patent. I also understand that the examiner cited Li (referred to there as “Zoest”) as the primary reference in the Office Action dated November 19, 2018. EX1002, 17-20, 90-97 (Office Action dated November 19, 2018 and Response to Office Action dated February 20, 2019). I further understand to overcome the Office Action, the applicant amended the claims to distinguish over the other prior art used in the examiner’s combination and argued

there was no motivation to combine. I understand that here, Petitioner relies on a new combination of references, with Hagai as the primary reference.

66. Li discloses a system for “encoding, transmission and decoding of multimedia files that can include” multiple tracks. EX1005, 1:19-22. Li’s playback device includes a processor configured to extract meta data, locate video and audio chunks, decode video, identify corresponding subtitle and audio tracks, and superimpose subtitle text. EX1005, 2:26-67, Claims 1-8. The processor executes these tasks based on stored instructions for on-demand video playback. EX1005, 5:42-44, 27:30-31.

67. Li also discloses that the multimedia file is structured as a RIFF file composed of data chunks. EX1005, 12:46-63. The file includes a DMNU chunk, containing menu objects describing available audio and subtitle tracks (EX1005, 13:18-25, 23:67-24:4), and a DRM header, which contains decryption keys needed to decrypt encrypted content. EX1005, 27:67-28:2. The DRM header also “contains information identifying the master key,” which is used to encrypt the other decryption keys. EX1005, 28:2-4.

68. Li teaches that multimedia files can contain multiple video, audio, and subtitle tracks, and that playback involves selecting from among them. EX1005, 11:66-12:1, 24:27-29, 52:29-40. Li’s DMNU chunk includes AudioTrack and SubtitleTrack chunks that identify available subtitle and audio tracks. EX1005,

23:67-24:9. These chunks accompany portions of video identified in corresponding MediaTrack chunks. EX1005, 24:2-4, 25:27-30. The DMNU chunk also includes AudioSelectAction and SubtitleSelectAction chunks, which enable selection of user-desired tracks. EX1005, 24:19-31.

69. Li also supports as the use of digital rights management (“DRM”). EX1005, 27:42-56. For example, Li’s multimedia files may be partially encrypted. EX1005, 27:62-64. When the file is encrypted, Li’s multimedia file contains “DRM chunks,” which contain references to the encrypted portions of video and the keys used to decrypt them. EX1005, 27:64-67. DRM chunks may include frame identifiers, encryption status, byte offsets, lengths, and references to decryption keys for encrypted blocks. EX1005, 28:5-15, Fig. 2.9. The keys needed to decrypt the content are stored in a DRM header. EX1005, 27:67-28:2. The DRM header itself may also be encrypted, in which case a separate encryption key is requested and received to decrypt the DRM header. EX1005, 52:44-49.

### **C. Park**

70. U.S. Patent Application Publication No. 2005/0198681 to Daniel John Park, titled “Playout Buffer Management To Minimize Startup Delay,” was filed on March 8, 2004 and published on September 8, 2005 (“Park,” EX1006). EX1006, Cover. I understand that Park is prior art because its filing and publication dates

predate the priority date of the 141 patent. Park was not disclosed to the Patent Office during prosecution of the 141 patent. EX1001, References Cited.

71. Park discloses techniques for “managing a playout buffer to minimize delays” in response to user fast forward, reverse, and pause commands. EX1006, [0001], [0012]. For example, Park discloses a method of playout buffer management to flush audio and video data that is stored in the playout buffer in response to a user command, such as a fast forward or reverse, or a change in AV program. EX1006, [0012], [0026], Claim 7. Park explains that flushing the playout buffer minimizes playout delay by discarding outdated audio and video data. EX1006, [0023], [0027].

#### **D. Schmitz**

72. U.S. Patent No. 5,544,318 to Patrick Schmitz et al., titled “Asynchronous Media Server Request Processing System For Servicing Reprioritizing Request From a Client Determines Whether Or Not To Delay Executing Said Reprioritizing Request,” was filed on April 16, 1993, and issued on August 6, 1996 (“Schmitz,” EX1007). EX1007, Cover. I understand that Schmitz is prior art because its filing and issuance dates predate the priority date of the 141 patent. Schmitz was not disclosed to the Patent Office during prosecution of the 141 patent. EX1001, References Cited.

73. Schmitz is directed to navigation through multimedia content. EX1007, 6:18-22. Schmitz proposes a method of using asynchronous requests to

enable arbitrary navigation through content rather than imposing linear constraints. EX1007, 5:63-65. To reduce unnecessary bandwidth use, Schmitz discloses flushing or cancelling any pending requests made obsolete when a user navigates to a new playback location. EX1007, 1:9-14, 6:62-65, Abstract.

74. While the technique of Schmitz has special application to non-linear media (like encyclopedias), it expressly states that its techniques also apply to time-linear media, like video, that is ordinarily played back sequentially. EX1007, 1:67-2:7, 6:1-10. In Schmitz, time-linear media refers to formats such as video tapes, which require synchronous, sequential playback and do not permit instant seeking to arbitrary positions. EX1007, 1:67-2:7. Schmitz proposes an improvement by using asynchronous requests to enable navigation to arbitrary playback locations—an approach that departs from prior systems constrained by the physical and sequential limitations of traditional linear media. Asynchronous requests allow the playback device to perform other operations or make additional requests while prior requests are pending, enabling users to navigate throughout media without waiting for forced synchronization between client and server. EX1007, 6:15-22, 5:18-21. In effect, Schmitz enables random access to media content. EX1007, 5:63-65.

75. Schmitz also discloses flushing queued requests. EX1007, 1:9-14. For example, Schmitz discloses a multimedia system in which the client may cancel queued media requests that become obsolete when a user navigates to a new

playback location. EX1007, 1:9-14; Abstract. This cancellation improves performance by discarding requests for content the user no longer intends to view, thereby reducing unnecessary data processing and bandwidth use. EX1007, 6:62-65.

**IX. GROUND 1: Claims 1, 3-8, 10, 20, 22-27, and 29 are obvious over Hagai, Li, and General Knowledge of a POSITA**

**A. A POSITA would have combined Hagai and Li**

76. A POSITA would have been motivated to combine Hagai and Li. Both references relate to the transmission and processing of multimedia content for playback.

77. Hagai relates to “methods and apparatuses for transmitting, receiving, and playing audio data or video data (moving pictures, still pictures, graphics, etc.) through a network.” EX1004, 1:9-12. A POSITA would have understood that audio data and video data are both components of multimedia content. Hagai describes known ways of building systems for transmission and playback, including a “data transmission apparatus,” or “HTTP server,” which transmits audio and video to a “data reception/playback apparatus,” or “playback device.” EX1004, 1:26-29, 1:53-58, Figs. 18-19. These systems could transmit multimedia content from servers over a network to devices for playback. EX1004, 2:56-58. However, these prior systems could not handle certain “playback control functions,” including playback from an arbitrary point in the video. EX1004, 3:1-13. Hagai provides a way of adding these playback functions, including by using information affiliated with the

multimedia content—metadata—to inform requests for playback from specific positions. EX1004, 4:13-32.

78. Li also relates to distributing and using multimedia files in a system where the multimedia is transmitted from a server over a network to a playback device. EX1005, Abstract, 1:18-22, 12:9-18, Fig. 1. As Li explains, it was known that decoding a multimedia file for playback, such as when “seek[ing] to a particular portion of the multimedia information,” required accessing certain related information during the decoding and playback process. EX1005, 51:60-64, 53:15-20. For example, the device may use “DMNU” chunks that descriptive metadata about media tracks, such as language options and references to information defining how the device may select corresponding audio and subtitle tracks. EX1005, 8:10-16, 8:53-59, 23:47-50, 24:57-61, Fig. 2.5. Further, Li describes DRM information stored within the multimedia file, such as that stored in “DRM chunks,” to decrypt encrypted video data for playback. EX1005, 27:42-28:15.

79. A POSITA would have been motivated to combine Hagai such that the DMNU chunks of Li would be used by Hagai to provide the playback device with the necessary information regarding the stored audio tracks, video tracks, and subtitle tracks.

80. A POSITA would have been further motivated to combine Hagai’s multimedia playback system with Li’s disclosure of how various additional metadata

may be used for decoding and playback, including for DRM-protected multimedia. As noted above, Hagai already discloses using metadata during the video decoding and playback process, which was widely known. *Supra* §§ VII.A.1.e, VII.A.1.d, VIII.A. The combination would incorporate the additional information disclosed by Li in the same places where Hagai discloses this information may be stored. Since Hagai's system already uses metadata during the decoding and playback process, it would have been straightforward to add Li's functionality to take advantage of additional information sourced from the same places.

81. A POSITA would have had a reasonable expectation of success in making this combination because it merely adds information to the metadata already affiliated with a multimedia file and additional processing based on this information. The combination of these teachings would have yielded predictable results because it is no more than a combination of conventional, well-known features in video transmission and playback systems.

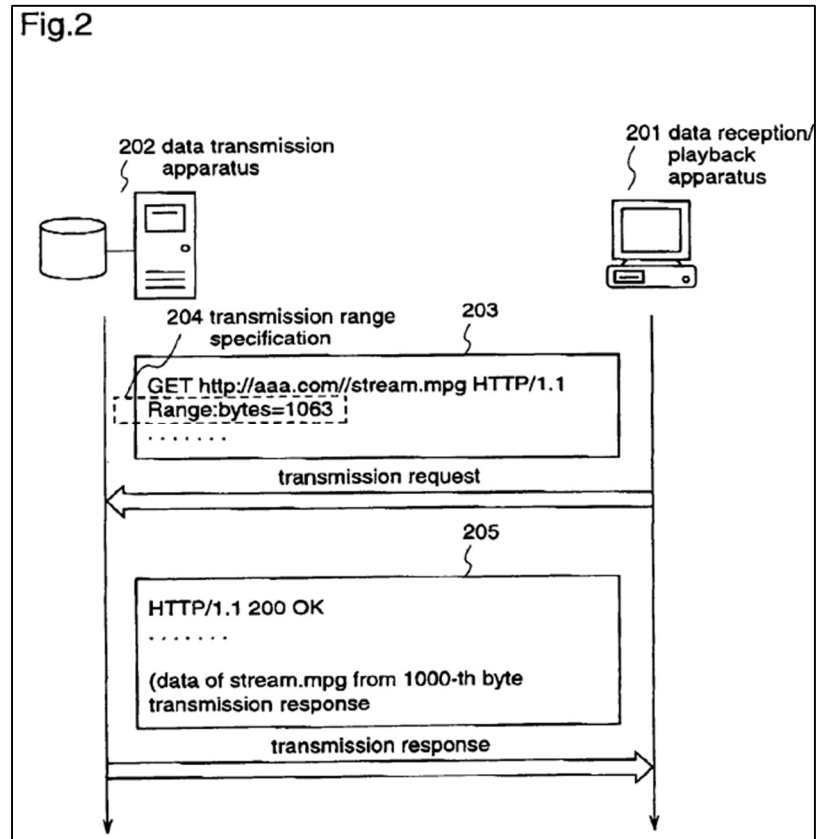
**B. Claim 1**

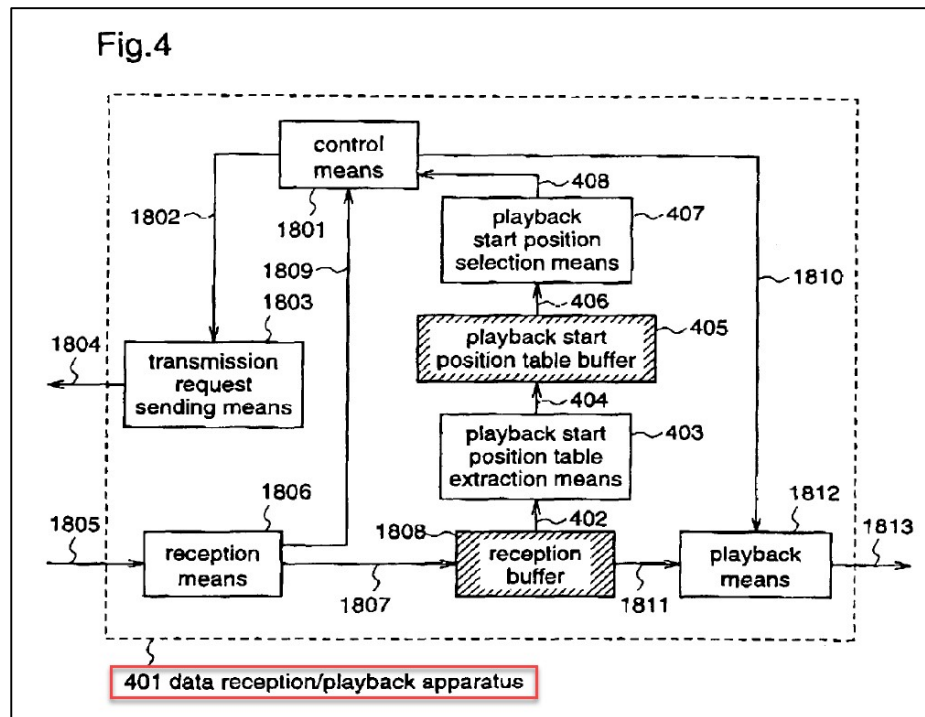
**1. [1.pre] “A *playback device, comprising:*”**

82. To the extent the preamble of claim 1 is limiting, Hagai renders obvious this limitation.

83. Hagai discloses a data reception/playback apparatus 201, 401 (“playback device”) that requests audio or video from a data transmission apparatus

202 (“server”) using HTTP. EX1004, 2:55-57, 2:59-67, 9:4-10:10 (explaining that the playback apparatus “plays the stream data” and “outputs the played data as an output signal 1813.”), Figs. 2, 4.





84. Hagai further provides examples of devices that a POSITA would have known could serve as playback devices, including “devices which are connectable to the Internet, such as personal computers, mobile terminals, handy phones and the like.” EX1004, 1:17-20, Figs. 2, 4; *supra* § VII.A.1.a. A POSITA would have understood that these disclosed devices could be playback devices because they can communicate with a server using HTTP and can play audio and/or video content received from a server. *Supra* § VII.A.1.a.

2. [1.a] “a processor; and”

85. Hagai renders obvious limitation [1.a].

86. Hagai discloses that the playback device comprises “control means 1801,” “transmission request sending means 1803,” “reception means 1806,”

“playback means 1812,” “playback start position table extraction means 403,” and “playback start position selection means 407.” EX1004, 9:1-14, Figs. 2, 4, 17, 18. A POSITA would have understood that the “means” disclosed in Hagai would be implemented by a processor because the functions they perform—such as sending requests, receiving data, issuing commands, and initiating playback—were conventional operations typically executed by processors in multimedia playback systems, as discussed further below.

87. As explained in Section IX.B.1, Hagai discloses a playback device that can receive and playback multimedia content. EX1004, 9:4-10:10, 10:8-10, Figs. 2, 4; *supra* §§ IX.B.1 (Limitation [1.pre]), VII.A.1.a. A POSITA would have understood that the playback device must include a processor to receive and playback multimedia content. Furthermore, it was well-known to a POSITA that each of the exemplary playback devices disclosed in Hagai, including “personal computers, mobile terminals, handy phones and the like,” would have a processor. EX1004, 1:17-20; *supra* §§ IX.B.1, VII.A.1.a.

