Roman M. Silberfeld, SBN 62783
RSilberfeld@RobinsKaplan.com
Daniel L. Allender, SBN 264651
DAllender@RobinsKaplan.com
ROBINS KAPLAN LLP
2049 Century Park East, Suite 3400
Los Angeles, CA 90067
Telephone: (310) 552-0130
Facsimile: (310) 229-5800

David M. Stein, SBN 198256
DStein@BrownRudnick.com
BROWN RUDNICK LLP
2211 Michelson Drive, 7th Floor
Irvine, CA 92612
Telephone: (949) 440-0231
Facsimile: (949) 486-3686

Christopher A. Seidl
(*pro hac vice* to be filed)
CSeidl@RobinsKaplan.com
Aaron R. Fahrenkrog
(*pro hac vice* to be filed)
AFahrenkrog@RobinsKaplan.com
Shui Li (*pro hac vice* to be filed)
SLi@Robinskaplan.com
ROBINS KAPLAN LLP
800 LaSalle Avenue, Suite 2800
Minneapolis, MN 55402
Telephone: (612) 349-8500
Facsimile: (612) 339-4181

Attorneys for Plaintiff
DivX, LLC

ROBINS KAPLAN LLP
ATTORNEYS AT LAW
LOS ANGELES

UNITED STATES DISTRICT COURT
CENTRAL DISTRICT OF CALIFORNIA

DIVX, LLC, a Delaware limited liability company,

Plaintiff,

v.

HULU, LLC, a Delaware limited liability company,

Defendant.

Case No. 2:21-cv-01615

COMPLAINT FOR PATENT INFRINGEMENT

DEMAND FOR JURY TRIAL

Plaintiff DivX, LLC ("Plaintiff" or "DivX"), by its attorneys, for its complaint ("Complaint") against Defendant Hulu, LLC ("Defendant" or "Hulu") for patent infringement alleges as follows:

INTRODUCTION

1. This complaint addresses Hulu's large-scale use of DivX's patented technologies that relate to video streaming over the internet. DivX's inventions enable streaming of episode- and feature-length video content securely and to a wide variety of playback devices, resulting in an excellent user experience. DivX developed these technologies to improve the way internet users experience video streaming and to bring video streaming securely to many devices. DivX's engineers developed these technologies a decade ago, overcoming the technical challenges imposed by computing capabilities and internet bandwidth that were a fraction of what they are today. DivX's investments in these innovations are protected by multiple patents issued by the United States Patent and Trademark Office. DivX recognizes that video streaming over the internet has dramatically improved the consumer entertainment experience, allowing consumers to watch a large library of content anywhere and at any time. Hulu, however, may not use, without permission, the innovations that DivX spent years developing to overcome the technical hurdles that stood in the way. DivX brings this lawsuit to address Hulu's infringement and defend DivX's investments in video streaming technologies.

THE PARTIES

2. DivX is a Delaware limited liability company. Its principal place of business is 4350 La Jolla Village Drive, Suite 950, San Diego, California, 92122. DivX owns patents covering foundational internet video streaming technologies, including those asserted here.

3. Hulu, LLC, is a Delaware limited liability company[1] with its corporate address at 2500 Broadway, Santa Monica, California, 90404.[2]

4. Hulu promotes itself as a leader in the market for streaming video content, which includes films, television series, and other video content. Upon information and belief, Hulu designs, operates, tests, manufactures, uses, offers for sale, sells, and/or imports into the United States video streaming software and services that generate billions of dollars of revenue for Hulu each year.

NATURE OF THE ACTION

5. This Complaint alleges patent infringement. DivX alleges that Hulu has infringed and continues to infringe, directly and/or indirectly, five DivX patents: 10,257,443 (the "'443 patent"), U.S. Patent Nos. 9,794,318 (the "'318 patent"), 10,412,141 (the "'141 patent"), 10,542,061 (the "'061 patent"), and 10,326,987 (the "'987 patent"), copies of which are attached as Exhibits 1-5 (collectively, the "DivX Patents").

6. The DivX Patents cover foundational internet video streaming technologies that are necessary for Hulu's streaming platform to deliver high-quality video immediately upon the touch of a button, anywhere and on any device. The DivX Patents disclose technologies that enable many benefits, including:

- the inventions claimed in the '443 patent speed up decryption of securely encrypted video data, which reduces the computing resources needed for the playback device and makes video playback start more quickly, while protecting the content from piracy;
- the inventions claimed in the '318 patent allow streaming video to start playing more quickly and allow viewers to use trick play functions right away upon startup, improving the user experience;

[1] https://icis.corp.delaware.gov/Ecorp/EntitySearch/NameSearch.aspx.

[2] https://www.bloomberg.com/profiles/companies/3276470Z:US-hulu-llc.

- the inventions claimed in the '141 patent allow streaming video with multiple tracks to start playing more quickly and allow viewers to use trick play functions right away, improving the user experience;
- the inventions claimed in the '061 patent enable playback devices to select video streams based on the device's specific capabilities while also making protected video streams easier to decrypt, providing optimal video quality for different devices while maintaining playback startup speed and protecting the content from piracy; and
- the inventions claimed in the '987 patent provide video streaming users with higher-quality video and fewer video stalls for a more seamless user experience.

7. Hulu directly infringes the DivX Patents by making, using, offering to sell, selling, and/or importing into the United States internet video streaming technology, software, and services that practice the inventions claimed in the DivX Patents.

8. Hulu indirectly infringes at least three of the DivX Patents by inducing its consumer end-users to directly infringe those DivX Patents. Hulu induces infringement by providing software (the Hulu application) that, when used by viewers to stream video to televisions, personal computers, phones, tablets, and other devices, as directed and intended by Hulu, causes those users to make, use, and practice the inventions claimed in the DivX Patents.

9. DivX seeks damages and other relief for Hulu's infringement of the DivX Patents.

JURISDICTION AND VENUE

10. This is an action for patent infringement under the Patent Laws of the United States, 35 U.S.C. § 1 et seq., over which this Court has subject matter jurisdiction pursuant to 28 U.S.C. §§ 1331 and 1338(a).

11. This Court has both general and specific jurisdiction over Hulu because Hulu is headquartered[3] and has committed acts within the Central District of California giving rise to this action and has established minimum contacts with this forum such that the exercise of jurisdiction over Hulu would not offend traditional notions of fair play and substantial justice. Defendant Hulu, directly and through subsidiaries and intermediaries (including distributors, retailers, franchisees and others), has committed and continues to commit acts of patent infringement in this district, by, among other things, making, using, testing, selling, licensing, importing, and/or offering for sale/license products and services that infringe the DivX Patents.

12. Venue is proper in this district and division under 28 U.S.C. §§ 1391(b)-(d) and 1400(b) because Hulu has committed acts of infringement in the Central District of California and has a regular and established physical place of business and employees in the Central District of California.

FACTUAL BACKGROUND

I. DivX

13. For over 20 years, DivX has invested in developing innovative technology for delivering digital video over the internet. In 2000, delivering digital video over the internet to consumers presented many technical challenges without existing solutions. Jérôme Rota and Jordan Greenhall founded DivX to address these challenges and improve the consumer digital video experience.

14. DivX recognized that consumers wanted *accessible*, *high-quality* digital video content. To satisfy this demand, DivX created a new implementation of the MPEG-4 video standard. DivX completed this new implementation in 2001 and released it as the DivX Codec 4.0. A "codec" is a computer program for encoding—that is, compressing—and decoding digital video files. Over the next

[3] https://www.bloomberg.com/profiles/companies/3276470Z:US-hulu-llc.

decade, DivX developed and released numerous new and improved versions of the DivX Codec. DivX bundled the DivX Codec with other features for video encoding, decoding, and playback and packaged it as the "DivX Software."

15. In addition to providing the DivX Codec, the DivX Software functioned like a master translator for digital video files, allowing for variations in codecs, containers, and playback across different file types on different devices. It allowed consumers to compress, decode, and play back digital video using a single program that could allow users to access and use the variety of technologies available on the market, all in one place.

16. During the same period that DivX continuously evolved and improved its DivX Codec and DivX Software, consumer access to and use of digital video over the internet became more widespread as computing power and network bandwidth increased. These developments led to widespread adoption of the DivX Software, a large base of DivX users, and creation of billions of DivX video files (with the ".divx" file extension).

A. DivX Digital Rights Management (DRM)

17. Digital Rights Management, or DRM, is the foundation of many DivX innovations. A robust DRM system allows owners of video content (like studio movies) to control access to the video content and provide increased protection against piracy. DRM is therefore fundamental to distribution of video over the internet, because DRM enables secure downloading and playback of videos.

18. In 2000, when DivX began creating an internet video platform, content owners such as Hollywood studios would not release their premium video content on an internet platform because they feared that piracy and losing control of their content would severely diminish the value of their rights.

19. From 2000 to 2005, DivX met with content owners such as Disney, Warner Bros., Sony, Universal, and Paramount Pictures about technical solutions to overcome their concerns and to implement the strict security requirements that the

owners demanded. During the same time period, DivX also met with major consumer electronics (CE) manufacturers about overcoming challenges to implementing DRM features in CE devices. DivX recognized at the time that existing technologies would not meet the content protection concerns of studios, and it had to innovate to serve that market need.

20. DivX's engineers worked to build a DRM system that would solve these long-standing technical problems, and as a result of DivX's research and development efforts, DivX DRM became one of the first DRM systems accepted by major Hollywood studios.

B. DivX Open Video System (OVS)

21. DivX was one of the first companies in the world to create a commercially viable internet video streaming platform, called the "Open Video System" ("OVS"). DivX OVS was an internet-based video-on-demand system that built upon the quality and performance of the DivX Software. Eric Grab, a named inventor on the '443 patent, was the lead of the engineering team that discovered the innovations necessary to build the platform. DivX OVS officially launched on September 6, 2001, at a time where broadband internet access was not yet ubiquitous, and in a business environment where Hollywood studios were not yet ready to embrace digital distribution. After the launch of DivX OVS, DivX engineers continued to invest in technical improvements and innovations for the platform, and their innovations expanded the platform to enable playback on a wide variety of playback devices.

22. DivX's investments in OVS produced many key innovations for delivering video over the internet:

- A flexible, key-based DRM system that tied purchased video content to a viewer rather than a device, preventing unauthorized access when the device was sold or obtained by others while improving the viewer experience.

- A core codec that offered industry-best compression and performance enabling full-screen, DVD-like quality that was vastly superior to the pixelated, postage-stamp size viewing experience associated with internet video at the time.
- A "progressive download" feature that allowed the viewer to begin watching a purchased or rented video after only a few minutes while the file continued to download in the background.

23. DivX OVS was a successful video streaming platform. Throughout the mid-2000s, hundreds of millions of devices spanning virtually every major consumer electronics manufacturer were released supporting DivX OVS playback. Blockbuster, Netflix, Amazon, and others approached DivX about using DivX's technology to power their streaming platforms.

C. DivX Stage6

24. In 2006, DivX launched "Stage6" —one of the first HTTP-based websites for sharing and streaming high-resolution video. Streaming video from an HTTP-based website allows a web server to continuously send data to a viewer over a single HTTP connection that remains open. DivX Stage6 implemented DivX's video compression, codec, and playback technology in an HTTP-based environment and allowed users to upload, share, and view larger video files than other competing platforms from that time, like YouTube.

25. A key technical innovation from DivX's efforts leading to Stage6—the improved multimedia file structure claimed in the '443 patent—became the foundation of HTTP-based adaptive streaming systems. Specifically, the'443 patent enabled secure HTTP progressive download, which formed the basis of HTTP-based adaptive streaming systems.

26. DivX Stage6 was one of the earliest websites that supported sharing and streaming of high-resolution video. Even in 2007, when computing resources and network bandwidth were far more limited than today, DivX Stage6 supported

streaming of 720p and 1080p high-definition video. The quality of the high-resolution video playback on Stage6 surprised reviewers, with one commenting "DivX has clearly got something right with web playback of higher-resolution video!"[4] DivX Stage6 enjoyed rapid user traffic growth, and by January 2008, it had over 10,000,000 monthly views.

D. DivX SDKs and CTKs

27. The success of the DivX Software and the DivX Codec created consumer demand to be able to play DivX video files on many CE devices, including DVD and other media players. To meet CE manufacturers' needs to satisfy this demand, DivX created CE software development kits ("SDKs") that would allow DVD players and other media players to play DivX files (from CD, DVD, USB, or network) while also incorporating a secure DRM protocol to protect against piracy and offering a variety of other features that created a high-quality video playback experience. DivX also developed Certification Test Kits ("CTKs") for CE manufacturers to certify their licensed devices and communicate to customers that their devices were compatible with DivX files. DivX SDKs and CTKs incorporated DivX's video compression, codec, playback, and DRM technology to provide an enjoyable user experience on a wide range of devices.

28. DivX has licensed its SDKs and CTKs to many CE companies. DivX's innovative technologies have been integrated into more than one and a half billion CE devices via the DivX SDKs and CTKs. To this day, numerous CE companies licenses DivX's SDKs and CTKs, including leading digital television, smartphone, in-car video device, DVD / Blu-ray disc, integrated circuit (IC), and other CE device manufacturers.

[4] *DivX Stage6 (beta)—the high-def rival to YouTube*, Hexus.net, May 1, 2007.