**3. [1.b] “a non-volatile storage containing an application for causing the processor to perform the steps of:”**

88. Hagai renders obvious limitation [1.b].

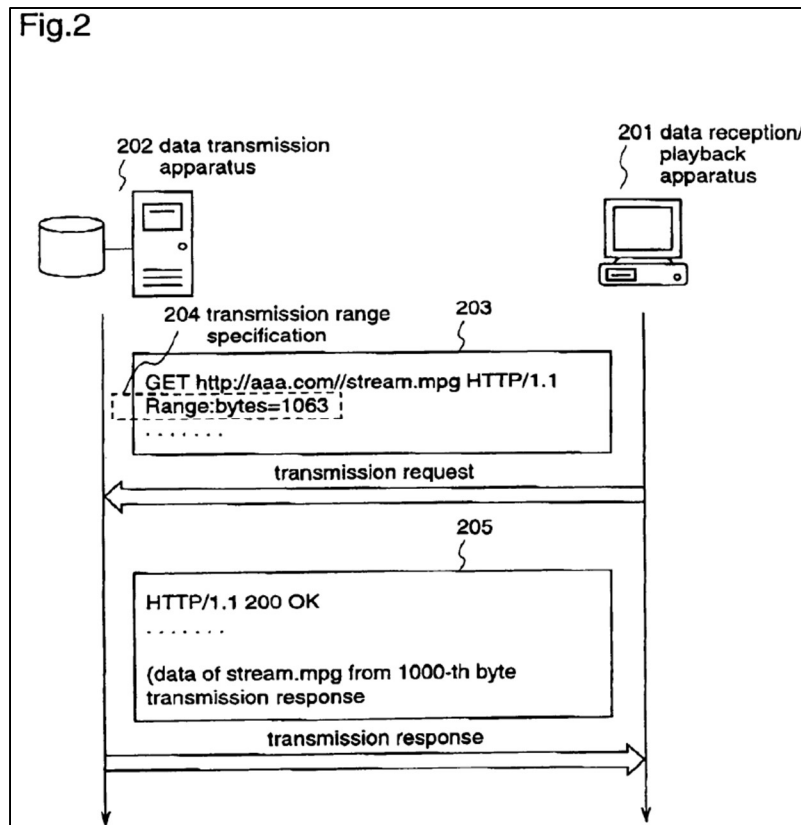
89. As explained above in limitations IX.B.1 and IX.B.2, a POSITA would have understood that Hagai renders obvious a playback device with a processor. EX1004, 1:17-20; *supra* §§ IX.B.1 (Limitation [1.pre]), IX.B.2 (Limitation [1.a]),

VII.A.1.a. A POSITA would also have known that such processors execute applications stored in a non-volatile storage to perform certain steps, including the ones explained below in Sections IX.B.4 - IX.B.19. It was a well-known and routine practice at the time to store playback and control applications in non-volatile storage to ensure operational readiness and data persistence.

4. ***[1.b(i)] “establishing at least one connection for communicating with a remote server system;”***

90. Hagai discloses limitation [1.b(i)].

91. Hagai discloses that the playback device communicates with the server using Hypertext Transfer Protocol (HTTP). EX1004, 1:25-28, 8:19-27.



92. A POSITA would have understood that the playback device must establish a connection with a remote server system, such as the server, because the playback device transmits multiple data requests to, and receives data from, the server. For example, as discussed *infra* in at least Sections IX.B.5 (Limitation [1.b(ii)]), IX.B.7 (Limitation [1.b(iv)]), IX.B.9 (Limitation [1.b(vi)]), and IX.B.11 (Limitation [1.b(viii)]), the playback device obtains a playback-speed/file mapping table (*see, e.g.*, EX1004, 18:34-38, 52-62) and a playback start position table (*see, e.g.*, EX1004, 8:41-48, 9:17-19) from the server, and transmits requests to the server for stream data from a start position. *See, e.g.*, EX1004, 9:63-10:4. A POSITA would have understood that such communications between the playback device and the server could not occur without first establishing a connection between the playback device and server. *Supra* § VII.A.1.a.

93. Additionally, a POSITA would have understood that the server is a remote server system because the playback device communicates with the server over the Internet, and the playback device sends requests to, and receives responses from, the server using an HTTP connection. EX1004, 1:16-25, 1:30-2:14; *supra* § VII.A.1.a.

**5. [1.b(ii)] “obtaining information from a remote server system describing at least one video track, multiple audio tracks, and multiple subtitle tracks;”**

94. Hagai in combination with Li renders obvious limitation [1.b(ii)].

95. Hagai discloses “obtaining information from a remote server system.”

For example, Hagai explains that the server transmits to the playback device a playback-speed/file mapping table in advance of transmitting the streaming data file:

The playback-speed/file mapping table may be transmitted from the data transmission apparatus to the data reception/playback apparatus in advance of data transmission, or the data reception/playback apparatus may be provided with the mapping table. In this case, the data reception/playback apparatus selects a file to be requested for transmission, and conducts a transmission request for the selected file.

EX1004, 18:34-38, 18:58-62.

96. The information obtained in this table describes multiple video tracks that can be transmitted from the server by identifying their playback speeds and files containing them. EX1004, 18:38-45. For example, as illustrated in Figure 16, the file name “stream\_s2.mpg” has a playback speed of “2.0,” which indicates that the video and audio tracks of the “stream\_s2.mpg” file have a playback speed double that of “normal playback speed.” EX1004, 5:8-11, 18:14-20, Fig. 16.

**Fig.16**

| 1601<br>{<br>playback speed | 1602<br>{<br>file name |
|-----------------------------|------------------------|
| 1.0                         | stream.mpg             |
| 2.0                         | stream_s2.mpg          |
| 3.0                         | stream_s3.mpg          |

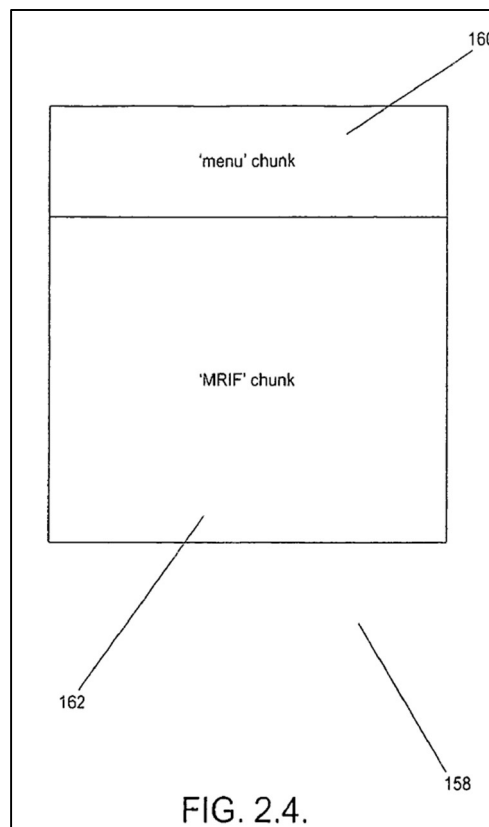
97. Thus, a POSITA would have understood that the playback-speed/file mapping table includes information describing at least one video track and multiple audio tracks because it provides the speed at which the playback device plays the video and/or audio tracks of each multimedia file (e.g., stream\_s2.mpg) available from the server.

98. To the extent that Hagai does not disclose information describing at least one video track, multiple audio tracks, and multiple subtitle tracks, Li does. For example, Li discloses a multimedia file that includes “PlayAction” chunks, which describe available tracks and define how to select them using references to a “MediaTrack” chunk. EX1005, 24:2-9, Fig. 2.5.

99. Li discloses a multimedia file stored on a server structured as a RIFF (Resource Interchange File Format) file. EX1005, 51:50-58 (describing that the

multimedia file is located at a first or second server), 12:46-53. Li explains that a RIFF file typically includes an 8-byte header to identify the file. *Id.*, 12:52-63. The remainder of the file comprises chunks of data, whereby each chunk includes an 8-byte header to identify the file and the length in bytes of the data following the header. *Id.* Li explains that a RIFF file stores multimedia data and associated metadata in chunks and lists, which may also be called “list chunks.” *Id.*

100. One RIFF chunk of the media file is the “DMNU” chunk 158, which includes a menu chunk 160 and a MRIF chunk 162. EX1005, 13:39-46, 14:62-67, 23:3-6, Fig. 2.4.



101. Li further discloses that the menu chunk 160 “contains the information necessary to construct and navigate through the menus.” EX1005, 23:6-8. The menu chunk 160 further includes a “WowMenu” chunk 175 that “contains the hierarchy of menu chunk objects,” containing references to media tracks. *Id.*, 23:13-15, 23:30-35, Fig. 2.5. Li explains that “objects” contain information describing a subject, which can be “anything having characteristics capable of description,” like a multimedia file. *Id.*, 7:60-67, 8:37-43, 15:19-21. These chunks include descriptive metadata about media tracks, such as language options and references to media chunks—information defining how the device may select corresponding tracks. *Id.*, 8:10-16, 8:53-59, 23:47-50, Fig. 2.5.

102. Each WowMenu chunk 175, as illustrated below in Fig. 2.5 includes “a number of ‘ButtonMenu’ chunks 176” that “can describe such things as the overlay to use when the button is highlighted by the user, the name of the button and what to do in response to various actions performed by a user navigating through the menu.” EX1005, 23:41-47, Fig. 2.5 (annotated). The “Action” chunk 178, of the WowMenu chunk, defines the responses when a button is selected.

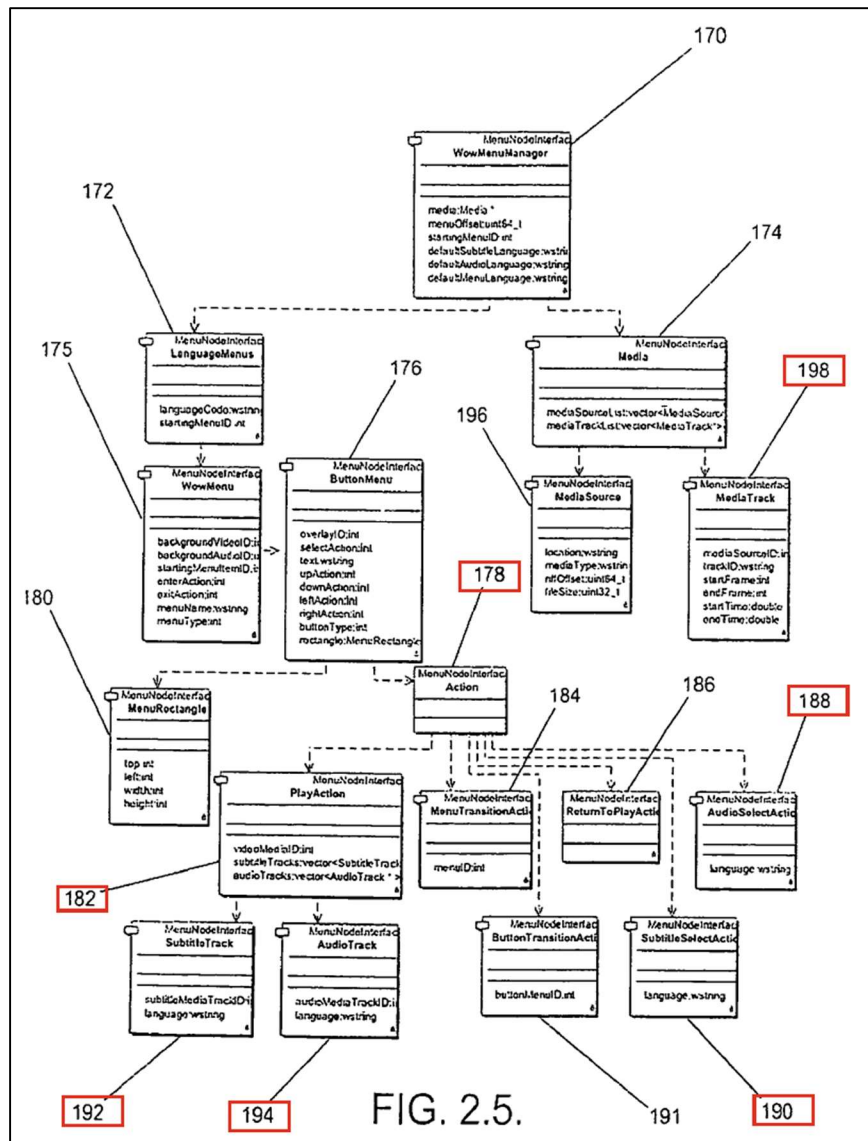


FIG. 2.5.

103. One of Li's menu chunk objects is the "PlayAction" chunk 182. EX1005, 23:58-63, Fig. 2.5 (annotated). The PlayAction chunk 182 identifies the video, audio, and subtitle tracks within the multimedia file, and identifies the corresponding audio and subtitle tracks "using 'SubtitleTrack' 192 and 'AudioTrack' 194 chunks." *Id.*, 23:63-24:2. The SubtitleTrack and AudioTrack chunks contain references to a "MediaTrack" chunk 198, which identifies portions

of multimedia tracks. *Id.*, 24:2-4 (“The ‘SubtitleTrack’ and ‘AudioTrack’ chunks both contain references to a ‘MediaTrack’ chunk 198.”), 24:43-45. These references allow selection of, for example, audio and subtitle tracks corresponding to a selected video track. *Id.*, 25:27-33.

104. The menu chunk objects also include the “SubtitleSelectAction” chunk 190 and “AudioSelectAction” chunk 188, each of which contains information that can be used to select from among available tracks, which may be in externally referenced files. EX1005, 24:19-31. When audio and subtitle tracks are in external files, they may be retrieved and synchronized with corresponding video tracks stored on the server. *Id.*, 53:36-47.

105. A POSITA would have understood that these chunks (i.e., SubtitleTrack chunk 192 and AudioTrack chunk 194) describe multiple audio tracks and multiple subtitle tracks because these chunks identify the tracks available for selection by the playback device and provide information necessary to locate and select them in response to user preferences as described below:

The ‘PlayAction’ chunk identifies audio and subtitle tracks using ‘SubtitleTrack’ 192 and ‘AudioTrack’ 194 chunks.

EX1005, 23:67-24:2.

Each ‘AudioSelectAction’ chunk 188 contains information that can be used to select a particular audio track. In one embodiment, the audio track is selected from audio tracks contained within a multimedia file in accordance with an embodiment of the present invention. In other

embodiments, the audio track can be located in an externally referenced file.

Each 'SubtitleSelectAction' chunk 190 contains information that can be used to select a particular subtitle track. In one embodiment, the subtitle track is selected from a subtitle contained within a multimedia file in accordance with an embodiment of the present invention. In other embodiments, the subtitle track can be located in an externally referenced file.

EX1005, 24:19-31, Fig. 2.1.

106. The 141 patent directly refers to Li's menu chunks as providing the descriptive information necessary for track selection in response to user preferences, as described below:

Media files such as the media files described in U.S. patent application Ser. Nos. **11/016,184** and 11/198,142, incorporated by reference above include menu information and/or information from multiple media sequences (i.e. distinct media presentations).

EX1001, 9:49-53 (emphasis added).

When the media is formatted in an AVI format or in a format similar to any of the file formats described in U.S. patent applications Ser. Nos. **11/016,184** and 11/198,142, information concerning each of the media sequences can be downloaded by downloading the RIFF header for each media sequence. Once information concerning the media sequences has been obtained, a selection (154) can be made concerning the media sequence that is to be played. When the media file contains a single media sequence, the decision can be automatic. When the media file contains multiple sequences, the decision can be made based upon a user instruction that is obtained via a menu interface generated using menu information obtained from the remote media file by the file parser.

EX1001, 10:14-27 (emphasis added).

107. A POSITA would have been motivated to store on the server of Hagai the DMNU chunks of Li, which include the SubtitleTrack and AudioTrack chunks. For example, Hagai discloses that the server stores information such as a multimedia file and a playback-speed/file mapping table that describes available tracks and their respective playback speeds. EX1004, 17:55-60, 18:51-59. Li explains that the SubtitleTrack and AudioTrack chunks also reference MediaTracks, so a POSITA would know to relate them to their corresponding video track in Hagai. EX1005, 23:3-24:10, 24:19-31, Figs. 2.1, 2.5. Thus, in the combined configuration, the server of Hagai would store the multimedia file and playback-speed/file mapping table, and would also store the DMNU chunks describing available tracks and providing information necessary to select them. A POSITA would have understood that modifying the server to store Li's DMNU chunk, which includes the SubtitleTrack and AudioTrack chunks, is a combination of prior art elements according to known methods to achieve predictable results that would involve storing metadata chunks, in addition to the playback-speed/file mapping table and the multimedia file of Hagai, on Hagai's server. EX1004, 1:29-45, 2:19-25.