E. DivX Plus Streaming

29. In 2011, DivX released the DivX Plus Streaming SDK, an end-to-end internet video streaming software that rivaled Blu-ray DVDs in quality and feature-set (such as user commands for seeking in the video, fast-forward, and rewind). The DivX innovations incorporated in DivX Plus Streaming include several that provide the foundation for the widespread technological success of video streaming today.

30. DivX Plus Streaming was one of the earliest secure streaming software packages that supported Dynamic Adaptive Streaming over HTTP, abbreviated "DASH." DASH standardizes certain aspects of adaptive bitrate streaming of video over the internet and has been widely adopted as a protocol used by many of today's video streaming services. Fast start and smooth switching among video streams of different resolutions, depending on bandwidth, both improve the viewer experience during DASH. The innovations incorporated in DivX Plus Streaming improve both of these aspects of the streaming user experience.

31. DivX's engineers' efforts to create DivX Plus Streaming produced many innovations fundamental to today's video streaming services, including:

- Adaptive bitrate streaming that delivered video streams configured for each specific screen size on which the user wanted to watch the video. Configuring video streams based on the characteristics of individual playback devices ensures the optimal balance of video quality and playback performance.
- Trick play that allowed the user to pause, rewind, fast-forward, seek to a new point in the video, and quickly resume playback. Although now common in today's streaming platforms, these features required significant innovations to deliver them in 2011, when the streaming playback experience was far more limited.

II. Hulu

32. Hulu launched its video streaming service in 2008 and claims to be a "leading premium streaming service offering live and on-demand TV and movies, with and without commercials, both in and outside the home."[5]

33. Hulu has implemented additional technical features and capabilities in its service over time, and its service today differs significantly from the service it offered in 2008. For example, Hulu did not adopt a scalable HTTP-based adaptive streaming solution until 2013, when it adopted DASH.[6]

34. Today, Hulu depends on an enjoyable viewer experience to maintain and increase demand for its service. Hulu Senior Vice President and Head of Experience Ben Smith has explained that "[t]he future of the television experience is all about predicting what the consumers want."[7] To fuel this vision, Hulu has a dedicated Viewer Experience Center, which is the second largest Hulu office worldwide, just behind its headquarters.[8] It hosts "the technical aspect of ensuring which streams are coming through at the quality level that [Hulu's] customers want."[9]

[5] https://press.hulu.com/corporate/.

[6] https://www.streamingmediaglobal.com/Articles/Editorial/Featured-Articles/Hulus-Move-to--DASH-105110.aspx?utm_source=feedburner&utm_medium=feed&utm_campaign=Feed%3A+StreamingMediaMagazine-FeaturedArticles+%28StreamingMedia.com%3A+Featured+Articles%29.

[7] https://www.bizjournals.com/sanantonio/news/2017/11/28/hulus-new-san-antonio-office-goes-online-slideshow.html.

[8] https://careers.hulu.com/san-antonio.

[9] https://www.bizjournals.com/sanantonio/news/2017/11/28/hulus-new-san-antonio-office-goes-online-slideshow.html.

A. Technologies Implemented by Hulu

35. Hulu's viewer experience today depends on a combination of video compression, DRM, and adaptive streaming technologies to achieve device interoperability, secure playback, and fast and smooth streaming.

36. Hulu strives to deliver its service to many different user devices.[10] For example, Hulu touts that members can "watch thousands of shows and movies anytime, anywhere"[11]—indicating support for mobile devices. Further, Hulu advertises that "[a]s a Hulu subscriber, you'll be able to stream your favorite content from the comfort of your very own couch to the confines of public transportation using any one of our supported devices."[12] Hulu streaming software allows users to watch content using a variety of internet-connected devices that support the Hulu service, including smart TVs, game consoles, streaming media players, smartphones, and tablets.[13] Hulu supports numerous playback devices and operating systems in its video streaming ecosystem.[14]

37. Hulu relies on DRM technology to control the playback of copyrighted material. In particular, Hulu states that "content protection is critical for both MPEG-DASH and HLS formats."[15] DRM enables Hulu to reduce piracy, protect investments in content, and license content from third parties.

---

[10] https://medium.com/hulu-tech-blog/building-the-hulu-experience-in-the-living-room-10eabf5391d6.

[11] https://www.hulu.com/welcome?orig_referrer=https%3A%2F%2Fwww.google.%20com%2F.

[12] https://help.hulu.com/en-us/supported-devices.

[13] *Id.*

[14] *Id.*; https://medium.com/hulu-tech-blog/introducing-the-hulu-technical-landscape-93f4c136c568.

[15] https://medium.com/hulu-tech-blog/the-challenges-of-live-linear-video-ingest-part-one-live-versus-on-demand-system-requirements-89238f3af4f6.

38. Hulu employs video compression and DRM techniques to optimize delivery of content at maximum quality and speed.[16] Hulu also implements those techniques adaptively using HTTP, so that its service can automatically adjust to a variety of alternative quality levels depending on the user's internet speed, ensuring the best quality possible for streaming.[17]

B. Hulu's Reliance on DivX Technology

39. A sudden drop in network bandwidth can cause videos to freeze, buffer, or play with poor quality. Viewers expect video playback and quality to resume immediately after the network bandwidth increases. Hulu has adopted DivX's patented technology (e.g., the '987 patent) that optimizes video quality and reduces the number of video stalls during network bandwidth changes by selecting optimally encoded video streams based on the network data rate and the data stored in a buffer on the playback device.

40. Viewers prefer immediate playback after jumping to a new location in a video by fast-forwarding, rewinding, or seeking ("trick play"). Hulu has adopted DivX's patented technology (e.g., the '318 and '141 patents) that reduces trick play delay by implementing a client application at the player to commence playing video content and to request non-sequential portions of the video file without receiving the complete video file, while providing support for alternative audio tracks and subtitle tracks.

41. Content security protects videos from being pirated or viewed without authorization. Viewers expect computationally demanding content security

[16] https://medium.com/hulu-tech-blog/the-challenges-of-live-linear-video-ingest-part-one-live-versus-on-demand-system-requirements-89238f3af4f6.

[17] https://help.hulu.com/en-us/video-quality; https://www.digitaltrends.com/movies/how-does-hulu-work/; https://medium.com/hulu-tech-blog/the-challenges-of-live-linear-video-ingest-part-one-live-versus-on-demand-system-requirements-89238f3af4f6.

processes—e.g., authentication, encryption, and decryption—to remain in the background without damaging the quality of experience. Hulu has adopted DivX's patented technology (e.g., the '443 patent) that provides new structures and organizations of video and decryption information in multimedia files and thus reduces the processing resources needed to download, buffer, decrypt, and play back video.

42. Viewers also expect to stream video, with a high quality of experience, on any and all computing devices, such as laptops, smartphones, tablets, televisions—anything with a screen. Hulu has adopted DivX's patented technology (e.g., the '061 patent) that introduces new top level index file structures for efficiently interconnecting device-specific video streams and cryptographic encryption data to reduce the computing burdens on the streaming ecosystem while optimizing quality of experience on each different playback device.

43. Startup delays test viewers' patience and make them less likely to want to use the video streaming platform. But launching a video with poor quality to make it start faster also frustrates viewers. To balance these competing considerations, Hulu has adopted DivX's patented technology (e.g., the '061 patent) that reduces startup delay by implementing a top level index file that references common cryptographic information used to encrypt and decrypt the video. The result is that a consumer can quickly launch a high-quality video.

44. DivX initially met with Hulu in 2008 to discuss DivX's technology. DivX also met with Hulu in May 2019 to engage in good faith negotiations for Hulu to take a license to DivX's patents. Hulu has not obtained a license to DivX's patents or its technology, and Hulu continues to profit by providing its infringing video streaming service to tens of millions of Hulu subscribers without permission and without any compensation to DivX.

THE DIVX PATENTS

45. DivX solely owns all rights, titles, and interests in and to the DivX Patents, each described below.

I. Technical Background of Streaming Video

46. The DivX Patents are directed to improvements to computer systems for video streaming. Video "streaming" refers to the computing process of continuously providing digital video to an end user through a computing device.

47. Video streaming is accomplished by providing digital video files from server computers that host (store) the video files, over the interconnected computer networks that make up the internet, to client computers (consumer devices, such as desktop computers, laptop computers, smartphones, and smart televisions) that can interpret the video files and convert them to pixels displayed on the screen during playback.

48. The ability to perform video streaming, and the level of performance that can be provided to an end user (such as high-resolution, smooth playback, without stalls or errors), depends on the computing resources of the computing devices—server computers, network computers, and client computers—used in the video streaming system. Those computing resources include the processing power of the computers, the input/output (I/O) and data transmission capabilities of the computers, and the memory (storage) available on the computers.

49. Before digital video, video was stored on analog media such as tape. Transition from analog media to digital video brought new challenges. For example, the amount of data required to represent a video in digital form at its full recorded resolution is massive. The computing resources of servers, networks, and client computers, however, are limited. Streaming digital video, therefore, requires computing techniques to reduce the amount of data that must be processed by server computers, transmitted over networks, and interpreted and converted to displayed video by client devices. These techniques are generally referred to as

"encoding" (converting the data to a particular digital format) and "decoding" (translating the digital format to a format that can be rendered and displayed on a display device).

50. Video encoding and decoding rely on a computing technique called "compression" to reduce the size of the digital video files that must be processed and transmitted while simultaneously preserving sufficient playback performance and quality on the client device. Video compression employs data compression techniques specific to digital video content to reduce file sizes while maintaining playback quality. Because digital video is frequently represented as a series of still image "frames" played back quickly (for example, at a rate of 30 frames per second), video compression techniques take advantage of similarities among pixels in a single frame (spatial redundancy) and similarities among pixels across different frames (temporal redundancy) to reduce the amount of data that must be stored in the digital video file, transmitted over computer networks, and decoded by the client computer and converted to pixel data for display during playback.

51. Video compression presents unique computing challenges different from other data compression techniques (for example, audio, text, or pictures). Consumers expect the same high-quality experience from video streamed on smartphones, televisions, and personal computers as they do from cable programming and physical media (such as DVD or Blu-ray discs). Video compression, digital video files, encoding, and decoding techniques, therefore, must overcome the technical challenge of delivering superior video quality on all types of devices in the most bandwidth-efficient way possible with the least latency (the time it takes to transmit the digital video files over the network).

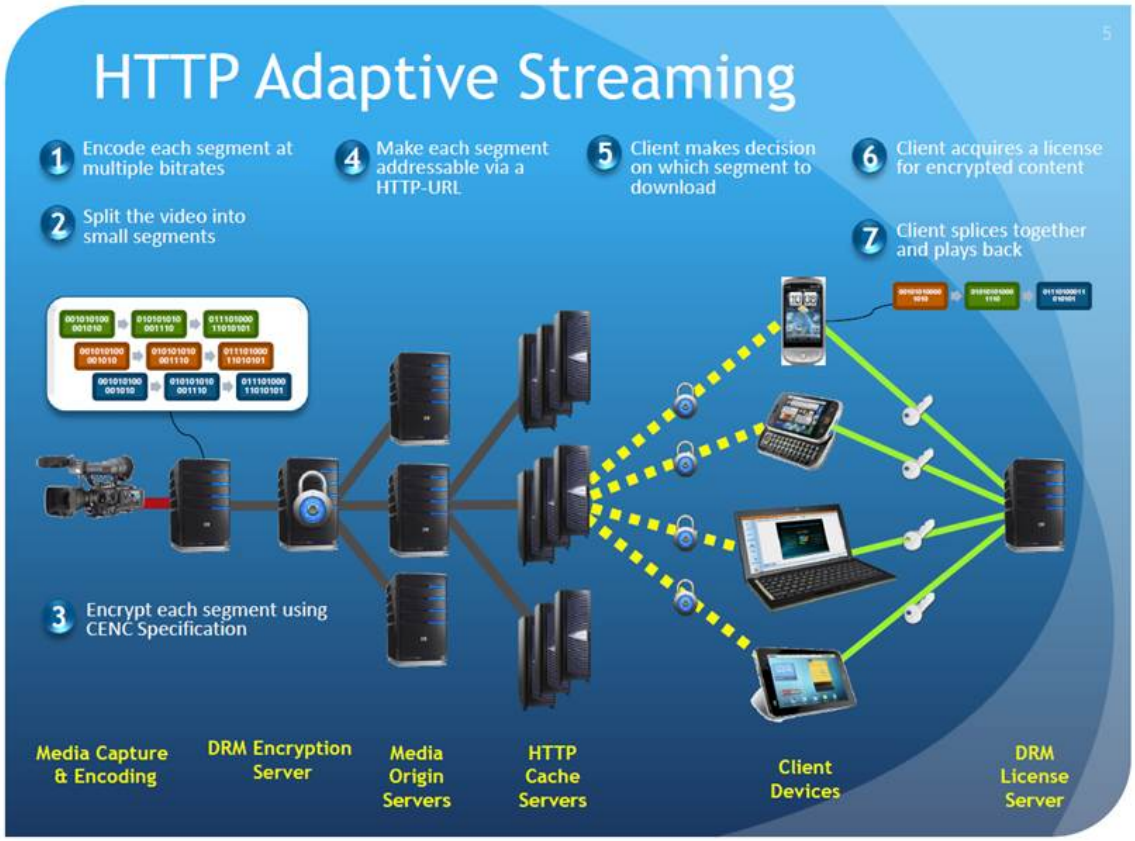
52. Video compression techniques produce specific types of computer files for representing video data. These files include data structured in a certain, defined way to represent both the video data and other information required to effectively decompress, decode, and play back the video on the display of a

playback device (client computer). Examples of compressed video files used for video streaming include DivX files, AVI files, MP4 files, and Matroska files.

53. Preventing piracy of digital video is another significant technical challenge for streaming video. Digital rights management (DRM) is an access control method that has been developed to protect digital media. DRM is designed to prevent the end user that has obtained digital media from modifying, copying, converting, or using the digital media in any way other than that permitted by the digital content provider. DRM often includes encryption of digital video data in specific ways using specific encryption structures and encryption and decryption mechanisms. Video streaming involves sending portions of files over a network for decryption and decoding on devices on which other software may be executing. Video streaming technology providers face unique technical challenges in providing adequate security of the video content and control over access rights while reducing the burdens on the encoding and decoding computers relating to encryption and decryption.

54. Adaptive bitrate streaming (ABS) is a specific technique used when streaming multimedia over computer networks to playback devices. ABS differs from other types of streaming because it involves detecting the streaming conditions in real time and adjusting the quality of the streamed media accordingly so the user does not experience stalls in video playback caused by changes in bandwidth or processing capabilities. For ABS, the playback server system encodes a particular video title in separate, multiple streams, at different bitrates, to be streamed consistent with the capabilities of the network and playback device, including bandwidth. If available bandwidth changes, for example, ABS allows the device to switch to a lower-resolution stream of the same video data, which requires less data transmission and processing. This allows the video content to keep playing the video without any stall. The process of stream switching in ABS requires the

ability to seek to a particular location in a video file and commence playback without access to all of the preceding portions of the file.[18]



55. Trick play, or trick mode as it is sometimes called, is a feature that gives viewers visual feedback while they are rewinding or fast-forwarding a stream (i.e., 'scrubbing' through it). For time-based trick mode, users are presented with a progress bar that displays their location in the content and allows them to seek to the desired timestamp using the standard trick play controls. When scene information is not available, a user only has a visual timeline and numeric time information to locate the desired position in the content. Once the new location is selected, the system buffers a minimal amount of stream data and begins playback. A scene-based trick mode is based on the availability of scene information in the form of BIF-files or a common file format for carousels of still frame images that

[18] https://dashif.org/docs/DASH264-v1.5.pdf.

are often used in systems that implement HLS or DASH. If such data are accessible for a given title, the scene-based trick mode can be used during playback.

56. In sum, streaming digital video data presents unique technical challenges relating to video compression and content protection that affect the computing systems that encode and encrypt digital video, the digital video file types created by those computing systems, and the computing systems that process those file types to decrypt and decode the digital video to provide streaming users with a high-quality experience. DivX's patented inventions provide technical solutions, through computing improvements, to these technical challenges.

II. The '443 Patent

57. The '443 patent, entitled "Multimedia Distribution System for Multimedia Files with Interleaved Media Chunks of Varying Types," duly and legally issued on April 9, 2019, with Jason Braness, Jerome Rota, Eric William Grab, Jerald Donaldson, Heather Hitchcock, Damien Chavarria, Michael John Floyd, Brian T. Fudge, and Adam H. Li as the named inventors. The '443 patent claims priority to U.S. Patent No. 7,519,274, filed on December 8, 2003. The '443 patent also claims priority to Patent Cooperation Treaty Patent Application No. PCT/US04/41667, filed on December 8, 2004.

Summary of the '443 Inventions

58. The inventions claimed in the '443 patent speed up decryption of securely encrypted video data, which make video playback start more quickly. The '443 inventions achieve this goal by introducing a file structure that includes a set of DRM chunks, an index, and partial frame encryption, to enable secure video streaming and faster playback for secure media.

59. Specifically, the '443 claims are directed to a new, improved multimedia file structure incorporating (1) partial frame encryption, (2) video content encoded as a plurality of video "chunks," (3) an index to navigate to video chunks within the file, and (4) a set of DRM chunks containing information to

identify and decrypt encrypted content, each having a corresponding video chunk. The '443 patent's new, improved file structure integrates three concepts: (1) partial frame encryption, (2) chunk-based adaptive bitrate streaming, and (3) a set of DRM chunks containing decryption information for a corresponding video chunk:

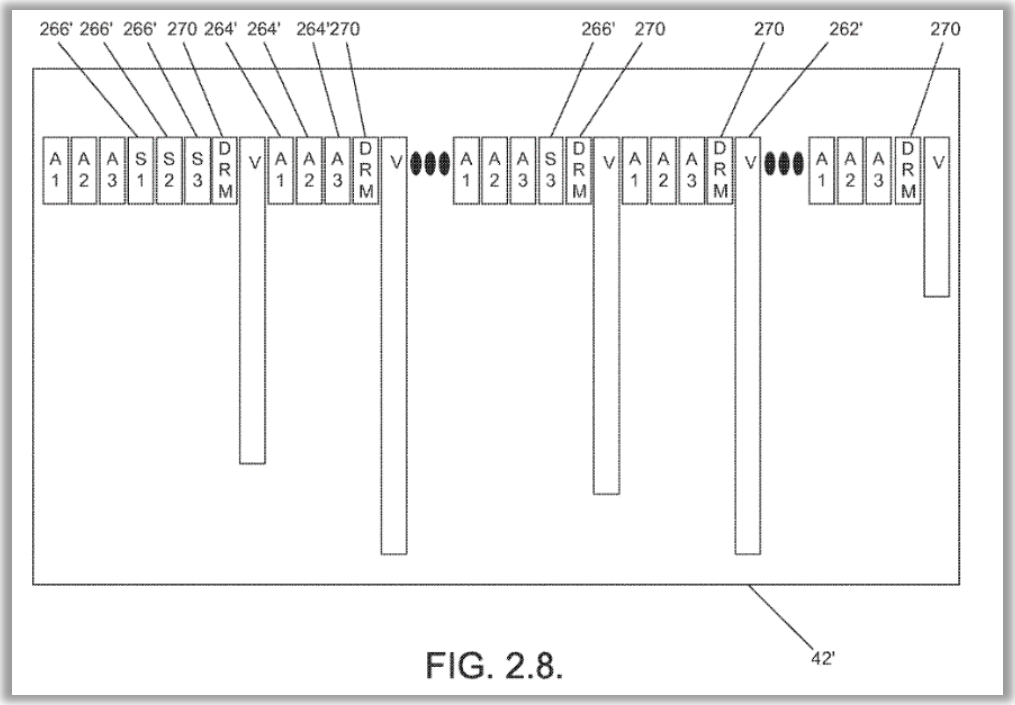


FIG. 2.8.

'443 patent, FIG. 2.8, 27:32-52.

60. The '443 inventions improve upon existing multimedia file formats, including, for example, RIFF and AVI file formats. The '443 inventions provide compatible extensions to those file formats with new structures of DRM and video data, including a set of DRM chunks and video chunks to facilitate efficient, chunk-based adaptive bitrate streaming and efficient decryption of video content. *See id.* at 12:56-13:28.

61. The '443 inventions' new, improved multimedia file structure improve the playback device's ability to navigate and play back the file's digital video content. The new, improved multimedia file structure improves the user experience by reducing delays in loading and playing a video, by making decryption more

efficient. The new, improved multimedia file structure and systems for encoding and decoding that file structure secure video content while reducing the computing resources needed to decrypt the video content. The '443 inventions enable Hulu to stream video to a diverse array of consumer devices while protecting the video content with secure encryption and decryption, allowing Hulu to both offer its service to a diverse device ecosystem and provide high-quality video content.

Technical Problems Addressed by the '443 Inventions

62. The '443 inventions address problems with the computing burdens and delays caused by securely protecting video content. The '443 patent's new, improved multimedia file structure addresses technical challenges with the computing resources required to play back compressed, encrypted digital video. A multimedia file's structure facilitates its delivery to and decryption, decoding, and playback on playback devices, including, for example, televisions, phones, tablets, and other consumer electronics devices. *See, e.g.*, *id.* at 12:56-14:2, FIG. 2.0. Before the '443 inventions, existing multimedia file formats could contain encrypted, compressed digital video content. A playback device must decrypt encrypted video content before that content can be played back on the device. The video content also must be decoded from its compressed format for playback. Decoding compressed digital video content on a playback device is computationally expensive; the greater the compression, the greater the processing power required to decode. Decrypting compressed digital video content requires additional processing resources, demanding even more processing power.

63. Existing multimedia file formats stored the file's decryption information in a single "chunk" within the file or in a separate file. The playback device would need to expend processing power to either (1) parse a large chunk containing decryption information for the entire file before decrypting any portion of that file, or (2) request, receive, and parse a separate file to obtain decryption information, all before performing any decryption or decoding. This additional

processing power either slows down the video processing and playback or requires additional, more expensive computing power in the playback device.

64. Both decryption and decoding are necessary to playing back encrypted digital video content on a playback device. Encryption provides security; compression enables transfer over the internet by minimizing the bandwidth required to transmit the video data in an acceptable time period. Enabling higher resolution of internet-delivered video content means a larger amount of data representing that video content. And a larger amount of video data means greater amount of encryption information needed to secure that video content. These increases in data place increased processing demands on playback devices to decrypt, decode, and play back the video. Processing demands require more complex, expensive decoders (e.g., consumer devices).

65. Accordingly, a need existed for an improved multimedia file structure to streamline the playback device's access to decryption information and decrease the processing power required to decrypt video content, all while maintaining security.

Technical Solutions and Benefits Provided by the '443 Inventions

66. The '443 patent claims specific, technical solutions to the technical challenges presented by existing multimedia file formats, specifically, by reciting a new, improved multimedia file structure incorporating partial frame encryption, video content encoded as video "chunks," an index to the video chunks within the file, and a set of DRM chunks each having a corresponding video chunk. The '443 claims are directed to improvements to the functionality of computer systems that receive, decrypt, and decode digital video content. The '443 claims are directed to a new, improved multimedia file structure, systems for decoding that file structure (claim 1 and dependents), and systems for encoding that file structure (claim 7 and dependents).

67. The new, improved multimedia file structure includes a set of DRM chunks, each containing decryption information for a corresponding video chunk within the file. *Id.* at 51:21-27 ("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."). Packaging a file's decryption information in a set of DRM chunks within the file allows the playback device to access decryption information for a particular portion of the video file as needed; the playback device does not need to process *all* of the file's decryption information or request, receive, and process a separate DRM file. In some claimed embodiments, the DRM chunks are interleaved with the video chunks. *Id.* at 51:29-37 (explaining that in some claimed embodiments, each DRM chunk precedes its corresponding video chunk). Interleaving the DRM chunks with the video chunks places relevant decryption information adjacent to the encrypted content, facilitating even more efficient decryption, decoding, and playback. *Id.* at 52:52-67, 53:22-27, FIG. 4.

68. The new, improved multimedia file structure also incorporates the benefits of partial frame encryption: less (i.e., more efficient) encryption and decryption plus secure video content. Further, the new, improved multimedia file structure includes DRM information that identifies the encrypted content. "In an encryption system in accordance with an embodiment of the present invention, the video chunks are only partially encrypted. 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." *Id.* at 27:53-58; *see also id.* at 27:63-28:6 and FIG. 2.9:

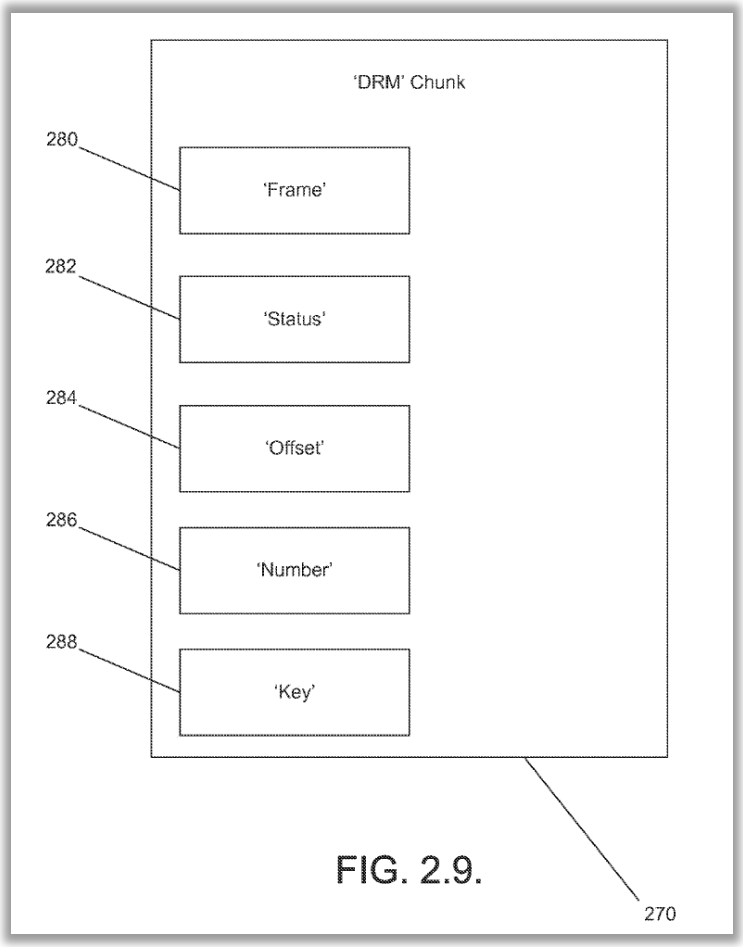


FIG. 2.9.