108. A POSITA would be motivated to modify Hagai such that the DMNU chunks of Li would be transmitted from the server to the playback device with the playback-speed/file mapping table before transmission of the streaming data file. A POSITA would have understood that the playback device should be provided with

the necessary information regarding the stored audio tracks, video tracks, and subtitle tracks prior to the selection of a track for playback. This track information is necessary for the playback device to select the media data to request.

109. This is a combination of prior art elements according to known methods to yield predictable results, as well as use of a known technique to improve a similar method in the same way. As noted above, because the server already stores the playback-speed/file mapping table and multimedia file of Hagai, and transmits the table to the playback device, modifying Hagai such that the server stores Li's metadata chunks and transmitting such information to the playback device would be a reasonable modification that provides expected results.

**6. [1.b(iii)] “selecting a video track from the at least one video track;”**

110. Hagai discloses limitation [1.b(iii)].

111. Hagai discloses that the playback-speed/file mapping table contains information about video tracks with different playback speeds and corresponding files. EX1004, 18:14-20, Fig. 16. Based on the information in the table, the playback device selects a video track to request from the server. *Id.*, 18:38-44. In response, the server determines the corresponding file containing the video track and transmits it to the device, including in response to byte range requests. *Id.*, 18:38-50; *infra* § IX.B.11 (Limitation [1.b(viii)]). A POSITA would have understood that Hagai discloses “selecting a video track.”

7. ***[1.b(iv)] “requesting a header describing the selected video track;”***

112. Hagai discloses limitation [1.b(iv)].

113. Hagai further discloses that the playback device requests the header describing the selected track. For example, Hagai discloses selecting a file for playback and requesting the associated stream data.

As shown in FIG. 2, in a transmission request 203 by which the data reception/playback apparatus 201 requests the transmission apparatus 202 to transmit data, a transmission range 204 indicating a transmission start position is specified separately from specifying a file to be transmitted, whereby the data reception/playback apparatus can receive the data from a desired position in the file. The specified transmission start position indicates the number of bytes from the head of the file. ...

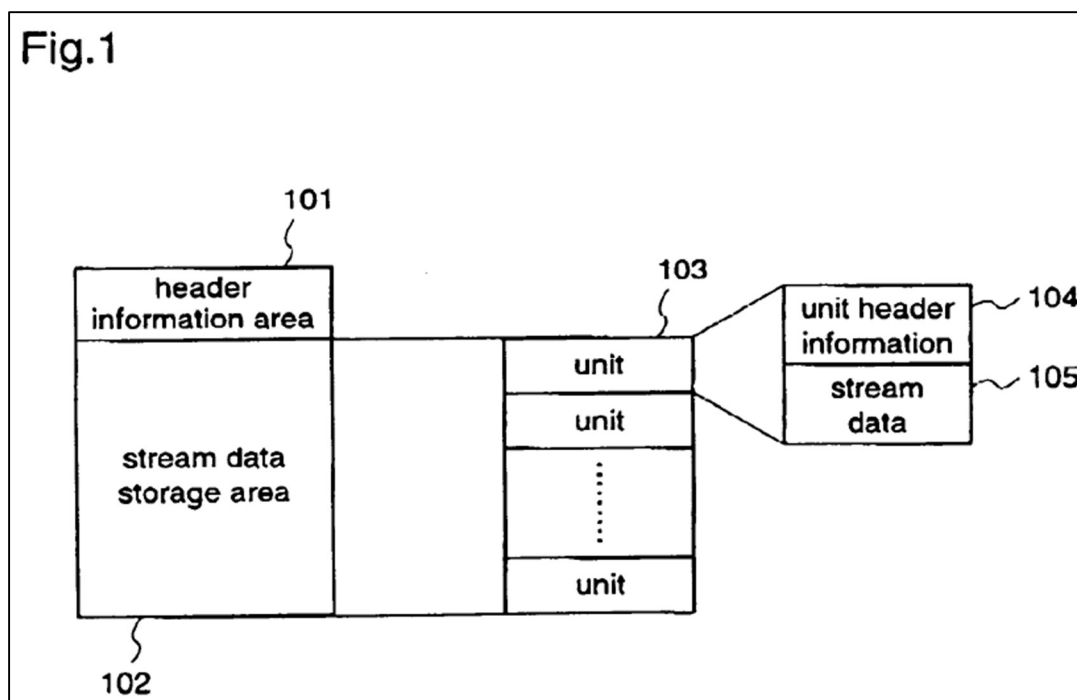
EX1004, 8:21-36; *see also id.*, 18:58-67.

114. A POSITA would have understood that when the playback device sends an initial transmission request to the server for a portion of stream data starting from the head position of the multimedia file, the data would include the header information area 101, which is the first part of the file. EX1004, 8:21-29, 14:44-48, 19:51-55, 18:58-67; *see also id.*, 9:19-23. A POSITA would have understood that the request would include the header information area 101 to enable the playback device to transmit requests for byte ranges starting from an arbitrary part of the multimedia file. *Id.*, 8:60-65, 9:63-10:4.

115. Hagai further discloses that the playback device does not require the playback start position table “when playing the data from the head of the file.”

EX1004, 14:44-48 (“However, since the data reception/playback apparatus does not need the playback start position table when playing the data from the head of the file, the data reception/playback apparatus may obtain the table only when performing jump playback.”); *see also id.*, 4:60-67, 19:51-55. Hagai also discloses that the playback start position table is stored in a file different from the stream data. *Id.*, 8:41-48, 10:32-34 (“[W]hereas in the first embodiment the playback start position table and the stream data are stored in different files.”). Thus, when playing back data from the head of the file, a POSITA would have understood that the playback device requests a header prior to receiving the start position table because it would initially receive the header information area 101 with the file and would only receive the start position table after a jump playback request.

116. Hagai further discloses that the header information area 101 describes the selected video track. For example, Figure 1 of Hagai discloses the general file format of stream data, which includes header information area 101 and a stream data storage area 102 that comprises a plurality of units 103. EX1004, 8:15-18, 10:35-41, Fig. 1.



117. A POSITA would have understood that the header information unit 101 describes the selected track because the header information area 101 provides metadata describing the type of media included in the multimedia file. EX1004, 8:15-18 (“A header information area 101 contains information such as the types of media stored in the respective units of the stream data storage area 102.”).

**8. [1.b(v)] “selecting an audio track from the multiple audio tracks;”**

118. Hagai in combination with Li discloses limitation [1.b(v)].

119. To the extent that Hagai does not disclose limitation [1.b(v)], Li does. As noted above, Li discloses that the media file contains “DMNU” chunks, including various menu chunk objects and references to media tracks. *Supra* § IX.B.5. These chunks include descriptive metadata defining how the device may select

corresponding tracks. *Supra* § IX.B.5. Li further discloses an “AudioSelectAction” chunk that is part of the DMNU chunk. EX1005, 24:19-20. This AudioSelectAction chunk contains “information that can be used to select a particular audio track,” and allows the playback device to locate and select audio tracks, which may be contained in a multimedia file or in an externally referenced file. *Id.*, 23:3-24:10, 24:19-24 (“In one embodiment, the audio track is selected from audio tracks contained within a multimedia file in accordance with an embodiment of the present invention. In other embodiments, the audio track can be located in an externally referenced file.”), Fig. 2.5; *supra* § IX.B.5 (Limitation [1.b(ii)]).

120. As discussed for limitation [1.b(ii)], a POSITA would have been motivated to store Li’s DMNU chunk on Hagai’s server and use the AudioSelectAction chunk to select an audio track. *Supra* § IX.B.5 (Limitation [1.b(ii)]). For example, in the combined configuration, the server of Hagai would store the multimedia file and playback-speed/file mapping table of Hagai, and the DMNU chunk of Li (containing the AudioSelectAction chunk). *Supra* § IX.B.5 (Limitation [1.b(ii)]). A POSITA would have understood that modifying the server to store the DMNU chunk of Li is a combination of prior art elements according to known methods to achieve predictable results and would involve storing metadata chunks, in addition to the playback-speed/file mapping table and the multimedia file of Hagai, on Hagai’s server. EX1004, 1:29-45, 2:19-25; EX1005, 23:67-24:2,

24:19-31, Fig. 2.1; *supra* § IX.B.5. Moreover, Hagai already discloses the selection of a video track, and Li provides another mechanism, via the AudioSelectAction chunk, to select an audio track. EX1005, 24:20-24; *supra* § IX.B.6 (Limitation [1.b(iii)]). Thus, modifying Hagai such that the server stores Li's DMNU chunk, including the AudioSelectAction chunk, would be a reasonable modification that provides expected results.

**9. [1.b(vi)] “obtaining index information indicating the locations of audio and video data within the selected audio and video tracks;”**

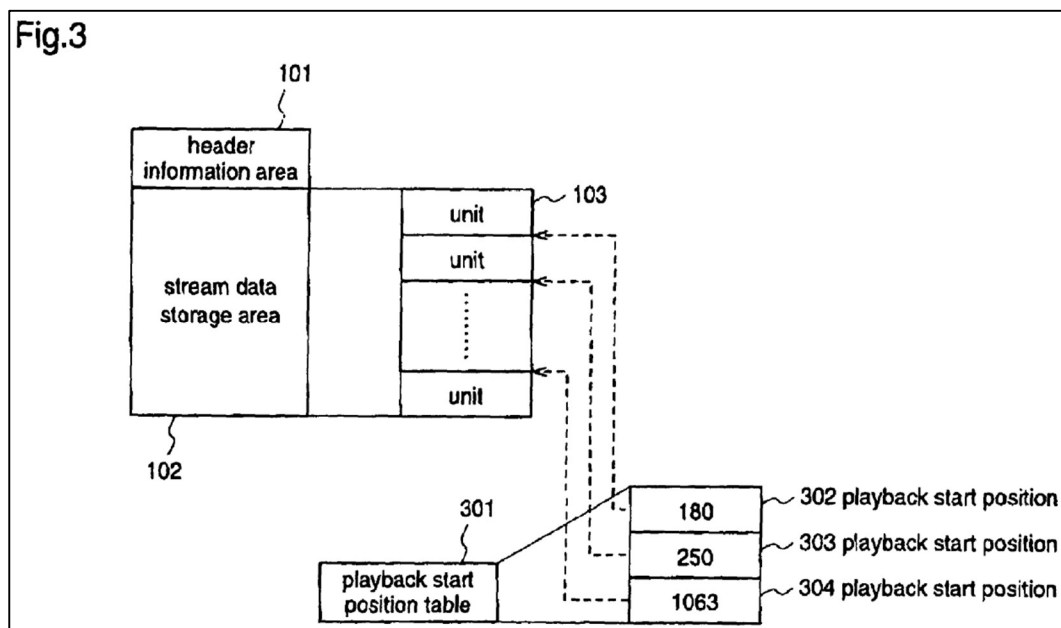
121. Hagai in combination with Li renders obvious limitation [1.b(vi)].

122. Hagai discloses obtaining index information indicating the locations of audio and video within the selected audio and video tracks. For example, as discussed *supra* in IX.B.7 (limitation [1.b(iv)]), Hagai discloses a general file format of stream data in Figure 1. EX1004, 8:11-18, Fig. 1. Hagai further discloses a “playback start position table” containing start positions enabling playback of the video and/or audio content from arbitrary points in the file:

However, when the data reception/playback apparatus receives the stream data from some midpoint of the file, the data reception/playback apparatus cannot specify the head position of the unit and therefore cannot read the stream data correctly. In order to solve this problem, in this first embodiment, the data transmission apparatus transmits a file containing a playback start position table 301 having at least one playback start position in the file as shown in FIG. 3, separately from the file containing the stream data, and the data reception/playback apparatus makes a request for data transmission from the transmission apparatus based on the playback start position table 301.

EX1004, 8:36-48.

123. As illustrated in Figure 3, the playback start position table 301 indexes playback start positions 302, 303, and 304, corresponding to “the head positions of units constituting the stream data storage areas.” EX1004, 8:55-59 (“In FIG. 3 playback start positions 302, 303, and 304 indicate, for example, the head positions of units constituting the stream data storage areas, and each playback start position indicates the number of bytes from the head position of the stream data storage area 102.”), Fig. 3.



124. The playback start position table further includes “the number of bytes from the head position of the stream data storage area 102,” enabling the playback device to “specify the unit head position even when” it receives data from an arbitrary point in the file. EX1004, 8:49-52; 8:55-59.

125. Thus, a POSITA would have understood that the playback start position table 301 is index information indicating the locations of audio and video data within the selected audio and video tracks because the playback start positions indicate “the positions in the file where data playback is possible” and allows the playback device to play the selected tracks even when playback begins from an arbitrary location in the stream, such as by selection of the location by a user in a jump playback action or otherwise. EX1004, 8:36-48 (“However, when the data reception/playback apparatus receives the stream data from some midpoint of the file, the data reception/playback apparatus cannot specify the head position of the unit and therefore cannot read the stream data correctly. In order to solve this problem, in this first embodiment, the data transmission apparatus transmits a file containing a playback start position table 301 having at least one playback start position in the file as shown in FIG. 3, separately from the file containing the stream data. . . .”), 8:55-65 (“which makes a request for data transmission based on the transmission start position table 301 containing the playback start positions (positions in the file where data playback is possible)”).

126. A POSITA would have understood this applies to both audio and video tracks. Hagai discloses that selection “enables playback of audio/video contents from an arbitrary position in stream data,” and that the stream data includes both audio and video. *See, e.g.*, EX1004, 8:4-8; *see also, id.* 1:9-12, 1:53-58, Claims 1,

7-11. A POSITA would have understood that each unit would contain both video and its corresponding audio, such as by interleaving the audio with the video within each unit. *Id.*, 8:15-28 (disclosing that the header describes the different “types of media stored in the respective units of the stream data storage area”). To the extent this is not expressly disclosed by Hagai, it is obvious in view of Li. Li discloses that its RIFF file structure stores “video” chunks and interleaved “audio” chunks within the same “RIFF” chunk. A POSITA would have understood that this file structure was well-known and would permit Hagai’s unit-based approach to achieve its disclosed result. Thus, to the extent a POSITA needed explicit disclosures or teaching of this type of file structure, a POSITA would be motivated to look to Li.

127. A POSITA would further understand that the playback device “obtains” the playback position table from the server because the playback device requests and receives the playback start position table separately from the stream data. EX1004, 9:24-29 (“Upon receipt of the transmission request 1804 from the data reception/playback apparatus 401, the transmission apparatus analyzes the transmission request 1804, and transmits to the data reception/playback apparatus the file of the playback start position table corresponding to the data identifier.”), 8:36-48 (“[T]he data transmission apparatus transmits a file containing a playback start position table 301 having at least one playback start position in the file as shown in FIG. 3, separately from the file containing the stream data. . . .”), 10:32-34

(explaining that “[T]he playback start position table and the stream data are stored in different files.”); *see also id.*, 4:60-67, 10:11-34; 19:51-59.

**10. [1.b(vii)] “determining byte ranges to request from the selected audio and video tracks using the index information;”**

128. Hagai discloses limitation [1.b(vii)].

129. For example, Hagai discloses that the playback start position selection means 407 from the playback device selects a playback start position from the playback start position table. EX1004, 9:37-46 (“Then, the playback start position selection means 407 selects a playback start position 408 from the playback start position table 406 which is stored in the playback start position table buffer 405. . . .”), Fig. 4; *supra* § IX.B.9 (Limitation [1.b(vi)]). The playback start positions indicate the head positions of units constituting the stream data storage area, which contains both audio and video, and each playback start position indicates the byte offset from the beginning of the stream data to the start position. *Id.*, 8:4-10, 8:55-65.