69. The new, improved multimedia file structure also incorporates an index to facilitate chunk-based access during playback. *Id.* at 28:7-17.

70. The '443 patent's new, improved multimedia file structure, and the systems for encoding and decoding that file structure, provide technical benefits that improve the functionality and capabilities of computer systems performing these operations. The file structure allows the playback device to efficiently identify, decrypt, and decode relevant video data for playback. By incorporating partial frame encryption, the new file structure reduces the computing resources required both for encoding and encrypting the data and for decrypting and decoding the data. Encrypting only a portion of an encrypted frame can further reduce the necessary computing resources. By encoding a set of DRM chunks each containing

decryption information relevant to a corresponding video chunk, the new file structure improves the performance of the computer system executing decryption and decoding operations, making decryption less computationally intensive. The playback device need only receive and parse the discrete DRM chunk relevant to the video chunk to be played back. And the DRM chunk includes information to identify the relevant encrypted content. By encoding the new file structure with an index to the video chunks within the file, the file enables chunk-based adaptive bitrate streaming and allowing the playback device to navigate the video file to locate and process only those chunks needed for playback. The new file structure was new and not well-known, routine, or conventional at the time of the '443 invention.

71. In sum, the inventions claimed in the '443 patent enable streaming of secure video content over the internet. The claims recite new computing techniques that improve the performance of computing systems (servers and playback devices, connected by networks) streaming video content over the internet and playing back that content. These improvements enhance the user experience for internet video delivery. The inventions achieve these benefits by providing a new kind of digital video file structure and new computing techniques for encoding and decoding that structure.

- For example, the inventions' new digital video file structure has video data encoded as individual portions, or "chunks" of video. The new digital video file structure also includes DRM information for each video chunk for easy access and streaming. The DRM information in the new file structure incorporates data used for decryption of partially encrypted frames of video. The new digital video file structure also includes an index for accessing and streaming individual video chunks and their associated DRM information protecting the video chunks.

- The inventions' new digital video file structure enables video streaming services to customize the security for their video content on a chunk-by-chunk basis. The new digital video file structure allows video streaming services to perform partial frame encryption on a chunk-by-chunk basis, which can enhance content security. The new digital video file structure also reduces the computing burdens imposed by the need to transfer DRM information to the playback device and process that information to decrypt encrypted video content. Reducing those computing burdens makes video playback faster and available on a wider range of computing devices.
- The inventions achieve the benefits presented above with new computing techniques for processing the new file structure to decode the file structure on a playback device and to encode the new file structure on an encoding computing system. For example, these new computing techniques enable playback devices to efficiently locate the information necessary to determine the specific portions of the frames of video that are encrypted, so that decryption can be performed based upon information contained within the file instead of based upon processing and analyzing the content of the frame to determine the portions that require decryption.

See, e.g., '443 patent at 1:26-35, 11:24-28, 11:60-63, 27:19-28:6, 51:21-38, 52:52-67, 53:22-27, FIG. 2.8, FIG. 2.9, FIG. 4.0.

Prosecution History of the '443 Inventions

72. The claims of the '443 patent issued at least because they recite a multimedia file incorporating partial frame encryption, a set of DRM chunks, and DRM information pointing to the encrypted data. The prior art cited during prosecution did not teach "a system for decoding multimedia files that receives at

least a portion of a multimedia file that includes 'at least one video track encoded as a plurality of video chunks, a set of digital rights management (DRM) chunks, and an index chunk' [wherein] 'each DRM chunk of the set of DRM chunks comprises DRM information to decrypt at least one partially encrypted frame of video' and 'the DRM information comprises an offset value that points to the start of an encrypted block within an encoded frame and a number value that indicates the number of encrypted bytes in the encrypted block.'" '443 File History,[19] Applicant Reply to Office Action of May 16, 2018 (Oct. 16, 2018), at 10-11. Specifically, the Candelore reference cited in prosecution did not "disclose partial encryption of encoded frames of video by encrypting at least one block of data within the partially encrypted frame so that only a portion of the encoded frame is encrypted." *Id.* at 11. Similarly, the Candelore reference did not "disclose partial encryption of encoded frames of video by encrypted at least one block of data within the partially encrypted frame so that only a portion of the frame is encrypted or DRM information that includes 'an offset value that points to the start of an encrypted block ***within an encoded frame***.'" *Id.* at 13. Further, the Candelore reference did not "disclose 'interleaving the video chunks and DRM chunks so that a DRM chunk for decrypting at least one partially encrypted frame within a particular video chunk is located before the particular video chunk.'" *Id.* The Notice of Allowance issued on November 16, 2018, without examiner comment. '443 File History, Notice of Allowance (Nov. 16, 2018).

73. During prosecution, the patent examiner did not reject any claims of the '443 patent for lack of subject matter eligibility under 35 U.S.C. § 101.

Claims Reciting the Technical Solutions of the '443 Inventions

74. Claim 1 of the '443 patent recites a system for decoding a new, improved multimedia file structure:

[19] Cited excerpts of the '443 file history attached as Exhibit 6.

1. A system for decoding multimedia files comprising:

at least one processor;

a non-volatile storage containing a decoder application;

wherein the decoder application causes the at least one processor to perform the steps of:

receiving at least a portion of a multimedia file, wherein:

the received at least a portion of the multimedia file comprises at least one video track encoded as a plurality of video chunks, a set of digital rights management (DRM) chunks, and an index chunk;

at least one video chunk of the plurality of video chunks contains at least one partially encrypted frame of video so that only a portion of the encoded frame is encrypted;

each DRM chunk of the set of DRM chunks comprises DRM information to decrypt at least one partially encrypted frame of video in at least one video chunk of the plurality of video chunks;

the DRM information comprises an offset value that points to the start of an encrypted block within an encoded frame and a number value that indicates the number of encrypted bytes in the encrypted block;

the index chunk includes information concerning the location of data chunks within the multimedia file including the locations of video chunks from the at least one video track; and

for each chunk of the plurality of video chunks:

determining whether the video chunk contains at least one partially encrypted frame of video;

> when a video chunk contains a partially encrypted frame of video, identifying a corresponding one of the set of DRM chunks that contains the DRM information for the partially encrypted frame of video, demultiplexing the partially encrypted frame from the video chunk, and decrypting the partially encrypted frame of video using the offset and number values from the DRM information for the partially encrypted frame of video; and
>
> decoding at least one encoded frame of video for display.

'443 patent, 55:51-56:25.

75. Claim 1 of the '443 patent therefore recites a system for decoding a new, improved multimedia file structure incorporating partial frame encryption, video content encoded as a plurality of video "chunks," an index to navigate to video chunks within the file, and a set of DRM chunks containing information to identify and decrypt encrypted content, each having a corresponding video chunk. *Id.*; *see also id.* at 1:26-35, 11:24-28, 11:60-63, 12:56-14:2, 27:19-28:17, 51:21-38, 52:52-67, 53:22-27, FIG. 2.8, FIG. 2.9, FIG. 4. Claim 1 and its dependents, therefore, recite limitations that enable the technical and performance benefits of the invention described above in ¶¶ 66-71. Claim 1 recites a novel, technical solution for more efficient decryption, decoding, playback, and chunk-based adaptive bitrate streaming.

76. Claims 2-6 and 15-16 of the '443 patent depend from claim 1, and each claim further describes the new, improved multimedia file structure. The additional elements in these claims, in conjunction with the elements of the claims from which each depends, therefore recite unconventional, new-and-improved multimedia file structures and systems for decoding them not well-known at the time of the '443 inventions.

77. Claim 7 of the '443 patent recites a system for encoding a new, improved multimedia file structure:

> 7. A system for encoding multimedia files comprising:
> a network interface;
> at least one processing unit;
> a non-transitory memory storing an encoding application, wherein the encoding application causes the at least one processing unit to encode multimedia files by performing the steps of:
> obtaining source media using the network interface, wherein the source media comprises video;
> encoding at least one video track as a plurality of video chunks, the video chunks being portions of the at least one video track, the video track comprising a series of encoded video frames;
> partially encrypting at least some of the encoded frames of video so that only portions of the encoded frames of video are encrypted;
> encoding the DRM information as a set of DRM chunks, wherein:
> each DRM chunk of the set of DRM chunks comprises DRM information to decrypt at least one partially encrypted frame of video in at least one video chunk of the plurality of video chunks; and
> the DRM information comprises an offset value that points to the start of an encrypted block within an encoded frame and a number value that indicates the number of encrypted bytes in the encrypted block;

> interleaving the video chunks and DRM chunks so that a DRM chunk for decrypting at least one partially encrypted frame within a particular video chunk is located before the particular video chunk; and
>
> encoding at least one index chunk that includes information concerning the locations of video chunks;
>
> writing the interleaved chunks to at least one multimedia file; and
>
> transmitting at least a portion of the at least one multimedia file using the network interface.

Id. at 56:41-57:10.

78. Claim 7 of the '443 patent therefore recites a system for encoding a new, improved multimedia file structure incorporating partial frame encryption, video content encoded as a plurality of video "chunks," an index to navigate to video chunks within the file, and a set of interleaved DRM chunks containing information to identify and decrypt encrypted content, each having and preceding a corresponding video chunk. *Id.*; *see also id.* at 1:26-35, 11:24-28, 11:60-63, 12:56-14:2, 27:19-28:17, 51:21-38, 52:52-67, 53:22-27, FIG. 2.8, FIG. 2.9, FIG. 4. Claim 7 and its dependents, therefore, recite limitations that enable the technical and performance benefits of the invention described above in ¶¶ 66-71. The limitation specifying that each DRM chunk precedes its corresponding video chunk creates increased efficiency during decryption and decoding of the file and during chunk-based adaptive bitrate streaming, because the playback system can use the index to request video chunks with their corresponding DRM chunks. Claim 7 recites a novel, technical solution for more efficient decryption, decoding, playback, and chunk-based adaptive bitrate streaming.

79. Claims 8-14 of the '443 patent depend from claim 7, and each claim further describes the new, improved multimedia file structure. The additional

elements in these claims, in conjunction with the elements of the claims from which each depends, therefore recite unconventional, new-and-improved multimedia file structure s and systems for encoding them not well-known at the time of the '443 inventions.

III. The '318 Patent

80. The '318 patent, entitled "Video Distribution System Including Progressive Playback," duly and legally issued on October 17, 2017, from a patent application filed on February 26, 2015, with Roland Osborne as the named inventor. The '318 patent claims priority to U.S. Provisional Application No. 60/883,659, filed on January 5, 2007.

Summary of the '318 Inventions

81. The inventions claimed in the '318 patent allow streaming video to start playing more quickly and allow viewers to use trick play functions right away. The '318 inventions achieve this goal by using the playback device, instead of the server, to request and receive a video file in smaller portions so the user has access to the received portions quickly, and by tracking which portions the playback device has requested and received to improve efficiency in downloading and playback.

82. Specifically, the '318 claims are directed to new, improved methods and systems for playing back remote digital video content as the playback device downloads the content, using HTTP requests to the server for portions or "chunks" of the video file. The inventions recited in the '318 claims improve playback of digital video content before the player receives the complete video file. The invention improves "playing multimedia files over a network and more specifically [] the progressive playback of multimedia files as they are downloaded over a network." '318 patent, 1:18-21. "Progressive playback" refers to playing remote video content as the player downloads the content, e.g., beginning to play video content before the player has received or downloaded the complete video file. *Id.* at

1:22-23. "With this feature a user can select a remote movie and commence watching it before it is fully downloaded. Even with a fast Internet connection, waiting for a movie to fully download can range from minutes to hours depending on the size of the media file. With progressive playback a user only has to wait a couple of seconds before playback can begin." *Id.* at 1:23-29.

83. The '318 inventions improve upon existing playback to enable streaming for longer video content (such as episode- or feature-length content) and to enable trick play functionality during playback without stalls or startup delay.

Technical Problems Addressed by the '318 Inventions

84. The '318 inventions address problems of slow start and lack of trick play functionality caused by inferior prior art systems. The '318 patent's new, improved playback methods and systems address multiple technical problems. Before the '318 inventions, existing digital video playback systems facilitated progressive playback of only short digital video clips because the systems downloaded remote video files *linearly*, from beginning to end. *Id.* at 1:30-45. Playback would begin only after the player had "buffered enough data to provide a likelihood that the media [would] play without interruption." *Id.* at 1:35-37. Because playback would begin only after the player had downloaded sufficient data, longer content would suffer from startup delay: "The buffering requirement can either be a fixed amount suitable for a large percentage of content, or a dynamic amount, where the player infers how much data is required to play the entire content without suffering buffer under-run." *Id.* at 1:37-41. Thus, existing systems did not support seeking, trick play (for example, pausing, rewinding, fast forwarding, skipping), or playing back longer content (e.g., feature-length movies). *Id.* at 1:30-45; *see also id.* at 1:58-2:8 ("When a long clip is started, it is impossible to seek or fast-forward to a point in the file that has not already been downloaded.").

85. Some existing streaming systems were "server-driven," as opposed to receiver-driven (e.g., based on instructions from the player). In server-driven systems, "the server parse[d] the data file and determine[d] which data to send" for playback. *Id.* at 1:46-57. Server-driven systems required custom computing systems, which increased expense: "[s]tandard HTTP web servers . . . do not typically provide this functionality, and custom web servers providing this functionality often scale poorly when called upon to deliver content simultaneously to a large number of players." *Id.* These systems required expensive, impractical, inefficient custom server designs unable to simultaneously supply digital video content to a large number of playback devices. *Id.*

86. Further, transmission of high quality video over the internet typically requires variable bitrate video, which means that different amounts of data can be used to encode a particular duration of the video. This variation in bitrate within a single video file makes it computationally difficult for the server or the playback device to determine the locations of the specific chunks of video corresponding to a particular playback time within the video file.

87. Accordingly, as demand for streaming digital video content increased, a need existed for new, improved, receiver-driven playback methods and systems able to perform efficient partial-download playback with trick play functionality (for example, requesting video content in non-linear or non-sequential chunks).

Technical Solutions and Benefits Provided by the '318 Inventions

88. The '318 patent claims specific, technical solutions to the technical challenges presented by existing playback systems, specifically, by reciting receiver-driven methods and systems in which the player receives an index to the remote video file to facilitate chunk-based streaming using HTTP requests for specific portions of the video file. *See, e.g.*, *id.* at 2:13-29. The system requests and queues chunks of the video file for download and playback based on instructions at the player. The '318 claims are directed to improvements to the functionality of

computers that request, receive, download, and play back digital video content stored in container files on a remote server. The '318 claims are directed to improved methods (claim 1 and dependents) and systems (claim 10 and dependents).

89. Some claims recite a technical solution related to the player's ability to deliver requests to the server for specific portions of a video file. "In several [claimed] embodiments, the ability to provide full featured progressive playback is due in part to the tight coupling of the playback engine for the media sequence (i.e., the system that decodes and plays back the encoded media) with a transport protocol that provides random access to the remote file. Interfacing of the playback engine with the transport protocol via a file parser can reduce latency and enable the client and media server to operate in parallel improving download efficiency and interactivity." *Id.* at 2:30-38. Further, the video "files are formatted to include an index to the data within the file and a transport protocol that allows for downloading specific byte ranges within a file." *Id.* at 2:38-42; *see also id.* at 6:10-34 ("When the media file includes an index, a device configured with a client application in accordance with an embodiment of the invention can use the index to determine the location of various portions of the media. Therefore, the index can be used to provide a user with 'trick play' functions. . . . [T]he client application requests portions of the media file using a transport protocol that allows for downloading of specific byte ranges within the media file.").

90. The '318 inventions provide improved playback methods and systems that enable a client application at the player to commence playing video content and to request non-sequential portions of the video file without receiving the complete video file. *Id.* at 5:18-39. The inventions create a client computing application capable of implementing progressive playback with trick play functionality. *Id.* Systems for downloading and playing back video over a network, before the '318 inventions, did not provide these benefits and advantages, and therefore these

improved methods and systems were new and not well-known, routine, or conventional at the time of the '318 inventions.

91. The '318 patent claims technical—not merely conceptual—solutions to recognized, but unsolved playback system shortcomings. The '318 inventions specify a client application including, for example, a download manager, a playback engine, and a file parser to facilitate progressive playback with trick play functionality. *Id.* at 7:1-37. One embodiment of the client application at the player includes a download manager "that is responsible for coordinating the downloading of specific byte ranges [or "chunks"] of a file from a remote server"; a playback engine "that coordinates the playback of a media file in response to user interactions"; and a file parser that "interfaces between the playback engine and the download manager" and "maps high level data requests from the playback engine to specific byte ranges that can then be requested using the download manager." *Id.* In short, the '318 invention provides a playback system able to navigate a remote digital video file, request portions or "chunks" of that video file, and download and play back those portions, while randomly accessing other portions of the file and queuing those chunks for playback, based on instructions from the system's user.

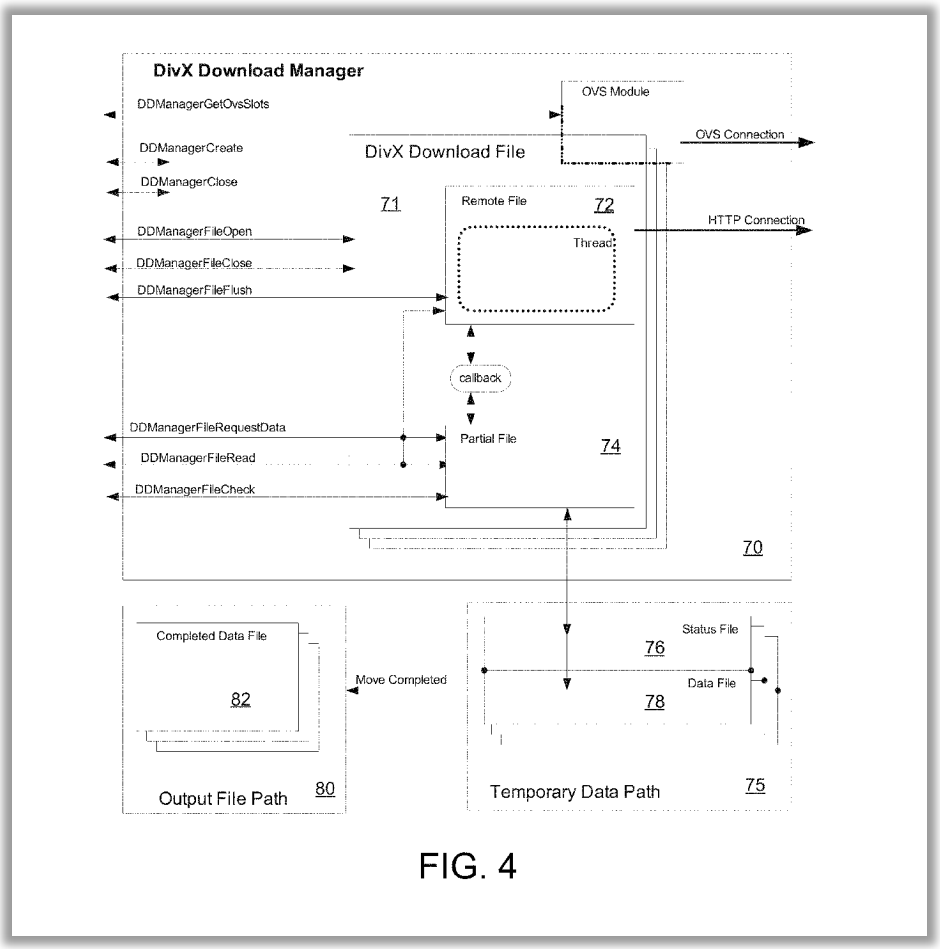


FIG. 4

'318 patent, Fig. 4.

92. The '318 inventions improve on existing playback methods and systems. "Instead of sequentially downloading a media file and waiting until the required information has been downloaded to perform a 'trick play' function, client applications in accordance with embodiments of the invention can determine portions of a media file that are required to support a specific 'trick play' function and request those portions of the file from the remote server. When a 'trick play' function involves skipping to portions of the media that have not been downloaded, such as fast forwarding and skipping between chapters, latency can be significantly reduced compared to sequential download." *Id*.

93. In sum, the inventions recited in the '318 patent improve the performance and user interactivity of video streaming over the internet. The claims

recite new computing techniques that improve the performance of computing systems (servers and playback devices, connected by networks) streaming video content over the internet and playing back that content. These improvements enhance the user experience for internet video streaming. The inventions also reduce the computing burdens on video streaming servers and economic costs associated with those burdens. For example, the inventions enable random seeking within an episode- or feature-length video upon beginning streaming. The inventions enable random seeking in higher-resolution videos streamed using variable bitrates. The inventions achieve these benefits with a new computer file structure containing multimedia data and new computing techniques for processing the new file structure on a playback device.

- For example, the inventions' new file structure includes media data organized in portions, or "chunks." The new file structure also includes an index to the media data chunks at certain locations within the file.
- The inventions achieve the benefits presented above with new computing techniques for processing the new file structure, including new techniques for sequencing the requesting and unpackaging of streamed multimedia data. For example, these new techniques include processing the index data to convert video "chunk" references to specific HTTP requests for use in communication with an HTTP-based web server hosting media data. These new techniques also include monitoring data requested and received from the server to make streaming more efficient and reduce computing resources needed at the playback device.
- The inventions' new file structure and new computing techniques to convert chunk references to HTTP requests enables video streaming services to reduce computing burdens on their media streaming

> servers and reduce associated costs. The cost reductions enabled by the inventions also allow video streaming services to scale their services to many different users and playback devices while mitigating computing burdens. The inventions' new computer file structure and new computing techniques also facilitate playback of episode- and feature-length video without stalls and enable random seeking upon beginning streaming.

See, e.g., '318 patent at 1:18-2:8, 2:12-57, 3:30-37, 4:1-9, 4:27-29, 4:53-64, 5:10-13, 5:18-39, 6:17-31, 6:59-7:37, 8:6-37, 9:29-47, 10:15-53, 10:57-65, 11:1-23, FIG, 2, FIG. 3, FIG. 4, FIG. 5, FIG. 9.

Prosecution History of the '318 Inventions

94. The claims of the '318 patent issued at least because they recite a receiver-driven, chunk-based playback method or system compatible with HTTP and facilitating trick play functionality. The prior art cited in prosecution did not teach "the processes of the method being performed 'using a playback engine' . . . . Indeed, the claimed invention specifically seeks to improve upon prior art systems similar to those described in [the prior art] by enabling simplification of server processing and instead relying upon playback devices to determine the appropriate responses to instructions." '318 File History,[20] Applicant Response to Office Action Dated November 3, 2016 (Feb. 3, 2017), at 7; *see also* Notice of Allowance (May 19, 2017), at 3 (allowing claims and noting that Applicant's arguments were "persuasive to overcome the current rejection" because "none of the currently applied prior art references, nor any other candidate prior art reference[,] expressly and sufficiently discloses the amended features of a 'playback engine implemented of [sic] a playback device' performing the steps of 'downloading an index from a remote media file . . . ', 'requesting chunks for downloading based upon a received

[20] Cited excerpts of the '318 file history attached as Exhibit 7.

instruction and maintaining a queue of the requested chunks . . . ', 'receiving a requested chunk and removing the chunk from the queue of requested chunks . . . ', 'maintaining a playback queue of the receiving chunks for playback pending commencement of playback by the playback device . . . ', and ' providing chunks maintained in the playback . . . queue to a decoder to enable playing of the media by the playback device.'").

95. During prosecution, the patent examiner did not reject any claims of the '318 patent for lack of subject matter eligibility under 35 U.S.C. § 101.

Claims Reciting the Technical Solutions of the '318 Inventions

96. Claim 1 of the '318 patent recites a new, improved method for chunk-based download and playback of remote digital video content supporting trick play functionality that delivers the technical benefits described in the '318 patent specification:

> 1. A method for obtaining media from a media file for playback on a playback device from a remote server, the media formatted to represent media as chunks, comprising:
>
> download an index from a remote media file, using a playback engine implemented on the playback device, wherein the index is used to convert a chunk reference into a specific HTTP request for use in downloading;
>
> request chunks for downloading based upon a received instruction and maintain a queue of the requested chunks, using the playback engine implemented on the playback device, wherein the requested chunks are downloaded by identifying media chunks corresponding to a sequence of key frames identified for the received instruction;

> receive a requested chunk and remove the received chunk from the queue of requested chunks, using the playback engine implemented on the playback device;
>
> maintain a playback queue of received chunks for playback pending commencement of playback by the playback device, using the playback engine implemented on the playback device; and
>
> provide chunks maintained in the playback queue to a decoder to enable playing of the media by the playback device, using the playback engine implemented on the playback device.

'318 patent, 11:43-67.

Claim 1 of the '318 patent therefore recites a new, improved, receiver-driven method for chunk-based download and playback with trick play functionality. *Id.*; *see also id.* at 1:18-2:8, 2:12-57, 3:30-37, 4:1-9, 4:27-29, 4:53-64, 5:10-13, 5:18-39, 6:17-34, 6:59-7:37, 8:6-37, 9:29-47, 10:15-53, 10:57-65, 11:1-23, FIG, 2, FIG. 3, FIG. 4, FIG. 5, FIG. 9. Claim 1 and its dependents, therefore, recite limitations that enable the technical and performance benefits of the invention described above in ¶¶ 88-93. Claim 1 recites a novel, technical solution for enabling download and playback of remote digital video content with trick play functionality using an index and HTTP requests to navigate the remote video file and request and queue content.

97. Claims 2-9 of the '318 patent depend from claim 1, and each claim further describes how the new, improved playback method facilitates efficient, chunk-based, partial-download playback with trick play functionality. The ordered combinations of elements in claims 2-9, in conjunction with the elements of the claims from which each claim depends, therefore recite unconventional, new-and-

improved digital video playback methods not well-known at the time of the '318 inventions.

98. Claim 10 of the '318 patent recites a playback device for implementing a new, improved method for chunk-based download and playback of remote digital video content supporting trick play functionality that delivers the technical benefits described in the '318 patent specification:

> 10. A playback device for streaming media content, the playback device comprising:
> at least one hardware processor that is configured to:
> download an index from a remote media file, wherein the index is used to convert a chunk reference into a specific HTTP request for use in downloading;
> request chunks for downloading based upon a received instruction and maintain a queue of the requested chunks, wherein the requested chunks are downloaded by identifying media chunks corresponding to a sequence of key frames identified for the received instruction;
> receive a requested chunk and remove the received chunk from the queue of requested chunks;
> maintain a playback queue of received chunks for playback pending commencement of playback by the playback device; and
> provide chunks maintained in the playback queue to a decoder to enable playing of the media by the playback device.

Id. at 12:25-45.