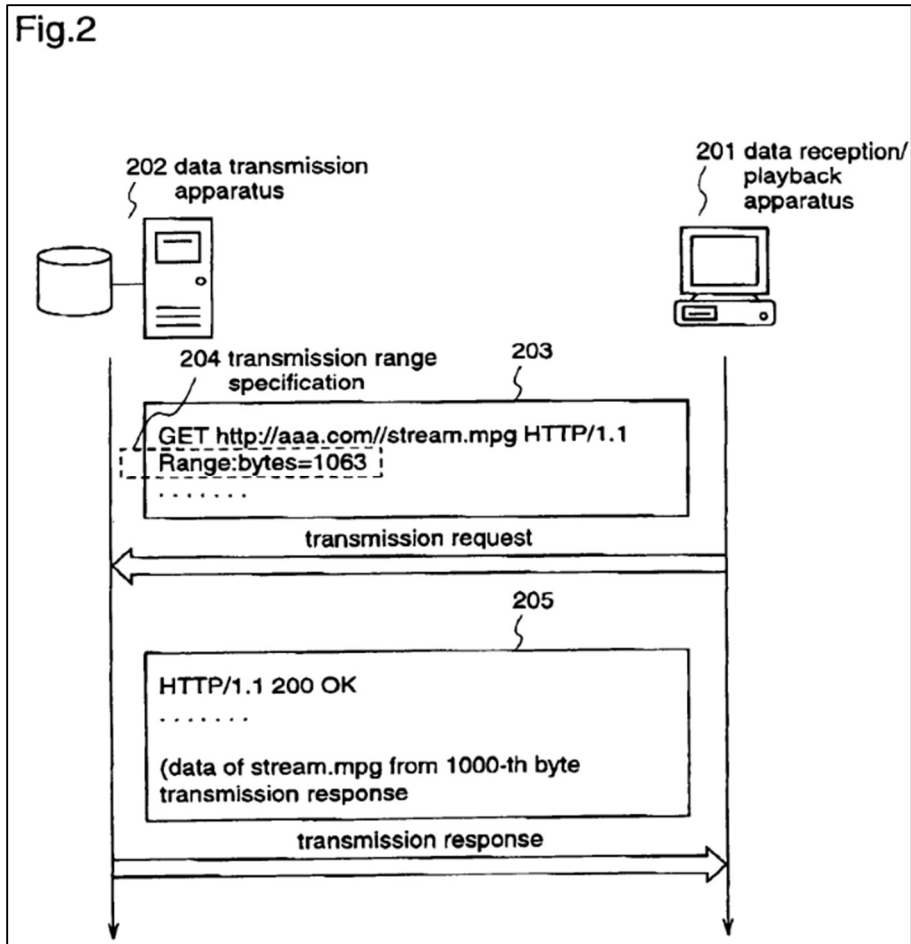
130. Based on the selected playback start position, the playback device determines a byte range to request that starts with the byte identified in the start position table and terminates at, for example, the next playback start position. EX1004, 8:60-65, 9:63-10:4; *infra* § IX.B.11 (Limitation [1.b(viii)]). A POSITA would have understood that by determining a subset of the file to request using a selected position in the start position table, Hagai’s system is “determining byte

ranges to request from the selected audio and video tracks using the index information.”

**11. [1.b(viii)] “requesting byte ranges from the selected video track and the selected audio track from the remote server system;”**

131. Hagai in combination with Li renders obvious limitation [1.b(viii)].

132. Hagai discloses that after selecting the playback start position, the transmission request sending means 1803 of the playback device sends an HTTP transmission request 1804 to the server for a specified transmission range 204. EX1004, 9:63-10:4. The transmission range 204 identifies a transmission start position that “indicates the number of bytes from the head of the file” to the selected start position. *Id.*, 8:27-29, 8:55-59. HTTP requests were well-known in the art, and a POSITA would have understood that it was conventional to request byte ranges including both a start and end point—especially to request a subset of data from large files such as those containing videos. *Supra* §§ VII.A.1.a, VII.A.1.c. Thus, a POSITA would have understood that the “transmission range 204” identifies the start of the byte range, which is requested by the playback device using HTTP. EX1004, 8:19-29, 8:55-65, 9:41-62, Fig. 2.



133. A POSITA would have understood that the transmission request is a request for the byte range of the “selected video track and the selected audio track.” The request represents the byte range between the selected transmission start position and a terminus, like the next playback start position in the stream data storage area, which contains both video and audio from the selected tracks. EX1004, 8:4-10, 8:27-29, 8:55-59, 9:63-10:21; *supra* § IX.B.10 (Limitation [1.b(vii)]). To the extent Hagai does not expressly disclose this limitation, it is rendered obvious in view of Li which describes how its RIFF file structure stores video chunks and interleaved audio chunks within the same RIFF chunk. EX1005, 1:54-65. As noted

above, a POSITA would have been motivated to combine this teaching with the unit-based approach of Hagai to ensure that its requests based on index information request byte ranges of the audio and video tracks. *Supra* § IX.B.9.

**12. [1.b(ix)] “buffering received bytes of information comprising audio and video data;”**

134. Hagai discloses limitation [1.b(ix)].

135. After sending the byte range request to the server, the playback device receives the requested data and stores it in a buffer. Specifically, Hagai discloses that the reception means 1806 of the playback device “receives a transmission response 1805 in response to the transmission request 1804, and stores the stream data of the transmission response 1805 in the reception buffer 1808.” EX1004, 10:4-7, Fig. 2. Hagai further discloses that the requested stream data includes both audio and video data. EX1004, 2:56-58 (“As described above, the reception/playback apparatus 1701 plays the audio or video that is transmitted from the transmission apparatus 1702 through the network.”), 5:7-10. Thus, a POSITA would have understood Hagai to disclose receiving and buffering bytes of audio and video data in preparation for playback.

**13. [1.b(x)] “checking that sufficient data is buffered to commence playback and playing back the buffered audio and video data;”**

136. Hagai discloses limitation [1.b(x)].

137. Hagai describes that the playback device monitors whether the amount of buffered data is sufficient before commencing or resuming playback. The playback device includes “playback means for playing the stream data stored in the reception buffer” and “reception buffer monitor means for monitoring the quantity of data stored in the reception buffer.” EX1004, 6:10-14, 16:48-51. As explained above, although Hagai discloses five embodiments that address different aspects of a method and system for receiving and playing audio and video stream data through a network, a POSITA would have understood that these embodiments are compatible modules that can be combined (e.g., the first and third embodiments). *Supra* § VIII.A.

138. When the buffer monitor means detects that stream data is delayed, leaving the buffer level below a threshold, the playback means suspends playback. EX1004, 6:15-21. The playback means then resumes playback when the buffer monitor detects that the quantity of delayed stream data exceeds a predetermined value. EX1004, 6:21-27.

In this apparatus, when the reception buffer monitor means detects that reception of stream data is delayed for a playback time, the playback means halts playback of all the stream data, and the transmission request sending means requests the transmission apparatus to suspend

data transmission in connections including stream data other than the stream data which is delayed in reception. When the reception buffer monitor means detects that the quantity of the stream data which is delayed in reception exceeds a predetermined value, the playback means restarts playback of all the stream data, and the transmission request sending means requests the transmission apparatus to restart data transmission in the connection which is suspended.

EX1004, 6:15-27; *see also id.*, 16:61-67, 17:34-41. Thus, a POSITA would have understood that the playback device can check for sufficient buffered data before commencing playback. *Supra* § VII.A.1.d.

**14. [1.b(xi)] “responding to a received seek instruction by:”**

139. Hagai discloses limitation [1.b(xi)].

140. Hagai describes a “jump playback function,” which allows playback to begin “from an arbitrary position in stream data.” EX1004, 8:4-10. For example, the playback device can respond by jumping to a playback time playback time (e.g. 3 seconds) indicated by the user. EX1004, 9:47-54. A POSITA would have understood the “jump playback function” as corresponding to a seek instruction because it is a user-controlled function that replicates the interactivity of physical media by allowing the user to play content from arbitrary positions. *Supra* VII.A.1.d. Hagai discloses responding to the instruction as explained below.

**15. [1.b(xi)(1)] “pausing playback;”**

141. Hagai discloses limitation [1.b(xi)(1)].

142. As discussed above, Hagai discloses that data may be played from an arbitrary point in response to a user request (i.e. a seek instruction). EX1004, 9:47-54; *supra* § IX.B.14 (Limitation [1.b(xi)]). Hagai explains that when this is the case, the playback device temporarily halts data reception and playback to determine the new transmission start position. EX1004, 11:20-25 (“On the other hand, when the data is to be played from some midpoint in the file, data reception is halted temporarily, and data playback is performed from the playback start position 408 which is selected from the playback start position table by the playback start position selection means 407.”). Hagai explains that “data reception is halted temporarily” during this process, and playback resumes from the selected playback start position once the new start point is determined. EX1004, 11:20-25, 13:47-56.

16. *[1.b(xi)(2)] “determining byte ranges to request from the selected audio and video tracks based upon a new playback location using the index information;”*
17. *[1.b(xi)(3)] “requesting byte ranges required to play the selected audio and video tracks from the new playback location from the remote server;”*
18. *[1.b(xi)(4)] “buffering received bytes of information comprising audio and video data pending resumption of playback; and”*
19. *[1. b(xi)(5)] “checking that sufficient data is buffered to commence playback and playing back the buffered audio and video data.”*

143. Upon receiving a seek instruction, Hagai’s playback device repeats the same sequence of operations used for playback prior to the seek instruction. EX1004, 9:17-10:21. These operations recited in [1.b(xi)(2)]-[1.b(xi)(5)] are identical to limitations [1.b(vii)]-[1.b(x)] above, except that they are performed in relation to the new playback location based on the seek instruction received in limitation [1.b(xi)]. Accordingly, these elements are obvious for the same reasons set forth with respect to [1.b(vii)]-[1.b(x)].

**C. Claim 3: “The playback device of claim 1, wherein the step of requesting a header describing the selected video track comprises the step of requesting a DRM header.”**

144. Hagai combined with Li renders obvious claim 3.

145. A POSITA would have understood that when the playback device of Hagai sends a transmission request to the server for a header describing the selected video track, the request would also include a request for the DRM header. *Supra*

§ IX.B.7 (Limitation [1.b(iv)]). For example, it was well-known and conventional for server systems storing multimedia content to encrypt the data to prevent unauthorized playback. *Supra* § VII.A.1.e. A POSITA would have further understood that when encrypting data of the multimedia file, the encrypting device generates a key to decrypt data, which is stored in the DRM header.

Where partial encryption is used, the ‘DRM’ chunks contain a reference to the portion of a ‘video’ chunk that is encrypted and a reference to the key that can be used to decrypt the encrypted portion. The decryption keys can be located in a ‘DRM’ header, which is part of the ‘strd’ chunk (see description above). The decryption keys are scrambled and encrypted with a master key. The ‘DRM’ header also contains information identifying the master key.

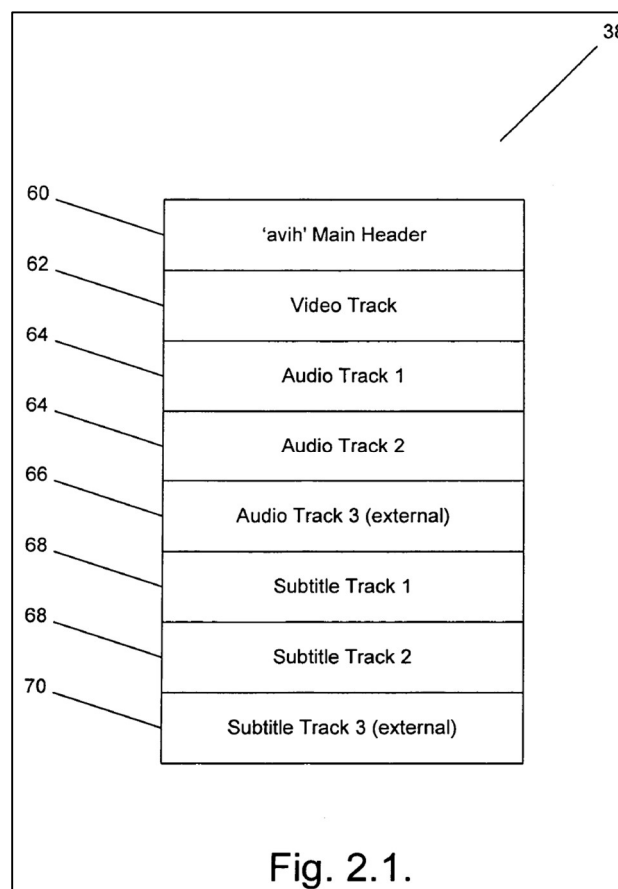
EX1005, 27:64-28:15.

A process for locating the required multimedia information from a multimedia file including DRM and displaying the multimedia information in accordance with an embodiment of the present invention is illustrated in FIG. 4.0. The process 620 includes obtaining the encryption key required to decrypt the DRM header (622). The encryption key is then used to decrypt (624) the DRM header and the first DRM chunk is located (626) within the ‘movi’ list chunk of the multimedia file. The encryption key required to decrypt the ‘DRM’ chunk is obtained (628) from the table in the ‘DRM’ header and the encryption key is used to decrypt an encrypted video chunk. The required audio chunk and any required subtitle chunk accompany the video chunk are then decoded (630) and the audio, video and any subtitle information are presented (632) via the display and the sound system.

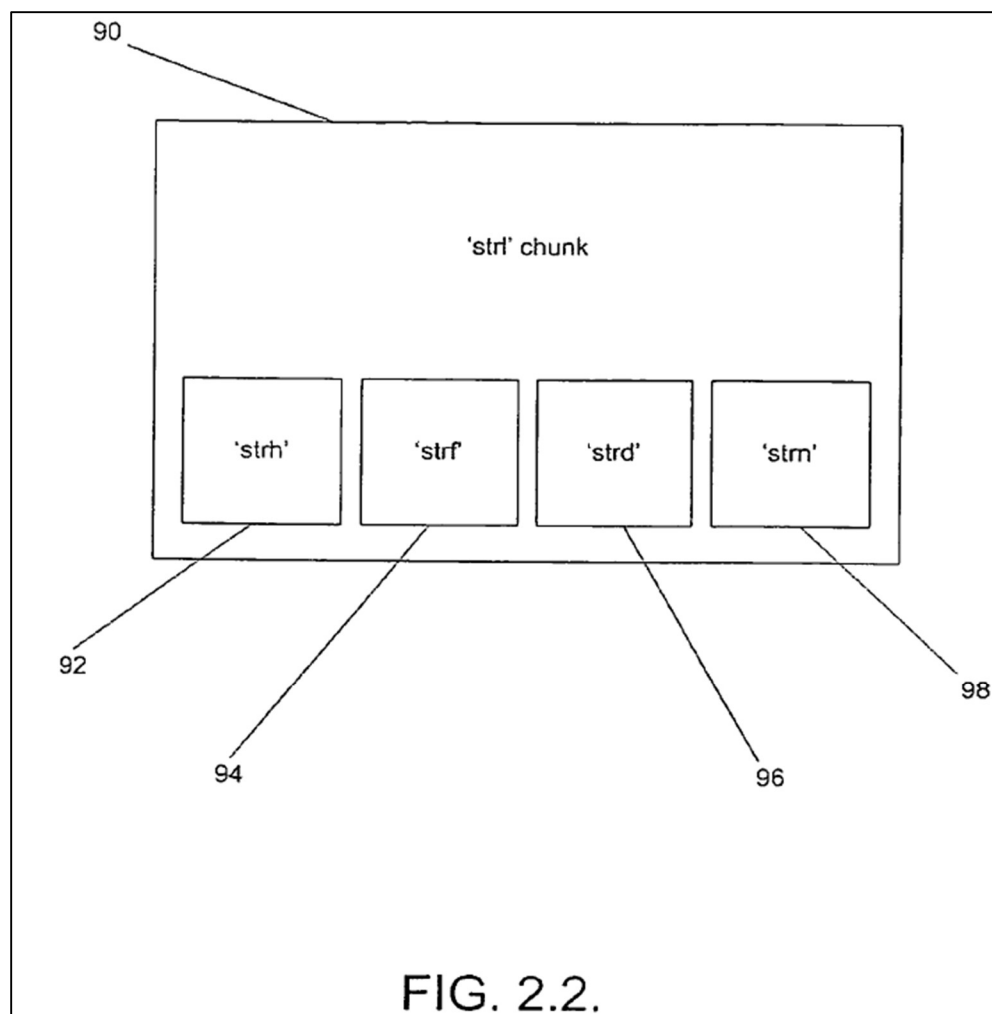
EX1005, 52:41-55; *see also supra* § VII.A.1.e.