99. Claim 10 of the '318 patent therefore recites a playback device for implementing a new, improved, receiver-driven method for chunk-based download

and playback with trick play functionality. *Id.*; *see also id.* at 1:18-2:8, 2:12-57, 3:30-37, 4:1-9, 4:27-29, 4:53-64, 5:10-13, 5:18-39, 6:17-34, 6:59-7:37, 8:6-37, 9:29-47, 10:15-53, 10:57-65, 11:1-23, FIG, 2, FIG. 3, FIG. 4, FIG. 5, FIG. 9. Claim 10 and its dependents, therefore, recite limitations that enable the technical and performance benefits of the invention described above in ¶¶ 88-93. Claim 10 recites a novel, technical solution for enabling receiver-driven download and playback of remote digital video content with trick play functionality using an index and HTTP requests to navigate the remote video file and request and queue content.

100. Claims 11-18 of the '318 patent depend from claim 10, and each claim further describes how the new, improved playback method facilitates efficient, chunk-based, partial-download playback with trick play functionality. The ordered combinations of elements in claims 11-18, in conjunction with the elements of the claims from which each depends, therefore recite unconventional, new-and-improved digital video playback systems not well-known at the time of the '318 inventions.

IV. The '141 Patent

101. The '141 patent, entitled "Systems and Methods for Seeking Within Multimedia Content During Streaming Playback," duly and legally issued on September 10, 2019, with Roland Osborne as the named inventor. The '141 patent claims priority to U.S. Provisional Application No. 60/883,659, filed on January 5, 2007.

Summary of the '141 Invention

102. The inventions claimed in the '141 patent allow streaming video of multiple title tracks, audio tracks, and subtitle tracks to start playing more quickly and allow viewers to use trick play functions right away. The '141 inventions achieve this goal by using metadata describing a multimedia file to determine and request smaller portions of the multimedia file so the user has access to the

requested portions quickly, and commencing playback or trick play once the multimedia data in the buffer reaches a threshold sufficient to avoid playback stalls.

103. The '141 claims are directed to new, improved methods and systems for playing back digital video content over the internet as the playback device downloads the content. The '141 inventions enable user-driven or receiver-driven playback of digital video content selected from multiple tracks of media data. A user-driven or receiver-driven approach to playback occurs when the device, based on user instruction, determines the data that is requested and downloaded and parses the files to determine when to request and download additional data. *See* '141 patent, 1:40-55, 2:1-19.

104. The inventions improve "playing multimedia files over a network and more specifically [] the progressive playback of multimedia files as they are downloaded over a network." '141 patent, 1:28-31. The '141 inventions improve trick play functions for progressive playback of partially downloaded media files, that can contain one or more video tracks, one or more audio tracks and/or one or more subtitle tracks. '141 patent, 2:23-39. "Progressive playback" refers to playing remote video content as the player downloads the content, e.g., beginning to play video content before the player has received or downloaded the complete video file. *Id.* at 1:32-35. "With this feature a user can select a remote movie and commence watching it before it is fully downloaded. Even with a fast Internet connection, waiting for a movie to fully download can range from minutes to hours depending on the size of the media file. With progressive playback a user only has to wait a couple of seconds before playback can begin." *Id.* at 1:23-29. "Trick play" functions include performing playback functions "such as rewinding, fast forwarding and skipping between scenes that require non-sequential access of media content." *Id.* at 2:10-13. The invention enables trick play functions to occur without requiring the sequential downloading and processing of either an entire

media file or parts of media files required to execute a trick play instruction, while minimizing system latency. *Id.* at 5:39-45.

105. The '141 inventions improve upon existing playback systems to enable streaming for longer video content (such as episode- or feature-length content) and more complex media libraries, and to enable playback systems to provide seeking functionality during progressive playback of multi-track media files without stalls or startup delay.

Technical Problems Addressed by the '141 Inventions

106. The '141 inventions address these problems caused by inferior prior art systems: startup delay, streaming options limited to single track files, audio files, and/or files without subtitles, and lack of or limited trick play functionality. The '141 patent's new playback methods and systems address multiple technical problems. Existing digital video playback systems facilitated progressive playback for only short video clips because the systems downloaded video files *linearly*, from beginning to end. *Id.* at 1:44-45. Playback would begin only after the player had "buffered enough data to provide a likelihood that the media [would] play without interruption." *Id.* at 1:45-47. Because playback would begin only after the player had downloaded sufficient data, longer content would suffer from startup delay: "The buffering requirement can either be a fixed amount suitable for a large percentage of content, or a dynamic amount, where the player infers how much data is required to play the entire content without suffering buffer under-run." *Id.* at 1:47-51. Thus, existing systems did not support random seeking, trick play (for example, pausing, rewinding, fast forwarding, skipping), or playing back longer content (i.e., feature-length movies), and was not suitable for use with internet servers that "store files that can contain multiple titles, titles that include multiple audio tracks, and/or titles that include one or more subtitle tracks." *Id.* at 1:50-55, 2:34-39; *see also id.* at 2:3-10 ("When a long clip is started, it is impossible to seek or fast-forward to a point in the file that has not already been downloaded."). Multi-

track media, in particular, was not suitable for the existing smooth trick play functionality as the playback device must download the data for the other tracks, even if only certain tracks have been chosen for playback. *See* 10:46-11:12. Such systems were likely to suffer from buffer under-run when receiving trick play instructions, resulting in playback stalls and startup delays caused by access delays in data transmission and computing burdens placed on the network and device.

107. Some existing streaming systems were "server-driven," as opposed to receiver-driven (e.g., based on instructions from the player). In server-driven systems, "the server parse[d] the data file and determine[d] which data to send" for playback. *Id.* at 1:66-67. Server-driven systems required custom computing systems, which increased expense: "[s]tandard HTTP web servers . . . do not typically provide this functionality, and custom web servers providing this functionality often scale poorly when called upon to deliver content simultaneously to a large number of players." *Id.* These systems required expensive, impractical, inefficient custom server designs unable to simultaneously supply digital video content to a large number of playback devices. *Id.*

108. Accordingly, as demand for streaming digital video content increased, a need existed for a new, improved playback implementation able to facilitate (1) efficient non-linear partial-download playback with trick play functionality, (2) *receiver*-driven partial-download playback compatible with HTTP, and (3) delivery of video streaming to a large number of devices.

Technical Solutions and Benefits Provided by the '141 Inventions

109. The '141 patent claims specific, technical solutions to the technical challenges presented by existing playback systems, specifically, by enabling the playback device to support playback of multiple audio and subtitle tracks without downloading them all. *See, e.g., id.* at 2:23-39. The system selects video, audio, and/or subtitle tracks among other tracks in the file and requests specific portions of the selected tracks for download, buffering, and playback based on instructions at

the playback device. The '141 claims are directed to improvements to the functionality of computers that request, receive, download, buffer, and play back digital video, audio, and subtitle content stored in container files on a remote server. The '141 claims are directed to improved devices (claim 1 and dependents, and claim 12 and dependents) and methods (claim 20 and dependents).

110. Some claims recite a technical solution related to the player's ability to deliver requests to the server for specific portions of a video file. For example, "[i]n several embodiments, the ability to provide full featured progressive playback is due in part to the tight coupling of the playback engine for the media sequence (i.e., the system that decodes and plays back the encoded media) with a transport protocol that provides random access to the remote file. Interfacing of the playback engine with the transport protocol via a file parser can reduce latency and enable the client and media server to operate in parallel improving download efficiency and interactivity." *Id.* at 2:40-48. Further, the multi-track media "files are formatted to include an index to the data within the file and a transport protocol that allows for downloading specific byte ranges within a file." *Id.* at 2:48-52; *see also id.* at 6:20-44 ("When the media file includes an index, a device configured with a client application in accordance with an embodiment of the invention can use the index to determine the location of various portions of the media. Therefore, the index can be used to provide a user with 'trick play' functions. . . . [T]he client application requests portions of the media file using a transport protocol that allows for downloading of specific byte ranges within the media file.").

111. The '141 inventions provide an improved playback implementation that enables a client application at the player to commence playing video content and to request non-sequential portions of the video file without receiving the complete video file. *Id.* at 5:28-49. The inventions create a client computing application capable of implementing progressive playback and supporting trick play functionality for files containing multiple titles and for titles with multiple media

tracks. *Id.* at 5:28-49, 2:23-39. This implementation was new and not well-known, routine, or conventional at the time of the '141 inventions.

112. The '141 patent claims technical—not merely conceptual—solutions to recognized, but unsolved progressive playback shortcomings. The '141 inventions specify a client application with multiple "abstraction layers" to facilitate progressive playback with trick play functionality. *Id.* at 7:13-34. One exemplary embodiment of the player claimed by the '141 patent includes a download manager "that is responsible for coordinating the downloading of specific byte ranges of a file from a remote server"; a playback engine "that coordinates the playback of a media file in response to user interactions"; and a file parser that "interfaces between the playback engine and the download manager" and "maps high level data requests from the playback engine to specific byte ranges that can then be requested using the download manager." *Id.*

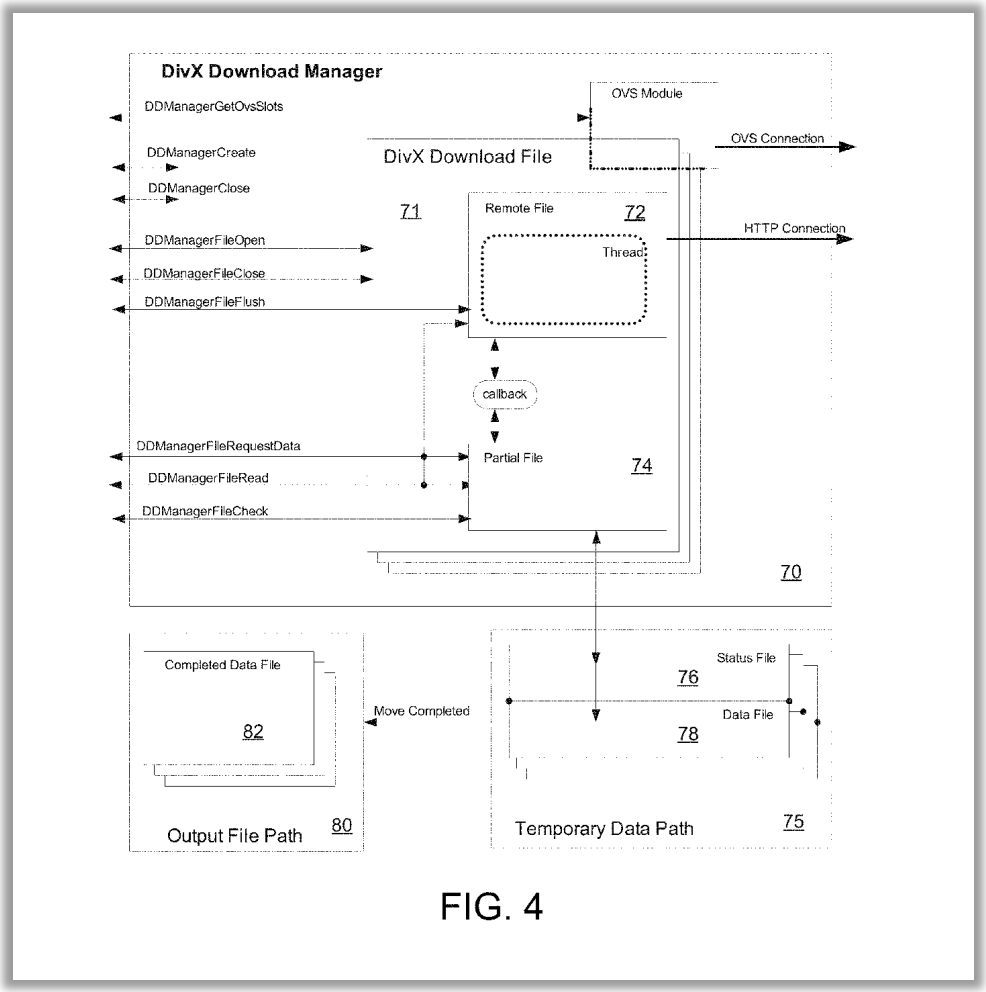


FIG. 4

'141 patent, Fig. 4.

113. The '141 inventions improve on existing playback systems by enabling trick play functionality for content that has yet to be downloaded. "Instead of sequentially downloading a media file and waiting until the required information has been downloaded to perform a 'trick play' function, client applications in accordance with embodiments of the invention can determine portions of a media file that are required to support a specific 'trick play' function and request those portions of the file from the remote server. When a 'trick play' function involves skipping to portions of the media that have not been downloaded, such as fast forwarding and skipping between chapters, latency can be significantly reduced compared to sequential download." *Id.*

114. The claims of the '141 patent enable trick play function for downloaded content by requiring the playback device to request and buffer sufficient data from the new playback location. The '141 patent claims recite an index, from which the playback engine can "refer to the index to determine the media information required to play the media file in the manner requested by the user," including by using a file parser and download manager to extract the necessary information from the index. *Id*. at 10:46-61. The playback engine can then select portions of the media file "based upon instructions, including 'trick play' instructions, received from the user and provide[] instructions to the file parser to download [] the selected" portions. *Id*. at 10:46-67. "When the playback engine receives the chunks from the file parser, the chunks are queued and provided to an appropriate decoder to enable the playing [] of the media. Playback of the movie can begin once enough of the movie has been downloaded. The buffered length can be determined by the length of the playback list shared with the" download component. *Id*. at 11:3-12.

115. In short, the '141 inventions provide a playback system able to navigate a remote multi-track digital video file, select specific video, audio, and/or subtitle tracks for download, and request specific portions of the selected tracks to download and play back, enabling random access to other portions of the file for download, buffering, decryption, and playback, based on instructions from the system's user.

116. In sum, the inventions claimed in the '141 patent improve performance and user interactivity of video streaming over the internet. The claims recite new computing techniques that improve the performance of computing systems (servers and playback devices, connected by networks) streaming video content over the internet and playing back that content. These improvements enhance the user experience for internet video streaming. The inventions also reduce the computing burdens on video streaming servers and economic costs associated with those

burdens. The inventions achieve these benefits with new computing techniques for processing computer files containing multimedia data on a playback device.

- For example, the inventions' new computing techniques enable random seeking within an episode- or feature-length video upon beginning streaming. The new computing techniques enable random seeking in higher-resolution videos streamed using variable bitrates. The file structure processed using the new computing techniques include video data, audio data, and subtitle data. The file structure also includes an index to the video and audio data that references byte ranges corresponding to locations within the video data and audio data in the file. Playback computing devices can employ the inventions' new computing techniques to use the index data to implement seeking and stream selection functionality within an HTTP streaming environment in a way that allows the playback device to play back the multimedia content without having to download all streams in their entirety.
- The inventions' new index referencing byte ranges enables video streaming services to reduce computing burdens on their media streaming servers and reduce associated costs. The cost reductions enabled by the inventions also allow video streaming services to scale their services to many different users and playback devices while mitigating computing burdens.
- The inventions' new computing techniques also facilitate playback of episode- and feature-length video without stalls. For example, the new computing techniques include measuring video data stored in a buffer on the computing device to determine when to begin playing back video upon startup or after a seek instruction from the user. The new computing techniques also include interpreting the new

index to convert user seek commands to requests to the streaming server for specific video portions. The inventions' new computing techniques enable video streaming services to offer an improved user experience that mimics the interactivity and quality of a DVD. *See, e.g.*, '141 patent at 1:32-2:19, 2:23-67, 4:11-19, 4:63-5:7, 5:20-23, 5:28-49, 7:4-34, 9:41-59, 9:67-10:4, 10:27-40, 10:46-65, 11:3-12, 11:21-34, FIG. 2, FIG. 3, FIG. 4, FIG. 5, FIG. 9.

Prosecution History of the '141 Inventions

117. The claims of the '141 patent issued at least because they recite a receiver- or player-driven playback implementation compatible with HTTP to facilitate trick play functionality. The prior art cited in prosecution did not teach "the recited features/elements" of the invention "*as a whole*." '141 File History[21], Notice of Allowance (April 19, 2019) (emphasis original). Specifically, "none of the applied prior art references . . . or additional candidate prior art references, expressly / sufficiently" disclosed a client or playback device that performed the steps of "obtaining index information indicating the locations of audio and video data within the selected audio and video tracks[,] determining bye ranges to request from the selected audio and video tracks using the index information[,] requesting byte ranges from the selected video track and the selected audio track from the remote server system[,] buffering received byes of information comprising audio and video data[,] checking that sufficient data is buffered to commence playback and playing back the buffered audio and video data[,] responding to a received seek instruction by[,] pausing playback[,] determining byte ranges to request from the selected audio and video tracks based upon a new playback location using the index information[,] and requesting byte ranges required to play the media sequence from

[21] Cited excerpts of the '141 file history attached as Exhibit 8.

the new playback location from the remote server." '141 File History[22], Notice of Allowance (April 19, 2019), at 2-3.

118. During prosecution, the patent examiner did not reject any claims of the '141 patent for lack of subject matter eligibility under 35 U.S.C. § 101.

Claims Reciting the Technical Solutions of the '141 Inventions

119. Claim 1 of the '141 patent recites a playback device for implementing the improved non-sequential playback and trick play implementation for multi-track media files that delivers the technical benefits described in the '141 patent specification:

> 1. A playback device, comprising:
> a processor; and
> a non-volatile storage containing an application for causing the processor to perform the steps of:
> establishing at least one connection for communicating with a remote server system;
> obtaining information from a remote server system describing at least one video track, multiple audio tracks, and multiple subtitle tracks;
> selecting a video track from the at least one video track;
> requesting a header describing the selected video track;
> selecting an audio track from the multiple audio tracks;
> obtaining index information indicating the locations of audio and video data within the selected audio and video tracks;
> determining byte ranges to request from the selected audio and video tracks using the index information;

[22] Cited excerpts of the '141 file history attached as Exhibit 8.

requesting byte ranges from the selected video track and the selected audio track from the remote server system;

buffering received bytes of information comprising audio and video data;

checking that sufficient data is buffered to commence playback and playing back the buffered audio and video data;

responding to a received seek instruction by:

pausing playback;

determining byte ranges to request from the selected audio and video tracks based upon a new playback location using the index information;

requesting byte ranges required to play the selected audio and video tracks from the new playback location from the remote server;

buffering received bytes of information comprising audio and video data pending resumption of playback; and

checking that sufficient data is buffered to commence playback and playing back the buffered audio and video data.

'141 patent, at 11:53-12:24.

120. Claim 1 of the '141 patent therefore recites a playback device implementing a receiver-based playback device supporting playback of multiple audio and subtitle tracks without downloading them all. *Id*.; *see also id*. at 1:28-2:19, 2:23-67, 4:11-19, 4:63-5:7, 5:20-23, 5:28-49, 7:4-34, 9:41-59, 9:67-10:4, 10:27-40, 10:46-11:12, 11:21-34, FIG. 2, FIG. 3, FIG. 4, FIG. 5, FIG. 9. Claim 1 and its dependents, therefore, recite limitations that enable the technical and performance benefits of the invention described above in ¶¶ 109-116. Claim 1 recites a novel solution for enabling non-sequential playback with trick play

functionality for multi-track media files using an index to identify the locations of audio and video data within selected media tracks and to determine the byte ranges to request from said tracks from a remote server system, buffering the received bytes of information, and checking that sufficient data is buffered to commence playback of the specifically requested data.

121. Claims 2-11 of the '141 patent depend from claim 1, and each claim further describes how the new, improved playback implementation facilitates efficient, non-sequential partial-download playback with trick play functionality for multi-track media files. The ordered combination of elements in each of claims 2-11, in conjunction with the elements of the claims from which they depend, therefore recite unconventional new and improved digital playback systems that were not well-known at the time of the '141 inventions.

122. Claim 12 of the '141 patent recites a playback device for implementing the improved playback and trick play implementation for multi-track media files that delivers the technical benefits described in the '141 patent specification:

> 12. A playback device, comprising:
> a processor; and
> a non-volatile storage containing an application for
> causing the processor to perform the steps of:
> establishing at least one connection for communicating
> with a remote server system;
> obtaining information from a remote server system
> describing at least one video track, and at least one audio
> track;
> selecting a video track from the at least one video track;
> requesting a header describing the selected video track,
> where the requested header comprises a DRM header;

decrypting the DRM header;

selecting an audio track from the at least one audio track;

obtaining index information indicating the locations of audio and video data within the selected audio and video tracks;

determining byte ranges to request from the selected audio and video tracks using the index information;

creating a buffer;

requesting byte ranges from the video track and the audio track from the remote server system;

buffering received bytes of information comprising audio and video data;

checking that sufficient data is buffered to commence playback;

decrypting encrypted frames of video using information from the decrypted DRM header;

playing back the buffered audio and the decrypted video data;

responding to a received seek instruction by:

pausing playback;

discarding buffered audio and video data;

determining byte ranges to request from the selected audio and video tracks based upon a new playback location using the index information;

requesting byte ranges required to play the selected audio and video tracks from the new playback location from the remote server;

buffering received bytes of information comprising audio

and video data pending resumption of playback;

checking that sufficient data is buffered to commence playback;

decrypting encrypted frames of video using information from the decrypted DRM header; and

playing back the buffered audio and decrypted video data.

Id. at 13:5-55.

123. Claim 12 of the '141 patent therefore recites a playback device implementing a receiver-based playback device supporting playback of multiple audio and subtitle tracks without downloading them all. *Id.*; *see also id*. at 1:28-2:19, 2:23-67, 4:11-19, 4:63-5:7, 5:20-23, 5:28-49, 7:4-34, 9:41-59, 9:67-10:4, 10:27-40, 10:46-11:12, 11:21-34, FIG. 2, FIG. 3, FIG. 4, FIG. 5, FIG. 9. Claim 12 and its dependents, therefore, recite limitations that enable the technical and performance benefits of the invention described above in ¶¶ 109-116. Claim 12 recites a novel solution for enabling non-sequential progressing playback with trick play functionality for multi-track media files using an index to identify the locations of audio and video data within selected media tracks and to determine the byte ranges to request from said tracks from a remote server system, buffering the received bytes of information, and checking that sufficient data is buffered to commence playback of the specifically requested data, and decrypting the requested data using a decrypted DRM header for playback. *Id.* at 13:5-55.

124. Claims 13-19 of the '141 patent depend from claim 12, and each claim further describes how the new, improved playback implementation facilitates efficient, non-sequential partial-download decryption and playback, with trick play functionality, for multi-track media files. The ordered combination of elements in each of claims 13-19, in conjunction with the elements of the claims from which they depend, therefore recite unconventional new and improved digital playback systems that were not well-known at the time of the '141 inventions.

125. Claim 20 of the '141 patent recites a method for practicing the improved non-sequential playback implementation with trick play functionality for multi-track media files that delivers the technical benefits described in the '141 patent specification:

> 20. A method of playing back content on a playback device, comprising:
> establishing at least one connection for communicating with a remote server system using a playback device;
> 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;
> selecting a video track from the at least one video track;
> requesting a header describing the at least one video track using the playback device;
> selecting an audio track from the multiple audio tracks using the playback device;
> obtaining index information indicating the locations of audio and video data within the selected audio and video tracks;
> determining byte ranges to request from the selected audio and video tracks using the index information;
> requesting byte ranges from the selected video track and the selected audio track from the remote server system using the playback device;
> buffering received bytes of information comprising audio and video data on the playback device;

checking that sufficient data is buffered to commence playback and playing back the buffered audio and video data using the playback device;

responding to receipt of a seek instruction at the playback device by:

pausing playback on the playback device;

determining byte ranges to request from the selected audio and video tracks based upon a new playback location using the index information;

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;

buffering received bytes of information comprising audio and video data pending resumption of playback using the playback device; and

checking that sufficient data is buffered to commence playback and playing back the buffered audio and video data using the playback device.

Id. at 14:31-15:6.

126. Claim 20 of the '141 patent therefore recites a method for implementing a receiver-based playback device with trick play functionality. *Id*. ; *see also id*. at 1:28-2:19, 2:23-67, 4:11-19, 4:63-5:7, 5:20-23, 5:28-49, 7:4-34, 9:41-59, 9:67-10:4, 10:27-40, 10:46-11:12, 11:21-34, FIG. 2, FIG. 3, FIG. 4, FIG. 5, FIG. 9. Claim 20 and its dependents, therefore, recite limitations that enable the technical and performance benefits of the invention described above in ¶¶ 109-116. Claim 20 recites a novel solution for enabling non-sequential playback with trick play functionality for multi-track media files using an index to identify the locations of audio and video data within selected media tracks and to determine the byte

ranges to request from said tracks from a remote server system, buffering the received bytes of information, and checking that sufficient data is buffered to commence playback of the specifically requested data.

127. Claims 21-30 of the '141 patent depend from claim 20, and each claim further describes how the new, improved playback implementation facilitates efficient, non-sequential partial-download decryption and playback, with trick play functionality, for multi-track media files. The ordered combination of elements in each of claims 21-30, in conjunction with the elements of the claims from which they depend, therefore recite unconventional new and improved digital playback systems that were not well-known at the time of the '141 inventions.

V. The '061 Patent

128. The '061 patent, entitled "Systems and Methods for Automatically Generating Top Level Index Files," was duly and legally issued on January 21, 2020, from a patent application filed December 3, 2018 with Jason Braness, Evan Wallin, and Ederson Ferreira as the named inventors. The '061 patent claims priority to U.S. Provisional Application No. 61/529,403, filed on August 30, 2011.

Summary of the '061 Inventions

129. The inventions claimed in the '061 patent select video streams based on device capability and video assets and streamline decryption of protected video streams, which provide optimal video quality while maintaining playback startup speed. The '061 inventions achieve this by selecting video streams based on playback device capabilities and encrypting video streams using common cryptographic information so that information does not need to be repeatedly downloaded.

130. Specifically, the '061 claims are directed to a new adaptive bitrate streaming process for protected video content using automatically generated top level index files. According to the '061 invention, "in several embodiments, the process of generating the top level index file involves determining all of the assets

or container files containing streams associated with a specific piece of content and then filtering the assets based upon one or more predetermined criterion." '061 patent, 7:2-7.