146. Li discloses that the multimedia file includes a “hdlr list chunk” that identifies “each video track, each audio track and each subtitle track” of the

multimedia file. EX1005, 13:17-25 (describing types of chunks, such as a hdlr chunk, of the multimedia file 30), 14:13-18 (“The ‘hdlr’ list chunk 38 is defined in the AVI file format and provides information concerning the format of the data in the multimedia file. Inclusion of a ‘hdlr’ list chunk or a chunk containing similar description information is generally required. The ‘hdlr’ list chunk includes a chunk for each video track, each audio track and each subtitle track.”), Fig. 2.1.



147. The hdlr list chunk 38 further includes a “stream descriptor list for each audio, video and subtitle track” using the “strl” chunk. EX1005, 14:29-31. Each strl chunk “serves to describe each track in the multimedia file.” EX1005, 14:35-36.



148. Li discloses that the multimedia file includes a “strd” chunk 96, as illustrated in Fig. 2.2, which indicates whether the content is protected by digital rights management (“DRM”) and includes a DRM header. EX1005, 14:40-44 (“Of particular interest is the ‘strh’ chunk 92, which specifies the type of media track, and the ‘strd’ chunk 96, which can be modified to indicate whether the video is protected by digital rights management.”), 27:64-28:2 (“Where partial encryption is used, the ‘DRM’ chunks contain a reference to the portion of a ‘video’ chunk that is encrypted and a reference to the key that can be used to decrypt the encrypted portion. The

decryption keys can be located in a ‘DRM’ header, which is part of the ‘strd’ chunk (see description above). The decryption keys are scrambled and encrypted with a master key. The ‘DRM’ header also contains information identifying the master key.”), Fig. 2.2.

149. The DRM header includes encryption keys to decrypt chunks of multimedia content stored in the multimedia file. EX1005, 27:64-28:2, 52:44-55. Because the encryption keys in the DRM header are necessary to decrypt the multimedia content, the DRM header must be requested and received prior to decoding and playback. EX1005, 52:49-55 (“The encryption key required to decrypt the ‘DRM’ chunk is obtained (628) from the table in the ‘DRM’ header and the encryption key is used to decrypt an encrypted video chunk. The required audio chunk and any required subtitle chunk accompany the video chunk are then decoded (630) and the audio, video and any subtitle information are presented (632) via the display and the sound system.”), Fig. 4. Thus, a POSITA would have understood that when the playback device requests a header from the server, the playback device would also request the DRM header of Li to access the keys required to decrypt the encrypted content of the multimedia file for playback on the playback device. EX1005, 27:64-28:4, 52:41-55; *supra* § IX.B.7 (Limitation [1.b(iv)]).

150. A POSITA would have further found it obvious to store the DRM header on the server of Hagai. For example, as explained in limitation [1.b(ii)],

Hagai discloses that the server stores the multimedia file and a playback-speed/file mapping table. *Supra* § IX.B.5 (Limitation [1.b(ii)]). Thus, in the combined configuration, the server of Hagai would store the multimedia file and playback-speed/file mapping table and would also store the DRM header corresponding to the tracks stored on the server. A POSITA would have understood that modifying the server to store the DRM header of Li is a combination of prior art elements according to known methods to achieve predictable results and would involve saving data chunks, in addition to the playback-speed/file mapping table and the multimedia file of Hagai, on Hagai's server. EX1004, 1:29-45, 2:19-25.

151. A POSITA would be motivated to modify Hagai such that the DRM header of Li is transmitted with Hagai's header information 101. A POSITA would have understood that requesting and receiving Li's DRM header alongside Hagai's header information area 101, as explained in limitation [1.b(iv)], would enable the playback device to decrypt received data without the delay caused by additional requests. EX1004, 8:15-18 ("A header information area 101 contains information such as the types of media stored in the respective units of the stream data storage area 102."); *supra* § IX.B.7 (Limitation [1.b(iv)]). Indeed, modifying the playback device of Hagai such that the request for the header includes a request for the DRM header is a combination of known methods to achieve predictable results because Hagai already discloses requesting header information 101 stored on the server, and

the combined transmission request from the playback device would merely request additional information stored on the server. EX1004, 1:29-45, 2:19-25; *supra* § IX.B.7 (Limitation [1.b(iv)]).

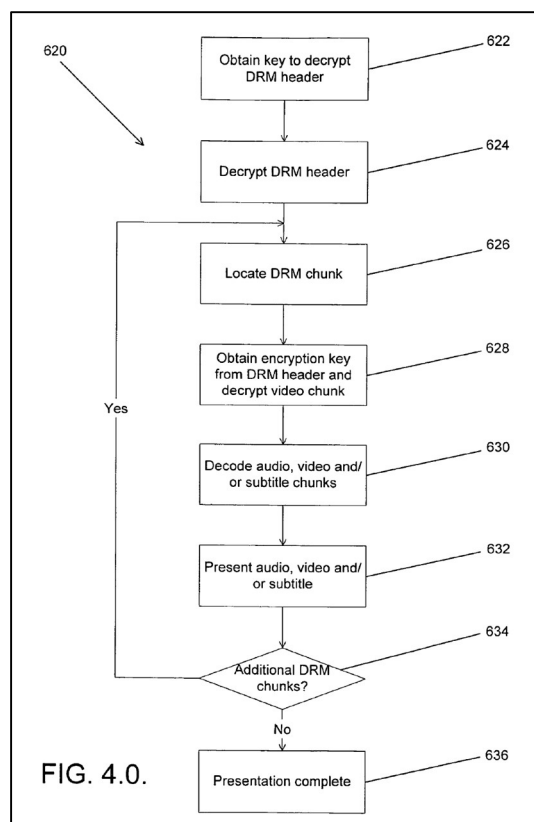
**D. Claim 4: “The playback device of claim 3, wherein the application is further capable of causing the processor to perform the step of decrypting the DRM header.”**

152. Hagai combined with Li renders obvious claim 4.

153. For example, Li discloses in Figure 4 a process for “locating the required multimedia information from a multimedia file.” EX1005, 52:41-44. The process 620 includes first obtaining a key to decrypt the DRM header in step 622 and subsequently decrypting the DRM header in step 624 using the encryption key.

The process 620 includes obtaining the encryption key required to decrypt the DRM header (622). The encryption key is then used to decrypt (624) the DRM header and the first DRM chunk is located (626) within the ‘movi’ list chunk of the multimedia file. The encryption key required to decrypt the ‘DRM’ chunk is obtained (628) from the table in the ‘DRM’ header and the encryption key is used to decrypt an encrypted video chunk. The required audio chunk and any required subtitle chunk accompany the video chunk are then decoded (630) and the audio, video and any subtitle information are presented (632) via the display and the sound system.

EX1005, 52:44-55.



EX1005, Fig. 4.

154. A POSITA would have understood that in the combined configuration, once the playback device requests and receives the DRM header, the playback device would obtain the encryption key and decrypt the DRM header. EX1005, 52:41-55; *supra* § IX.C (Claim 3); *supra* § VII.A.1.e. A POSITA would have understood that decryption techniques were conventional and well-known, and that the processor of the playback device could perform these decryption techniques using known methods. *Supra* §§ VII.A.1.a, VII.A.1.e.

155. As discussed in Section IX.C, a POSITA would have been motivated to combine Li's decryption functionality with Hagai's multimedia playback system

to enable playback of protected content using standard DRM techniques. A POSITA would have been further motivated to modify Hagai's multimedia playback system to include Li's DRM header decryption because the DRM header would have to be encrypted to secure the decryption keys within. EX1005, 27:67-28:4; *infra*, IX.E.1. Incorporating Li's DRM header processing into Hagai would have yielded predictable results, as it merely adds known decryption functionality without altering Hagai's core operation. *Supra* § IX.C (Claim 3).

**E. Claim 5**

**1. [5.pre], [5.a] “The playback device of claim 3, wherein: the DRM header identifies at least one key; and”**

156. Hagai combined with Li discloses limitations [5.pre] and [5.a].

157. For example, Li discloses that the DRM header contains a table of decryption keys (EX1005, 52:49-51), which are encrypted using a master key. EX1005, 28:2-3. The DRM header “contains information identifying the master key.” *Id.* 28:3-4. A POSITA would have been motivated to store Li's DRM header on the server of Hagai for the reasons discussed above. *Supra* §§ IX.C, IX.D (Claims 3, 4).

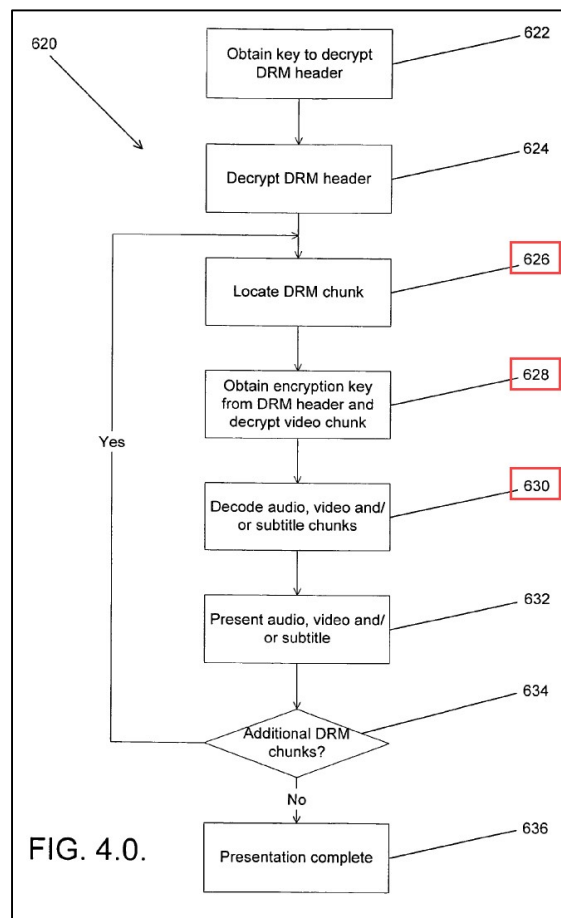
**2. [5.b] “the application is further capable of causing the processor to perform the steps of decrypting encrypted frames of video using the at least one key prior to decoding.”**

158. Hagai combined with Li discloses limitation [5.b].

159. As discussed above, the DRM header identifies the master key. *Supra* § IX.E.1. Li explains that the master key is used to scramble and encrypt the decryption keys. EX1005, 27:67-28:4. A POSITA would have understood that the same master key is used to decrypt the decryption keys, which are then used to decrypt the frames of video. *Id.*

160. For example, as discussed in claim 4, Li discloses in process 620 for locating multimedia information from a multimedia file as illustrated in Figure 4 includes the steps of first obtaining the encryption key 622 and using the encryption key to decrypt the DRM header 624. EX1005, 52:44-49 (“The process 620 includes obtaining the encryption key required to decrypt the DRM header (622). The encryption key is then used to decrypt (624) the DRM header and the first DRM chunk is located (626) within the ‘movi’ list chunk of the multimedia file.”). The process further includes locating the DRM chunk of the multimedia file 626, obtaining an encryption key (i.e., an encryption key decrypted using the master key) from the DRM header to decrypt an encrypted video chunk in step 628, and then decoding the video chunk in step 630. EX1005, 52:49-55 (“The encryption key required to decrypt the ‘DRM’ chunk is obtained (628) from the table in the ‘DRM’ header and the encryption key is used to decrypt an encrypted video chunk. The required audio chunk and any required subtitle chunk accompany the video chunk are then decoded (630) and the audio, video and any subtitle information are

presented (632) via the display and the sound system.”), 51:15-20 (“In embodiments where DRM is used to protect the video content of a multimedia file, the DRM information can be generated concurrently with the encoding of the video chunks. As each chunk is generated, the chunk can be encrypted and a DRM chunk generated containing information concerning the encryption of the video chunk.”), Fig. 4 (annotated).



161. A POSITA would have understood that the master key used to decrypt the encryption key obtained from the DRM header at step 628 is a key that used by the processor to perform the process of decrypting encrypted frames of video prior

to decoding. The application uses the master key to enable each of the keys to decrypt the individual video chunks in step 628.

162. A POSITA would have been motivated to combine Li with Hagai because it would be understood that the playback device must first decrypt the encoded video frames using a key prior to decoding; the playback device cannot decode and play back encrypted video data, and Li teaches a secure way of providing for that functionality. *Supra* § VII.A.1.e. Further, a POSITA would have had a reasonable expectation of success, and this combination would have yielded predictable results because it is no more than a combination of using conventional, well-known decryption techniques in a conventional way in video transmission and playback system.

**F. Claim 6**

**1. [6.pre], [6.a] “The playback device of claim 5, wherein: encrypted frames of video are only partially encrypted;”**

163. Hagai combined with Li discloses limitations [6.pre] and [6.a].

164. Li discloses that the multimedia file includes a “series of encoded video frames” that “are encoded as a series of video chunks.” EX1005, 4:42-45, Abstract. Li further discloses that the video chunks of the multimedia file could be partially encrypted. EX1005, 27:62-64 (“In an encryption system in accordance with an embodiment of the present invention, the video chunks are only partially encrypted.”). Because video frames may be encoded as video chunks, a POSITA

would have understood that partially encrypted video chunks may include partially encrypted frames. EX1005, 4:42-45 (“the series of encoded video frames are encoded as a series of video chunks”), 3:17-23 (“In another embodiment again, the video information is stored as chunks within the multimedia file. In addition, each chunk of video information can include a single frame of video. Furthermore, the audio information can be stored as chunks within the multimedia file and audio information from two separate audio tracks is not contained within a single chunk of audio information.”), 3:60-63 (“In addition, the processor can be configured to encode the video track such that a single frame of video is contained within each ‘video’ chunk.”).

165. Li further explains that DRM chunks of the multimedia file provide DRM information for the video chunk and include information “point[ing] to the start of the encrypted block within the frame.” EX1005, 28:10-13 (“The ‘status’ value can be used to indicate whether the frame is encrypted, the ‘offset’ value 284 points to the start of the encrypted block within the frame and the ‘number’ value 286 indicates the number of encrypted bytes in the block.”), 27:53-56; *infra*, § IX.F.2 (Limitation [6.b]). Furthermore, it was well-known and conventional to a POSITA to encrypt only a portion of a video frame as an optimization technique to reduce processing overhead while maintaining sufficient protection of digital content. *Supra* § VII.A.1.e. Thus, a POSITA would have understood that Li discloses a

system where only specific portions of the video frames—such as selected blocks within a frame—are encrypted, while other portions remain unencrypted.

166. As discussed with respect to claims 3-5, a POSITA would have been motivated to incorporate Li's partial encryption scheme into Hagai's multimedia playback system. The combination would have yielded predictable results, enabling secure playback of protected content using well-known partial encryption techniques without requiring changes to Hagai's existing byte-range-based retrieval and playback architecture.

167. Partial encryption was well-known to a POSITA before January 2007. *Supra* § VII.A.1.e. The specification of the 141 patent does not disclose any novel partial encryption or decryption technique but instead incorporates prior art by reference—including Li, which provides a detailed discussion of partial encryption.

**2. [6.b] “the received bytes of information contain references to portions of video frames that are encrypted; and”**

168. Hagai combined with Li discloses limitation [6.b].

169. A POSITA would have understood that the partially encrypted video chunks and frames of Li, as explained in limitation [6.a], include references to portions of video frames that are encrypted. Specifically, Li discloses that DRM chunks contain information needed to decrypt and decode media content.

The ‘DRM’ chunks 270 are ‘data’ chunks that contain digital rights management information, which can be identified by a FOURCC code ‘nndd’. The first two characters ‘nn’ refer to the track number and the

second two characters are ‘dd’ to signify that the chunk contains digital rights management information. In one embodiment, the ‘DRM’ chunk 270 provides the digital rights management information for the ‘video’ chunk 262’ following the ‘DRM’ chunk. A device attempting to play the digital rights management protected video track uses the information in the ‘DRM’ chunk to decode the video information in the ‘video’ chunk.

EX1005, 27:48-59.

170. The DRM chunk includes an offset value 284, which specifies “the start of the encrypted block within the frame,” and a number value 286, which “indicates the number of encrypted bytes in the block.” EX1005, 28:10-15 (“The ‘status’ value can be used to indicate whether the frame is encrypted, the ‘offset’ value 284 points to the start of the encrypted block within the frame and the ‘number’ value 286 indicates the number of encrypted bytes in the block. The ‘key’ value 288 references the decryption key that can be used to decrypt the block.”), Fig. 2.9.

171. Thus, a POSITA would have understood that the received DRM chunk includes offsets and byte ranges within frames—explicitly identifying which regions of the video frames are encrypted, enabling the playback device to locate and decrypt the encrypted portions of partially-encrypted frames.

3. **[6.c] the step of decrypting encrypted frames of video using the at least one key prior to decoding further comprises decrypting encrypted portions of frames of video using the references to the portions of the video frames that are encrypted.”**

172. Hagai combined with Li discloses limitation [6.c].

173. As discussed in limitation [5.b], Li discloses that the DRM chunk includes a key value 288, which “references the decryption key that can be used to decrypt” the encrypted portions of the frame. EX1005, 28:10-15 (“The ‘status’ value can be used to indicate whether the frame is encrypted, the ‘offset’ value 284 points to the start of the encrypted block within the frame and the ‘number’ value 286 indicates the number of encrypted bytes in the block. The ‘key’ value 288 references the decryption key that can be used to decrypt the block.”), Fig. 2.9; *supra* § IX.E.2 (Limitation [5.b]). A POSITA would have understood that the key value 288 identifies the key needed to decrypt the encrypted portions of the frame, which are located using the information in the DRM chunk. EX1005, 28:10-15, Fig. 2.9 *supra* § IX.F.2 (Limitation [6.b]). Thus, decrypting the encrypted portions of frames includes “using the references to the portions of the video frames that are encrypted.” EX1005, 28:10-15, Fig. 2.9.

174. As discussed with respect to claims 3-5, a POSITA would have been motivated to incorporate Li’s partial encryption scheme into Hagai’s multimedia playback system. The combination would have yielded predictable results, enabling secure playback of protected content using well-known partial encryption techniques without requiring changes to Hagai’s existing byte-range-based retrieval and playback architecture.

175. Partial encryption was well known to a POSITA prior to January 2007. *Supra* § VII.A.1.e. The specification of the 141 patent does not disclose any novel partial encryption or decryption technique but instead incorporates prior art by reference—including Li, which provides a detailed discussion of partial encryption.

**G. Claim 7: “The playback device of claim 1, wherein playing back the buffered video data from the requested byte ranges provides fast forward trick play.”**

176. Hagai discloses claim 7.

177. For example, the playback start position table enables the playback device to request, receive, and play back data from arbitrary points in the stream. EX1004, 9:63-10:4, 10:17-21; *supra* § IX.B.11 (Limitation [1.b(viii)]). Hagai further discloses that the playback start position table can include start positions corresponding to specific times (e.g., second 1, second 2, or second 3), which may be specified by the user. EX1004, 9:47-54. Based on the desired time, the playback device requests and receives data from the server. *Id.*, 9:63-10:7. The playback device stores the data in a buffer as it is received, then plays back the buffered data. Hagai, 10:4-10; *supra* § IX.B.12 (Limitation [1.b(ix)]).

178. A POSITA would have understood that, because Hagai enables playback from arbitrary points (for example, valid entry points to the stream that are selected by the user), the playback device could perform a series of jump playback operations forward from the present playback location and play back only buffered

key frames or intra frames. EX1004, 8:4-10. A POSITA would further understand that a series of jump playback operations displaying only these buffered key frames or intra frames (i.e., the I-frames at the start of each segment) to be a fast-forward operation because what is displayed to the user mimics the fast-forward operation in physical media, which was a known operation. *Id.*, 8:52-65, 9:50-62, 14:29-32; *supra* §§ VII.A.1.b, VII.A.1.d.

**H. Claim 8: “The playback device of claim 1, wherein playing back the buffered video data from the requested byte ranges provides fast rewind trick play.”**

179. Hagai discloses claim 8.

180. A POSITA would have understood that because Hagai enables playback from arbitrary points the playback device could play back buffered key frames or intra frames to perform a series of jump playback operations to an earlier playback location. EX1004, 8:4-10; *see supra* § IX.G. A POSITA would have understood a series of jump playback operations displaying only these buffered key frames or intra frames (i.e., the buffered I-frames) to be a fast-rewind operation because what is displayed to the user mimics the fast-rewind operation in physical media. EX1004, 8:52-65, 9:50-62, 14:29-32; *supra* §§ VII.A.1.b, VII.A.1.d.

## I. Claim 10

1. [10.pre], [10.a]: “The playback device of claim 1, wherein the application is further capable of causing the processor to perform the steps of: selecting a subtitle track from the multiple subtitle tracks;”

181. Hagai combined with Li discloses limitations [10.pre] and [10.a].

182. As discussed above in limitations [1.b(ii)] and [1.b(v)], a POSITA would have understood the server of Hagai would store the playback-speed/file mapping table and multimedia file of Hagai, as well as the DMNU chunks of Li. EX1005, 23:3-24:10, Fig. 2.5; *supra* §§ IX.B.5 (Limitation [1.b(ii)]), IX.B.8 (Limitations [1.b(v)]). A POSITA would have understood that Hagai’s multimedia file could store Li’s subtitle tracks that would be synchronized with the audio and video tracks during playback.

183. Li discloses that the SubtitleTrack chunk is used to identify multiple available subtitle tracks.

A ‘PlayAction’ chunk 182 identifies a portion of each of the video, audio and subtitle tracks within a multimedia file. The ‘PlayAction’ chunk references a portion of the video track using a reference to a ‘MediaTrack’ chunk (see discussion below). The ‘PlayAction’ chunk identifies audio and subtitle tracks using ‘SubtitleTrack’ 192 and ‘AudioTrack’ 194 chunks.

EX1005, 23:63-24:2; *supra* § IX.B.5 (Limitation [1.b(ii)]).

184. Li further discloses that a subtitle track of the multimedia file can be chosen for display, and that the SubtitleSelectAction chunk contains the information needed to select a subtitle track.

Each ‘SubtitleSelectAction’ chunk 190 contains information that can be used to select a particular subtitle track. In one embodiment, the subtitle track is selected from a subtitle contained within a multimedia file in accordance with an embodiment of the present invention. In other embodiments, the subtitle track can be located in an externally referenced file.

EX1005, 24:25-31.

When a subtitle track is chosen to be displayed, a determination can be made as to whether the previous video frame included a subtitle (this determination may be made in any of a variety of ways that achieves the outcome of identifying a previous ‘subtitle’ chunk that contained subtitle information that should be displayed over the currently decoded video frame).

EX1005, 52:58-64.

185. A POSITA would have understood that when the playback device “selects a file to be requested for transmission,” the apparatus is selecting a subtitle track from the multiple subtitle tracks because it would have been undesirable to display them all at once (e.g., in several different languages). EX1004, 18:58-67 (“In this case, the data reception/playback apparatus selects a file to be requested for transmission, and conducts a transmission request for the selected file.”).

186. Moreover, a POSITA would have been motivated to store Li’s DMNU chunk, which contains the SubtitleSelectAction chunk, on Hagai’s server and use

the SubtitleSelectAction chunk to select a subtitle track. EX1005, 23:3-24:10, Fig. 2.5. In the combined configuration, Hagai's server would store Hagai's multimedia file and playback-speed/file mapping table and Li's DMNU chunk. *Supra* §§ IX.B.5, IX.B.8 (Limitations [1.b(ii)], [1.b(v)]). A POSITA would have understood that modifying the server to store the DMNU chunk of Li is a combination of prior art elements according to known methods to achieve predictable results and would involve saving data chunks, in addition to the playback-speed/file mapping table and the multimedia file of Hagai, on Hagai's server. EX1004, 1:29-45, 2:19-25; EX1005, 23:67-24:2, 24:19-31, Fig. 2.1; *supra* § IX.B.5 (Limitation [1.b(ii)]). Moreover, Hagai already discloses the selection of a video track, and Li provides another mechanism, via the SubtitleSelectAction chunk, to select a subtitle track. EX1005, 24:25-31 ("Each 'SubtitleSelectAction' chunk 190 contains information that can be used to select a particular subtitle track. In one embodiment, the subtitle track is selected from a subtitle contained within a multimedia file in accordance with an embodiment of the present invention. In other embodiments, the subtitle track can be located in an externally referenced file."). Thus, modifying Hagai such that the server stores Li's SubtitleSelectAction chunks, and then selecting and using that information as recited in claims [10.a], [10.b], and [10.c] would be a reasonable modification that provides expected results.

**2. [10.b]: “requesting data from the selected subtitle track from the remote server system, where the data contains subtitle text; and”**

187. Hagai combined with Li discloses limitation [10.b].

188. As discussed above in limitations [1.b(ii)], [1.b(v)], and [10.a], a POSITA would have understood the server of Hagai would store the playback-speed/file mapping table and multimedia file of Hagai, as well as the DMNU chunks of Li. *Supra* §§ IX.B.5, IX.B.8, IX.I.1 (Limitations [1.b(ii)], [1.b(v)], [10.a]). The DMNU chunk contains the SubtitleTrack chunk, which identifies available subtitle tracks. EX1005, 23:3-24:10, Fig. 2.5.

189. Li discloses that the stored subtitle tracks of the multimedia file are encoded as a series of subtitle chunks that “are generated from text subtitles.” EX1005, 29:16, 4:42-50. Each chunk includes information relating to the “portion of the video sequence over which the subtitle should be superimposed” and the “information capable of being represented as text on a display.” *Id.*, 4:42-50, 11:66-12:4. Li further discloses that the SubtitleTrack chunk refers to a MediaTrack chunk, which identifies the appropriate portions of video tracks and corresponding subtitle tracks. *Id.*, 25:27-30 (“In several embodiments, a ‘MediaTrack’ chunk references the appropriate portion of the video track and a number of additional ‘MediaTrack’ chunks each reference one of the audio tracks or subtitle tracks.”). Thus, a POSITA

would have understood that the server of Hagai would also store the data containing subtitle tracks.

190. A POSITA would further understand that when the playback device requests a byte range for a selected video track, the request would further include a request for the selected subtitle track because the playback device would require the necessary subtitle information to playback the subtitles corresponding to, and in conjunction with, the selected video content. *Supra* §§ IX.B.9, IX.B.11 (Limitations [1.b(vi)], [1.b(viii)]).

**3. [10.c]: “*superimposing the subtitle text over video during playback.*”**

191. Hagai combined with Li discloses limitation [10.c].

192. Li further discloses that the subtitle text is superimposed over the decoded video, which is then displayed alongside the superimposed subtitles. EX1005, 53:5-8 (“If a ‘subtitle’ chunk is located, then the subtitle is superimposed on the decoded video. The video (including any superimposed subtitles) is then displayed with the accompanying audio.”). Thus, a POSITA would have understood that when the playback device of Hagai plays the video, the playback device would utilize the requested subtitle track data from the server to superimpose the text of the subtitles over the video and doing so would have only involved implementing a conventional technique. *See supra* § IX.I.1 (Limitation [10.a]).

**J. Claim 20:**

1. ***[20.pre] A method of playing back content on a playback device, comprising:***

193. To the extent the preamble of claim 20 is limiting, Hagai renders obvious this limitation. Specifically, Hagai discloses a method for playback on a playback device of audio and video content received from a server using HTTP. *See* EX1004, 2:55-57, 2:59-67, 7:64-67, 9:4-10:10 (explaining that the playback apparatus “plays the stream data” and “outputs the played data as an output signal 1813.”), Figs. 2, 4; *supra* § IX.B.1 (Limitation [1.pre]).

2. ***[20.a] establishing at least one connection for communicating with a remote server system using a playback device;***
3. ***[20.b] obtaining information from a remote server system using the playback device, where the obtained information describes at least one video track, multiple audio tracks, and multiple subtitle tracks;***
4. ***[20.c] selecting a video track from the at least one video track;***
5. ***[20.d] requesting a header describing the at least one video track using the playback device;***
6. ***[20.e] selecting an audio track from the multiple audio tracks using the playback device;***
7. ***[20.f] obtaining index information indicating the locations of audio and video data within the selected audio and video tracks;***
8. ***[20.g] determining byte ranges to request from the selected audio and video tracks using the index information;***

9. **[20.h] requesting byte ranges from the selected video track and the selected audio track from the remote server system using the playback device;**
10. ***[20.i] buffering received bytes of information comprising audio and video data on the playback device;***
11. ***[20.j] checking that sufficient data is buffered to commence playback and playing back the buffered audio and video data using the playback device;***
12. ***[20.k] responding to receipt of a seek instruction at the playback device by:***
13. ***[20.k(i)] pausing playback on the playback device;***

194. The method steps recited in limitations [20.a] through [20.k(i)] are substantively identical to the limitations recited in limitations [1.b(i)]-[1.b(xi)(1)], respectively. For the reasons described above, which are incorporated herein by reference, Hagai combined with Li renders these limitations of claim 20 obvious. Each of claim limitations [20.b], [20.d], [20.e], [20.h], and [20.j] recites that the step is performed “using the playback device,” which is not explicitly recited in limitations [1.b(i)]-[1.b(xi)(1)]. A POSITA would have understood that the method of claim 20 would be performed on the playback device described in the analysis of claim 1 and as described above in limitation [20.pre].

14. ***[20.k(ii)] determining byte ranges to request from the selected audio and video tracks based upon a new playback location using the index information;***
15. ***[20.k(iii)] requesting byte ranges required to play the selected audio and video tracks from the new playback location from the remote server using the playback device;***

16. *[20.k(iv)] buffering received bytes of information comprising audio and video data pending resumption of playback using the playback device; and*
17. *[20.k(v)] checking that sufficient data is buffered to commence playback and playing back the buffered audio and video data using the playback device.*

195. The method steps recited in limitations [20.k(ii)] through [20.k(v)] are substantively identical to the limitations recited in limitations [1.b(xi)(2)]-[1.b(xi)(5)], respectively. For the reasons discussed above for limitations [1.b(xi)(2)]-[1.b(xi)(5)], Hagai discloses the substantively identical limitations of claim 20. *Supra* §§ IX.J.8, IX.J.9, IX.J.10, IX.J.11; *see also* §§ IX.B.16, IX.B.17, IX.B.18, IX.B.19. Each of claim limitations [20.k(iii)] through [20.k(v)] recites that the step is performed “using the playback device,” which is not explicitly recited in limitations [1.b(xi)(2)]-[1.b(xi)(5)]. A POSITA would have understood that the method of claim 20 would be performed on the playback device described in the analysis of claim 1 and as described above in limitation [20.pre].

**K. Claim 22: “The method of claim 20, wherein requesting a header describing the selected video track further comprises requesting a DRM header using the playback device.”**

196. For the reasons discussed above for claim 3, claim 22 is unpatentable. *Supra* § IX.C (Claim 3). For the reasons discussed in claim 3, a POSITA would have understood that the playback device requests the DRM header. *Supra* § IX.C (Claim 3).

**L. Claim 23: “The method of claim 22, further comprising decrypting the DRM header.”**

197. For the reasons discussed above for claim 4, claim 23 is unpatentable.

*Supra* § IX.D (Claim 4). For the reasons discussed in claim 4, a POSITA would have understood that, once it requests and receives the DRM header, the playback device would obtain the encryption key and decrypt the DRM header. EX1005, 52:41-55; *supra* § IX.D (Claim 4).

**M. Claim 24: “The method of claim 23, wherein: the DRM header identifies at least one key; and the method further comprises decrypting encrypted frames of video using the at least one key prior to decoding using the playback device.”**

198. For the reasons discussed above for claim 5, claim 24 is unpatentable.

*Supra* § IX.E. For the reasons discussed in [5.b], a POSITA would have understood that the playback device of Hagai must decrypt the encoded video frames using a key prior to decoding because the playback device cannot play back the data of the video chunk in an encrypted form. EX1005, 52:49-55; *supra* § IX.E.2 (Limitation [5.b]).

- N. Claim 25: “The method of claim 24, wherein: encrypted frames of video are only partially encrypted; the received bytes of information contain references to portions of video frames that are encrypted; and decrypting encrypted frames of video using the at least one key prior to decoding further comprises decrypting encrypted portions of frames of video using the references to the portions of the video frames that are encrypted using the playback device.**

199. For the reasons discussed above for claim 6, claim 25 is unpatentable.

*Supra* § IX.F. Li discloses that the playback device uses DRM chunks, which reference “the decryption key that can be used to decrypt” the encrypted portions of the frame, to play encrypted video tracks. EX1005, 27:56-59, 28:14-15; *supra* § IX.F.3 (Limitation [6.c]). For the reasons discussed above for limitation [6.c], a POSITA would have understood that the playback device uses the references in the DRM chunk to identify the encrypted portions of video frames. *Supra* § IX.F.3 (Limitation [6.c]).

- O. Claim 26: “The method of claim 20, wherein playing back the buffered video data from the requested byte ranges provides fast forward trick play.”**

200. For the reasons discussed above for claim 7, claim 26 is unpatentable.

*Supra* § IX.G (Claim 7).

- P. Claim 27: “The method of claim 20, wherein playing back the buffered video data from the requested byte ranges provides fast rewind trick play.”**

201. For the reasons discussed above for claim 8, claim 27 is unpatentable.

*Supra* § IX.H (Claim 8).

**Q. Claim 29: “The method of claim 20, further comprising: selecting a subtitle track from the multiple subtitle tracks using the playback device; requesting data from the selected subtitle track from the remote server system using the playback device, where the data contains subtitle text; and superimposing the subtitle text over video during playback using the playback device.”**

202. For the reasons discussed above for claim 10, claim 29 is unpatentable.

*Supra* § IX.I (Claim 10).

**X. GROUND 2: Claims 9, 12-14, 16-18 and 28 are obvious over Hagai, Li Park, and General Knowledge of a POSITA**

**A. Motivation to Combine**

203. A POSITA would have been motivated to combine Hagai and Li for the reasons discussed in Ground 1. *Supra* § IX.A. A POSITA would have been further motivated to combine Hagai and Li with the buffer management technique taught by Park for the reasons set forth below.

204. Hagai describes a client-server streaming architecture that enables seeking to arbitrary positions in a media stream using a playback start position table to determine HTTP byte range requests. EX1004, 8:1-10; Claim 1. Hagai also teaches that the client buffers received stream data, controls playback based on data levels stored in the buffer levels, and monitors levels of buffered data “to avoid degradation in quality of played data due to interruptions of data.” EX1004, 6:10-14, 6:21-31, 16:48-51; *supra* § IX.B.13. Li complements Hagai by disclosing the use of metadata—such as stream descriptors and DRM headers—to coordinate playback

and manage access. *Supra* §§ VIII.B, IX.C (Claim 3); EX1005, Abstract, 1:18-22, 12:9-36, 27:42-28:15, 51:60-64, Fig. 1.

205. While neither Hagai nor Li explicitly discloses discarding previously buffered audio and video data in response to a received trick play instruction, like a seek command, Park does. Specifically, Park discloses a “method of playout buffer management” in which the client flushes the playout buffer when a playback command, such as fast-forward, fast-reverse, or a change in AV program, triggers a transition to a new stream segment. EX1006, [0012], [0026], Claim 7. Park explains that “[f]lushing the playout buffer of residual, e.g., unwanted, AV program material eliminates playout buffer delay, minimizing the reaction time of the system to user commands that change the AV stream.” EX1006, [0026].

206. A POSITA would have been motivated to combine the Hagai-Li combination with Park. As noted above, the Hagai-Li combination already discloses buffering received stream data and buffer management techniques, such as monitoring the levels of stored data in the buffer. EX1004, 6:10-14, 6:21-31, 16:48-51; *supra* §§ IX.B.12, IX.B.13. The combined system would further incorporate Park’s buffer management technique of flushing the audio and video data stored in the buffer. Since Hagai already discloses buffer management techniques, it would have been straightforward to add the additional buffer management technique taught

by Park. This additional technique would minimize playout buffer delays and improve responsiveness of the playback device of the Hagai-Li combination.

207. A POSITA would have had a reasonable expectation of success combining Hagai and Li with Park because Park uses the well-known technique—flushing from the buffer previously requested audio and video data that is no longer needed—to manage the efficiency of the playback device’s buffer. *Supra* § VII.A.1.d; *see also* EX1017, 13:49-61, Fig. 5. The combination of these teachings would have yielded predictable results because it involves conventional and well-known buffer management techniques in playback systems. *Supra* § VII.A.1.d.

#### **B. Claims 9 and 28**

**Claim 9: “The playback device of claim 1, wherein the application is further capable of causing the processor to perform the step of discarding audio and video data contained within the buffer in response to a received seek instruction”; and**

**Claim 28: “The method of claim 20, further comprising discarding audio and video data contained within the buffer in response to a received seek instruction.”**

208. Hagai in combination with Li and Park renders obvious claims 9 and 28.

209. For example, as discussed in limitation [1.b(x)], Hagai discloses that the playback device buffers received audio and video data, and a “reception buffer monitor” monitors the quantity of stored audio and video data in the buffer. EX1004, 6:10-14, 16:48-51; *supra* § IX.B.13 (Limitation [1.b(x)]). When the reception buffer monitor detects that the quantity of buffered data exceeds a predetermined value, the

playback device restarts playback of the buffered data. EX1004, 17:1-7 (“That is, data playback is halted until a predetermined quantity of data is stored in the reception buffer which has become empty (S1303). When a predetermined quantity of data is stored in the reception buffer, data playback for all streams is restarted (S1304).”). To the extent Hagai does not explicitly disclose discarding buffered data in response to a seek instruction, a POSITA would look to Park for its buffer management techniques to minimize latency and avoid playback of outdated data. *Supra* § VII.A.1.d; *see also* EX1017, 13:49-61, Fig. 5.

210. For example, Park discloses a “method of playout buffer management” that “flush[es] the playout buffer.” EX1006, [0012]. A POSITA would have understood that “flushing the playout buffer” of Park is equivalent to “discarding the audio and video data contained within the buffer” because a POSITA would have understood that both result in clearing out the previous audio and video data stored in the buffer of the playback device.

211. Park further discloses that the flushing is in response to a group of commands, such as a fast forward or rewind command. EX1006, [0012] (“flushing the playout buffer upon receipt of a command taken from the group of commands consisting of change of AV program, receipt of a new AV stream following a fast forward or fast reverse command, a pause command and a stop command”). A POSITA would have understood that Park’s flushing technique applies to a received

seek instruction because a seek instruction similarly leaves buffered data outdated by jumping to a new playback location. EX1006, [0026] (“The method of the invention also flushes the playout buffer of the receiving device anytime the current AV stream is terminated and a new AV stream is started. Flushing the playout buffer of residual, e.g., unwanted, AV program material eliminates playout buffer delay, minimizing the reaction time of the system to user commands that change the AV stream.”).

212. As discussed above, a POSITA would have been motivated to combine Park with the Hagai-Li combination because Hagai already discloses buffer management techniques, and incorporating the additional buffer management techniques disclosed in Park (e.g., flushing outdated audio data and video data from the buffer) would have been a straightforward modification to minimize playout buffer delays and improve the responsiveness of the playback device of the Hagai-Li combination. EX1006, [0023], [0027]; *supra* § X.A.

### **C. Claim 12**

#### **1. [12.pre] “A playback device, comprising:”**

213. For the reasons discussed above for limitation [1.pre], Hagai discloses limitation [12.pre]. *Supra* § IX.B.1 (Limitation [1.pre]).

#### **2. [12.a] “a processor; and”**

214. For the reasons discussed above for limitation [1.a], Hagai renders obvious limitation [12.a]. *Supra* § IX.B.2, (Limitation [1.a]).

3. ***[12.b] “a non-volatile storage containing an application for causing the processor to perform the steps of:”***

215. For the reasons discussed above for limitation [1.b], limitation [12.b] is unpatentable obvious. *Supra* § IX.B.3 (Limitation [12.b]).

4. ***[12.b(i)] “establishing at least one connection for communicating with a remote server system;”***

216. For the reasons discussed above for limitation [1.b(i)], limitation [12.b(i)] is unpatentable. *Supra* § IX.B.4 (Limitation [1.b(i)]).

5. ***[12.b(ii)] “obtaining information from a remote server system describing at least one video track, and at least one audio track;”***

217. For the reasons discussed above for limitation [1.b(ii)], limitation [12.b(ii)] is unpatentable. *Supra* § IX.B.5 (Limitation [1.b(ii)]).

6. ***[12.b(iii)] “selecting a video track from the at least one video track;”***

218. For the reasons discussed above for limitation [1.b(iii)], limitation [12.b(iii)] is unpatentable. *Supra* § IX.B.6 (Limitation [1.b(iii)]).

7. ***[12.b(iv)] “requesting a header describing the selected video track, where the requested header comprises a DRM header;”***

219. For the reasons discussed above for limitation [1.b(iv)], Hagai discloses requesting a header describing the selected video track. *Supra* § IX.B.7 (Limitation [1.b(iv)]); EX1004, 8:11-18.

220. Moreover, for the reasons discussed above for claim 3, Hagai in combination with Li renders obvious that the requested header comprises a DRM header. *Supra* § IX.C (Claim 3).

221. As discussed above in claim 3, it would have been obvious to a POSITA to store the DRM header of Li on the server of Hagai. EX1005, 14:40-44, 27:64-28:2, Fig. 2.2; *supra* § IX.C (Claim 3).

**8. [12.b(v)] “decrypting the DRM header;”**

222. For the reasons discussed above for claim 4, limitation [12.b(v)] is unpatentable. *Supra* § IX.D (Claim 4).

**9. [12.b(vi)] “selecting an audio track from the at least one audio track;”**

223. For the reasons discussed above for limitation [1.b(v)], limitation [12.b(vi)] is unpatentable. *Supra* § IX.B.8 (Limitation [1.b(v)]).

**10. [12.b(vii)] “obtaining index information indicating the locations of audio and video data within the selected audio and video tracks;”**

224. For the reasons discussed above for limitation [1.b(vi)], limitation [12.b(vii)] is unpatentable. *Supra* § IX.B.9 (Limitation [1.b(vi)]).

**11. [12.b(viii)] “determining byte ranges to request from the selected audio and video tracks using the index information;”**

225. For the reasons discussed above for limitation [1.b(vii)], limitation [12.b(viii)] is unpatentable. *Supra* § IX.B.10 (Limitation [1.b(vii)]).

**12. [12.b(ix)] “creating a buffer;”**

226. Hagai discloses limitation [12.b(ix)].

227. As discussed above for limitation [1.b(ix)], after sending byte range requests to the server, the playback device stores the received data in a buffer. Hagai, 10:4-7; *supra* § IX.B.12 (Limitation [1.b(ix)]). Specifically, Hagai discloses that the reception means 1806 of the playback device “receives a transmission response 1805 in response to the transmission request 1804 and stores the stream data of the transmission response 1805 in the reception buffer 1808.” EX1004, 10:4-7, Fig. 2. Hagai further discloses that the playback means plays the stream data from the buffer. EX1004, 10:8-10. A POSITA would have understood that, by receiving data and storing it for playback, Hagai’s system is “creating a buffer.”

**13. [12.b(x)] “requesting byte ranges from the selected video track and the selected audio track from the remote server system;”**

228. For the reasons discussed above for limitation [1.b(viii)], limitation [12.b(x)] is unpatentable. *Supra* § IX.B.11 (Limitation [1.b(viii)]).

**14. [12.b(xi)] “buffering received bytes of information comprising audio and video data;”**

229. For the reasons discussed above for limitation [1.b(ix)], limitation [12.b(xi)] is unpatentable. *Supra* § IX.B.12 (Limitation [1.b(ix)]).

**15. [12.b(xii)] “checking that sufficient data is buffered to commence playback;”**

230. For the reasons discussed above for limitation [1.b(x)], limitation [12.b(xii)] is unpatentable. *Supra* § IX.B.13 (Limitation [1.b(x)]).

**16. [12.b(xiii)] “decrypting encrypted frames of video using information from the decrypted DRM header;”**

231. For the reasons discussed above for claim 5, limitation [12.b(xiii)] is unpatentable. *Supra* § IX.E (Claim 5).

**17. [12.b(xiv)] “playing back the buffered audio and the decrypted video data;”**

232. For the reasons discussed above for limitation [1.b(x)] and claim 5, limitation [12.b(xiv)] is unpatentable. *Supra* §§ IX.B.13 (Limitation [1.b(x)]); IX.E (Claim 5).

**18. [12.b(xv)] “responding to a received seek instruction by:”**

233. For the reasons discussed above for limitation [1.b(xi)], limitation [12.b(xv)] is unpatentable. *Supra* § IX.B.14 (Limitation [1.b(xi)]).

**19. [12.b(xv)(1)] “pausing playback;”**

234. For the reasons discussed above for limitation [1.b(xi)(1)], limitation [12.b(xv)(1)] is unpatentable. *Supra* § IX.B.15 (Limitation [1.b(xi)(1)]).

**20. [12.b(xv)(2)] “discarding buffered audio and video data;”**

235. For the reasons discussed above for claim 9, limitation [12.b(xv)(2)] is unpatentable. *Supra* § X.B (Claims 9 and 28).

21. ***[12.b(xv)(3)] “determining byte ranges to request from the selected audio and video tracks based upon a new playback location using the index information;”***
22. ***[12.b(xv)(4)] “requesting byte ranges required to play the selected audio and video tracks from the new playback location from the remote server;”***
23. ***[12.b(xv)(5)] “buffering received bytes of information comprising audio and video data pending resumption of playback; and”***
24. ***[12.b(xv)(6)] “checking that sufficient data is buffered to commence playback;”***
25. ***[12.b(xv)(7)] “decrypting encrypted frames of video using information from the decrypted DRM header; and”***
26. ***[12.b(xv)(8)] “playing back the buffered audio and decrypted video data.”***

236. Upon receiving a new seek instruction, Hagai’s playback device repeats the same sequence of operations used for playback prior to the new instruction. It determines and requests new byte ranges corresponding to the new playback location, buffering the data, decrypting the encrypted video frames, and, once sufficient data is buffered, resumes playback from the new playback location. EX1004, 9:4-10:21, 11:5-35; *supra* § IX.E (Claim 5). These operations correspond to limitations [12.b(viii)] and [12.b(x)]-[12.b(xiv)] above, except that they are performed in relation to the “new playback location” selected by the client device. Accordingly, these elements are obvious for the same reasons set forth with respect to [12.b(viii)] and [12.b(x)]-[12.b(xiv)]. *Supra* §§ X.C.11, X.C.13-X.C.17.

- D. Claim 13: “The playback device of claim 12, wherein: the DRM header identifies at least one key; and the step of decrypting encrypted frames of video using information from the decrypted DRM header comprises decrypting encrypted frames of video using the at least one key prior to decoding.”**

237. For the reasons set forth with respect to [12.b(xiii)] and claim 5, claim 13 is unpatentable. *Supra* §§ IX.E, X.C.16

- E. Claim 14: “The playback device of claim 13, wherein: encrypted frames of video are only partially encrypted; the received bytes of information contain references to portions of video frames that are encrypted; and the step of decrypting encrypted frames of video using the at least one key prior to decoding further comprises decrypting encrypted portions of frames of video using the references to the portions of the video frames that are encrypted.”**

238. For the reasons set forth with respect to claim 6, claim 14 is unpatentable. *Supra* § IX.F.

- F. Claim 16: “The playback device of claim 12, wherein playing back the buffered video data from the requested byte ranges provides fast forward trick play.”**

239. For the reasons set forth with respect to claim 7, claim 16 is unpatentable. *Supra* § IX.G.

- G. Claim 17: “The playback device of claim 12, wherein playing back the buffered video data from the requested byte ranges provides fast rewind trick play.”**

240. For the reasons set forth with respect to claim 8, claim 17 is unpatentable. *Supra* § IX.H.

**H. Claim 18: “The playback device of claim 12, wherein the application is further capable of causing the processor to perform the steps of: obtaining information from a remote server system describing at least one subtitle track; selecting a subtitle track from the at least one subtitle track; requesting data from the selected subtitle track from the remote server system, where the data contains subtitle text; and superimposing the subtitle text over video during playback.”**

241. For the reasons set forth with respect to claim 10, claim 18 is unpatentable. *Supra* §§ IX.I.1, IX.I.2, IX.I.3.

**XI. GROUND 3: Claims 2, 11, 21, and 30 are obvious over Hagai, Li, Schmitz, and General Knowledge of a POSITA**

**A. Motivation to Combine**

242. As noted above, a POSITA would have been motivated to combine Hagai and Li for the reasons discussed in Ground 1. *Supra* § IX.A. A POSITA would have been further motivated to enhance the Hagai-Li combination by incorporating the server request management techniques taught by Schmitz for the reasons set forth below.

243. For example, Hagai describes a client-server streaming architecture that enables seeking to arbitrary positions in a media stream using a playback start position table to determine HTTP byte range requests. EX1004, 8:1-10; *supra* §§ IX.B.9-IX.B.11 (Limitations [1.b(vi)]-[1.b(viii)]). While neither Hagai nor Li explicitly discloses how to handle existing byte range requests rendered obsolete by a new seek instruction or uses asynchronous requests to the server for byte ranges from the selected video track and audio track, Schmitz does.

244. Schmitz is directed to navigation through multimedia content, including time-linear media, such as video and audio, and non-linear media, like encyclopedias. EX1007, 6:18-22 (“A mechanism according to the present invention permits a user to navigate around a modelled multimedia space as quickly as requested input events or instances are processed, and can provide media display service of user-selected quality in response to the requests.”), 1:25-30, 1:45-52 (“For a multimedia encyclopedia, for example, there is a linear dimension in the alphabetical ordering of items, and there is an X-Y space in the display of any given page. However, these dimensions have no mapping to the time-linear nature of a video or graphical sequence that might be displayed in association with a body of text about a given subject in the encyclopedia.”).

245. Prior systems typically restricted users to linear, sequential navigation. EX1007, 2:8-14. Schmitz’s system improves upon the prior art using cancelable asynchronous requests to allow arbitrary navigation and rapid content retrieval based on user interaction. EX1007, 1:25-30. Schmitz explains that “the present invention does not bind retrieval of the video context pictures synchronously to the user navigation,” thereby enabling fast system response to seek operations. EX1007, 13:30-32. These teachings naturally align with the Hagai-Li combination, which supports playback from arbitrary points.

246. Though Schmitz's techniques were novel to non-linear media (like encyclopedias), it states that the techniques apply equally to time-linear media (like video). EX1007, 1:67-2:7, 6:1-10. Traditionally, time-linear content was constrained by physical media like tapes, requiring sequential, synchronous operation. *Id.*, 1:67-2:7. Schmitz improves upon this by enabling random access to time-linear media through asynchronous requests, allowing users to seek freely without waiting for prior requests to be fulfilled. *Id.*, 6:15-22. This is functionally analogous to the jump playback operation supported by the Hagai-Li combination.

247. Schmitz discloses that the multimedia system includes a client device capable of canceling or flushing obsolete requests previously sent to the server in response to a user action. EX1007, 1:9-14 ("This invention relates to true multimedia server request mechanisms, and more specifically to a mechanism permitting prioritizing and cancellation of requests queued into such mechanisms to permit adaptive optimization of media server performance."), 6:58-65 ("Thus, the mechanism permits cancellation of outstanding queued requests that have become obsolete as the user navigates out of the area for which the request was needed."), 7:60-64, Abstract; *supra* § VIII.D. Schmitz also discloses that the client sends asynchronous requests to the server. EX1007, 7:60-64 ("In overview, the present invention provides a media server-client model that handles asynchronous media

service requests. . . .”); *see also id.*, 6:15-65, 21:56-59, 23:45-57, 4:60-67, 8:48-64, Claim 1; *supra* § VIII.D.

248. A POSITA would have been motivated to combine the Hagai-Li combination with Schmitz’s asynchronous request and request-flushing techniques. As noted above, Hagai already discloses sending HTTP requests for byte ranges of selected media tracks. A POSITA would have known that retaining stale or obsolete requests makes no technical sense and thus they would need to be discarded. To the extent a POSITA needed explicit teachings of such a technique, Schmitz expressly discloses such behavior, and a POSITA would be motivated to implement with Hagai and Li to provide for this functionality and reduce delays in communication between the server and playback device *Supra* § VII.A.1.c, VIII.D.

249. A POSITA would have also been motivated to incorporate Schmitz’ disclosure of asynchronous requests into the Hagai-Li combination. Because Hagai’s playback device already sends HTTP requests to the server, it would have been straightforward to implement Schmitz’s asynchronous requests because it would require only a simple modification of the Hagai-Li combination, which already includes a control means for controlling transmission requests, to include HTTP requests that can handle this type of operation. EX1004, 9:17-19, 9:63-67. For example, HTTP is extensible and request-header field names can be extended to add functionality such as flushing pending messages. The extensibility of HTTP’s

request methods was included in the original HTTP 1.1 specification in 1999 and was well known by 2007. EX1030, Abstract, 1. A POSITA would have understood that such a combination would improve the playback functionality of the Hagai-Li combination by decreasing server response time, reducing the number of outstanding server requests, and enabling prioritization of outstanding requests by the server. EX1007, 4:60-67, 6:62-7:5, 23:45-57.

250. A POSITA would have had a reasonable expectation of success implementing both modifications. Request flushing and asynchronous requests were known solutions in the field, and their integration would have predictably improved the performance and responsiveness of the Hagai-Li system. *Supra* § VII.A.1.c.

**B. Claims 11 and 30**

**Claim 11: “The playback device of claim [1], wherein the application is further capable of causing the processor to perform the step of flushing previous byte range requests in response to the received seek instruction.”**

**Claim 30: “The method of claim 20, further comprising flushing previous byte range requests in response to the received seek instruction.”**

251. Hagai in combination with Li and Schmitz renders obvious claims 11 and 30. As explained in limitation [1.b(viii)] and [12.b(x)], Hagai in combination with Li discloses that the playback device requests byte ranges from the selected

video and audio track from the remote server. *Supra* §§ IX.B.11, X.C.13 (Limitations [1.b(viii)] and [12.b(x)]).

252. To the extent that Hagai and Li do not disclose flushing previous byte range requests in response to a received seek instruction, Schmitz does. For example, Schmitz discloses a “multimedia server request mechanism” that “permits cancellation of outstanding queued requests that have become obsolete as the user navigates out of the area for which the request was needed.” EX1007, 1:10-14; Abstract. A POSITA would have understood that cancelling obsolete requests to be “flushing” them because it discards requests that are lower priority than new requests. Schmitz further discloses that the client cancels obsolete requests to the server related to subsets of data, such as subsets of bytes representing portions of images. EX1007, Abstract, 12:29-49. Accordingly, a POSITA would have understood that Schmitz’s request cancellation would apply to the byte range requests of Hagai-Li.

253. As discussed above, a POSITA would have been motivated to combine Schmitz with the Hagai-Li combination because a POSITA would know that obsolete requests would need to be discarded to maintain optimal server response times. *Supra* § XI.A. To the extent this would not be understood through Hagai alone, a POSITA would have looked to Schmitz for an explicit teaching and would have had a reasonable expectation of success because it would require only a simple

addition to the Hagai-Li combination of a known solution. EX1004, 9:17-19, 9:63-67; *supra* §§ VII.A.1.c, XI.A.

### **C. Claims 2 and 21**

254. The combination of Schmitz in view of Hagai and Li renders the device of claims 2 and 21 obvious. As discussed above, Hagai in view Li teaches every limitation of claims 1 and 20. *Supra* §§ IX.B, IX.J.

**Claim 2: “The playback device of claim [1], wherein the application is further capable of causing the processor to asynchronously request byte ranges from the selected video track and the selected audio track from the remote server.”**

**Claim 21: “The method of claim 20, further comprising asynchronously requesting byte ranges from the selected video track and the selected audio track from the remote server using the playback device.”**

255. Hagai in combination with Li and Schmitz renders obvious claims 2 and 21.

256. As explained in limitation [1.b(viii)], Hagai in combination with Li discloses that the playback device requests byte ranges from the selected video track and selected audio track from the remote server. *Supra* § IX.B.11 (Limitation [1.b(viii)]).

257. At the time of the invention, requests could be sent synchronously or asynchronously. *Supra* § VII.A.1.c. Thus, a POSITA would have understood that the Hagai-Li combination teaches these limitations based on the knowledge of a skilled artisan, even without explicit disclosure about this type of request. But even

if asynchronously requesting byte ranges is not expressly disclosed by Hagai or Li, it is in Schmitz, which a POSITA would have been motivated to look to for detail about incorporating this standard functionality.

258. A POSITA would have understood that Schmitz discloses asynchronous requests because it enables sending new requests independent of the status of prior requests. EX1007, 21:56-57; EX1001, 2:31-34; *supra* § VII.A.1.c.

259. For example, Schmitz discloses a multimedia system including a client module that sends asynchronous requests to the multimedia server. EX1007, 6:15-65, 21:56-59, 8:48-64, Claim 1 (“[a]n asynchronous media server request processing system for presenting information from media servers in response to media access requests from a client”). A POSITA would have understood that the client module generates asynchronous requests because the client module can generate and transmit requests to the multimedia server independent of outstanding requests. EX1007, 6:23-31 (“The present invention includes a client module that generates access requests to the media server, and a server module that includes the media server[.] According to the invention, each request defines a client object for every active multimedia instance. The media server queues the requests, which it processes as the requests reach the top of the queue.”), 21:56-59. A POSITA would further understand that the requests transmitted by the client module are asynchronous

requests because the client module can prioritize, re-prioritize, and cancel requests pending at the multimedia server.

Because the present invention permits cancelling a queued request that has not completed, requests may be prioritized and even re-prioritized at any time they are in the queue. This aspect of the present invention enhances overall system performance. Priority can be assigned according to expense of system performance (e.g., relative amount of video memory required for the image, relative time to complete the image), and/or according to user benefits (e.g., how important is it to now view a fully rendered image as opposed to a wireframe image, or a black and white image as compared to a full color image). High priority requests (those requiring less media resources/time) can be queued before lower priority requests.

EX1007, 23:45-57; *see also id.*, 21:56-59 (“Further, because multiple requests can be outstanding, the client callback must be passed either the request ID or the client state object to advise the client module which picture request has been completed.”).

260. As discussed above, a POSITA would have been motivated to combine Schmitz with the Hagai-Li system because Hagai already discloses sending HTTP requests for byte ranges of selected video and audio tracks, and sending asynchronous requests to the server was a well-known and conventional technique. *Supra* §§ VII.A.1.a, VII.A.1.c, XI.A. A POSITA would have further understood that such a combination would have been straight forward and would have had a reasonable expectation of success because it would require only a simple modification to the Hagai-Li combination, which already includes a control means

1801 for controlling transmission requests. EX1004, 9:17-19, 9:63-67; *supra* §§ VII.A.1.c, VIII.A, XI.A.

**XII. GROUND 4: Claims 15 and 19 are obvious over Hagai, Li, Park, Schmitz, and General Knowledge of a POSITA**

**A. Motivation to Combine**

261. A POSITA would have been motivated to combine Hagai, Li, and Park for the reasons stated in Grounds 1 and 2. *Supra* §§ IX.A, X.A. A POSITA also would have been motivated to combine Schmitz with Hagai and Li as explained above in Ground 3. *Supra* § IX.A. A POSITA would have further been motivated to combine Hagai, Li, and Park with Schmitz. A POSITA would be motivated to combine both Park and Schmitz with the Hagai-Li combination because they introduce independent techniques to reduce delays in different aspects of the data transmission process. While Schmitz teaches techniques for improving response time to requests being sent out by the client device (i.e., it makes efficiency improvements in communications with the server), Park offers an additional technique to improve efficiency of received data, by minimizing playout buffer delays and improving responsiveness of the playback device (i.e., improvements to buffer management and playback device performance). *Supra* §§ IX.A, X.A, XI.A. As a result, the implementation of both techniques would improve the overall performance of the Hagai-Li combination.

## **B. Claim 15**

262. The combination of Schmitz in view of Hagai, Li, Park, and Schmitz renders claim 15 obvious. As discussed above, Hagai in view of Li and Park teaches every limitation of claim 12. *Supra* § X.C.

**Claim 15: “The playback device of claim 12, wherein the application is further capable of causing the processor to asynchronously request byte ranges from the selected video track and the selected audio track from the remote server.”**

263. For the reasons discussed above for claims 2 and 21, Hagai in combination with Li, Park, and Schmitz renders obvious claim obvious. *Supra* § XI.C. As noted above, the inclusion of Park in the Hagai-Li combination does not impact the combination of Schmitz with Hagai and Li because it teaches how to efficiently handle received data and has no effect on Schmitz’s techniques for improving requests made by the client.

## **C. Claim 19**

264. The combination of Schmitz in view of Hagai, Li, Park, and Schmitz renders claim 19 obvious. As discussed above, Hagai in view of Li and Park teaches every limitation of claim 12. *Supra* § X.C.

**Claim 19: “The playback device of claim 12, wherein the application is further capable of causing the processor to perform the step of flushing previous byte range requests in response to the received seek instruction.”**

265. For the reasons discussed above for claims 11 and 30, Hagai in combination with Li, Park, and Schmitz renders obvious claim 19. *Supra* § XI.B.

As noted above, the inclusion of Park in the Hagai-Li combination has no impact on the combination of Schmitz with Hagai and Li. *Supra* § XII.A.

### **XIII. CONCLUSION**

266. For the reasons I have presented above, it is my opinion that the references serving as the bases for the grounds of rejection in the Petition render claims 1-30 of the challenged patent unpatentable.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further 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.

Dated: June 30, 2025

  
\_\_\_\_\_  
Hari Kalva, Ph.D.