131. The '061 inventions are directed to automatically generating a top level index file that references common cryptographic information and providing the top level index file and the encrypted common cryptographic information to the playback device, thereby reducing startup delay. '061 patent, 8:51-60, 18:56-60, 18:66-19:2. The inventions further include "gather[ing] information concerning the assets associated with a specific piece of content or title" and "apply[ing] one or more filters to the list of available assets to produce a list of assets that satisfies criteria including (but not limited to) criteria with respect to the capabilities of the playback device, the preferences of the user, and/or the requirements of the content owner," taking advantage of a full range of device capabilities on the market. '061 patent, 13:1-10. "When the assets associated with a piece of content have been retrieved, the playback server can filter the assets to exclude assets that are not capable of being played back by the playback device, not permitted to be played back, or are not desired to be played back." '061 patent, 13:18-22. Further, "[t]he playback device can then request portions of container files containing streams of content from one or more servers within a content delivery network 14 and can use the [common] cryptographic information to access protected content." '061 patent, 8:45-52.

132. The inventions claimed in the '061 patent enable Hulu to securely deliver videos based on optimal device capability and video assets, allowing Hulu to provide a high-quality user experience during streaming of Hulu videos.

Technical Problems Addressed by the '061 Inventions

133. The '061 inventions address problems presented by existing secure video streaming technology, which required more data and slowed down the start of playback. Downloading cryptographic information for protected videos consumes

additional startup bandwidth in adaptive bitrate streaming. At startup, adaptive bitrate streaming requires the playback device to "request[] portions of media from an initial set of streams," "download[] the requested media," "measure the available bandwidth," and "switch to higher or lower bitrate streams." '061 patent, 1:51-61. Further, "[s]ome or all of the assets associated with a specific piece of content may be encrypted or technically protected. Digital Rights Management (DRM) systems enable the communication of cryptographic information to playback devices so that the playback device can access protected streams in the clear (i.e. in an unencrypted form)." '061 patent, 18:34-39. The additional communications needed to request and obtain cryptographic information to protect video content with DRM requires additional bandwidth at startup.

134. Typical top level index files used during adaptive bitrate streaming only took into account a limited range of device capabilities (e.g., a limited number of device display resolution capabilities) to avoid burdening the adaptive bitrate streaming system. The video content "is typically stored on a media server as a top level index file pointing to a number of alternate streams that contain the actual video and audio data." '061 patent, 2:6-9. The top level index file "describes the location and content of container files containing" the alternative streams and "can be utilized by the playback device to stream and playback content." '061 patent, 6:61-65. The top level index accounts for device capabilities such as "the playback device's network bandwidth and video decoding capacity" and allows the streaming system to adjust the quality of the streamed media accordingly. '061 patent, 1:47-51. As the number of supported devices grows, so does the range of device capabilities, and storing an individual index on the server for every device imposes significant computing burdens and cost on the video streaming server system.

135. The '061 patent, therefore, addresses technical problems: allowing adequate content security while maintaining startup speed and tailoring top level index files to a wider range of device capabilities in an adaptive bitrate streaming

system while reducing the computing burdens on video streaming servers. *See, e.g.*, '061 patent, 18:34-39, 12:52-54.

Technical Solutions and Benefits Provided by the '061 Inventions

136. The '061 patent claims specific ways to address these technical challenges by automatically generating top level index files in response to a request for protected video streams from the playback device. The improved top level index files of the '061 patent maintain startup speed and allow a video streaming system to scale to optimize the user experience for a wide range of device capabilities on the market. Compared to the top level index files in prior systems, the '061 inventions allow the top level index files to take into account more assets or streams and provide a better user experience, including:

- video streams created for different classes of playback device;
- lower bitrate streams created for devices that will stream content over cellular data networks;
- higher bitrate streams created for devices that will stream content over a home network connected to the Internet via a high speed Internet connection;
- video streams created with different aspect ratios and different audio streams can be created for different languages;
- premium high resolution content that is only accessible to a playback device that has purchased high resolution content; and
- streams for different geographic regions.

'061 patent, 12:52-13:10.

137. The '061 claims are directed to a new adaptive bitrate streaming process for protected video content using automatically generated top level index files (claim 1 and dependents) and an improved server computer system configured to perform the new adaptive bitrate streaming process for protected video content using automatically generated top level index files (claim 14 and dependents).

138. The new adaptive bitrate streaming process in the '061 inventions includes creating a top level index file that references common cryptographic information and providing the top level index file and the encrypted common cryptographic information to the playback device. '061 patent, 8:51-60, 18:56-60, 18:66-19:2. The cryptographic information is encrypted "so that it can be accessed using the playback device's device cryptographic information." '061 patent, 18:66-19:2. "The playback device can then request portions of container files containing streams of content from one or more servers within a content delivery network 14 and can use the [common] cryptographic information to access protected content," thereby reducing startup delay caused by DRM. '061 patent, 8:45-52.

139. The new adaptive bitrate streaming process in the '061 inventions also includes "gather[ing] information concerning the assets associated with a specific piece of content or title" and "apply[ing] one or more filters to the list of available assets to produce a list of assets that satisfies criteria including (but not limited to) criteria with respect to the capabilities of the playback device, the preferences of the user, and/or the requirements of the content owner." '061 patent, 13:1-10. "When the assets associated with a piece of content have been retrieved, the playback server can filter (98) the assets to exclude assets that are not capable of being played back by the playback device, not permitted to be played back, or are not desired to be played back." '061 patent, 13:18-22. The new adaptive bitrate streaming process described in the '061 invention was new and not well-known, routine, or conventional at the time of the '061 patent.

140. The '061 inventions are directed to improvements to the functionality of computer systems that perform digital video streaming, including:

- streaming system architecture, '061 patent, FIG. 1, 7:38-9:26;
- playback devices, *id.*, FIG. 2, 9:27-10:20;
- content playback, *id.*, FIG. 3, 10:21-11:23;
- automatic generation of top level indexes, *id.*, FIG. 4, 11:24-12:50;

- filtering assets for inclusion in top level index files, *id.*, FIG. 5, 12:51-13:67;
- generating a top level index SMIL file, *id.*, FIG. 6, 14:1-18:32;
- protected streams, *id.*, FIG. 7, 18:33-19:19;
- play event reports, *id.*, FIG. 8, 19:20-20:5; and
- communicating during adaptive streaming, *id.*, FIG. 9, 20:6-41.

141. The system configured to perform the new adaptive bitrate streaming process in the '061 inventions also was new and not well-known, routine, or conventional at the time of the '061 patent.

142. The new adaptive bitrate streaming process for protected video content using automatically generated top level index files as described in the '061 claims, and system configured to perform the new process, provide technical benefits that improve the functionality and capabilities of computer systems performing these operations. *See, e.g.*, *id.* at 12:51-13:67, FIG. 5 (describing improvement to top level index files that allow a broad range of assets to be considered), 18:33-19:19, FIG. 7 (describing improvement to top level index files that reference common cryptographic information). By incorporating common cryptographic information in the top level index files and providing the index file and the encrypted common cryptographic information to the playback device, the playback device can request portions of container files and use the common cryptographic information to access protected content, thereby reducing startup delay caused by DRM. '061 patent, 8:45-52. By incorporating more assets in the top level index files compared to in prior systems, the streaming system can choose the optimal video stream for a wide variety of playback devices.

143. In sum, the inventions claimed in the '061 patent improve adaptive bitrate streaming. The claims recite new computing techniques that improve the performance of computing systems (servers and playback devices, connected by networks) performing ABS. These improvements enhance the user experience for

ABS. The inventions achieve these benefits by using new computing techniques to generate and deliver a new computer file structure providing an index to ABS video streams. The inventions also achieve these benefits with new computing techniques for generating the new top level index file structure. The new index file structure and new computing techniques allow the playback device to control ABS by selecting video streams depending on the device's streaming conditions.

- For example, the inventions' new index file structure includes a reference to cryptographic information for decrypting encrypted video streams. Including the reference to cryptographic information in the new index file structure reduces the communications and computations needed for the playback device to obtain the cryptographic information and decrypt encrypted video streams. This structure improves upon prior systems that required the playback device to perform additional network communications and data processing steps to obtain information needed to decrypt encrypted video. The inventions' new file structure reduce the time necessary for decryption and the computing resources needed at the playback device compared to prior approaches. The video streams provided in the new index file also share common cryptographic information. Sharing common cryptographic information further reduces the computing resources needed at the playback device. These improvements enable playback devices to begin playing back video faster compared to prior approaches.
- The inventions' new index file structure includes video streams and cryptographic information optimized for the specific playback device requesting the index. This structure improves device ABS performance by reducing the number of streams in the index the device has to process, which reduces the computing resources

needed at the playback device. Reducing the computing resources needed at the playback device expands the range of devices that the streaming service can support, speeds up startup when beginning streaming, and reduces the likelihood of video stalls during ABS. The device-optimized index file also enables a video streaming service to provide higher-resolution content for each device and reduce stalls and errors during playback. The device-optimized index file also allows a video streaming service to exercise control over the video streams available to each device and user and to improve security of those video streams compared to prior approaches.

- The inventions also provide new computing techniques for generating the new index file structure. For example, the inventions generate the new index file structure by interpreting data received from the playback device about the device's characteristics. The inventions automatically generate the index file during real-time communication with the device. The inventions filter available video streams based on characteristics sent by the specific device. The inventions generate the new index file from the locations of those streams stored on the video streaming service's servers. The inventions also incorporate a reference to cryptographic information into the new file structure to facilitate efficient decryption and playback.
- The inventions' new computing techniques, including automatically generating the new index file during communication with a playback device, eliminates the need for a video streaming service to generate and store a specific index file for all potential combinations of video content, stream bitrates, resolution, aspect

> ratio, user preferences, content permissions and restrictions, encoding formats, buffer sizes, device security levels, and other device- and user-specific criteria. Automatically generating the new index file during communication with a playback device thus enables a video streaming service to reduce computing costs of generating and storing index files on video streaming servers. These benefits increase as a video streaming service supports more users and more devices.

See, e.g., '061 patent at 1:39-61, 2:6-9, 2:33-50, 2:56-61, 3:34-58, 4:24-34, 4:45-50, 5:27-36, 6:29-39, 6:49-54, 6:61-7:20, 7:48-67, 8:13-20, 8:45-60, 9:52-58, 10:30-41, 10:45-49, 11:24-38, 11:47-12:50, 12:51-13:10, 13:18-67, 20:6-36, FIG. 3, FIG. 4, FIG. 5, FIG. 9.

Prosecution History of the '061 Inventions

144. During prosecution, the patent examiner did not reject any claims of the '061 patent for lack of subject matter eligibility under 35 U.S.C. § 101.

Claims Reciting the Technical Solutions of the '061 Inventions

145. The '061 claims recite methods and systems setting forth how to improve the functionality of computer systems that perform adaptive bitrate streaming of protected video content using automatically generated top level index files. Claim 1 of the '061 patent recites how to perform adaptive bitrate streaming of protected video content using automatically generated top level index files:

> 1. A method for streaming a piece of content using a set of server computer systems, comprising:
> obtaining common cryptographic information associated with a piece of content using an encrypting system;
> encoding multiple streams of video content associated with a piece of content using an encoding system;

encrypting the encoded multiple streams of video content into multiple streams of protected video content using the common cryptographic information using the encrypting system;

storing the multiple streams of protected video content in memory at a server system;

storing the common cryptographic information at a digital rights management system;

receiving a request for a top level index file from a playback device at the server system, where the request identifies the piece of content and includes information describing the playback device;

automatically filtering the multiple streams of protected video content into a plurality of alternative streams of protected video content based upon the received information describing the playback device using the server system;

automatically generating a top level index file in response to the request for a top level index file from the playback device using the server system, where the top level index file (i) identifies the location of each of the plurality of alternative streams of protected video content, where each of the plurality of alternative streams of protected video content encodes the piece of content at a different bitrate, (ii) describes at least a bitrate of each of a plurality of alternative streams of protected video content associated with the identified piece of content, and (iii) includes a reference to the common cryptographic information for

accessing the protected video content;

sending the top level index file to the playback device from the server system;

receiving at the server system a request for at least a portion of an initial stream of protected video content, selected from the plurality of alternative streams of protected video content, from the playback device;

sending the requested at least a portion of the initial stream of protected video content to the playback device from the server system;

receiving at the digital rights management system a request from the playback device for common cryptographic information to access the initial stream of protected video content;

encrypt the common cryptographic information using the digital rights management system; and

sending the encrypted common cryptographic information to the playback device from the digital rights management system.

'061 patent, 20:55-21:44.

146. The claim limitations of claim 1 achieve the benefits of maintaining startup speed and enabling a video streaming system to scale to optimize the user experience for a wide range of device capabilities on the market. In particular, the limitations of claim 1 include creating a top level index file that references common cryptographic information and providing the top level index file and the encrypted common cryptographic information to the playback device. '061 patent, 8:45-60, 18:56-60, 18:66-19:2. Further, the limitations of claim 1 include filtering the assets to exclude assets that are not desired to be played back. '061 patent, 13:1-10; 13:18-

22. *See also* '061 patent, 1:39-61, 2:6-9, 2:33-50, 2:56-61, 3:34-58, 4:24-34, 4:45-50, 5:27-36, 6:29-39, 6:49-54, 6:61-7:20; FIG. 1, 7:38-9:26; FIG. 2, 9:27-10:20; FIG. 3, 10:21-11:23; FIG. 4, 11:24-12:50; FIG. 5, 12:51-13:67; FIG. 6, 14:1-18:32; FIG. 7, 18:33-19:19; FIG. 8, 19:20-20:5; FIG. 9, 20:6-41. Claim 1 and its dependents, therefore, recite limitations that enable the technical and performance benefits of the invention described above in ¶¶ 136-143. Claim 1 recites a novel solution of adaptive bitrate streaming of protected video content using automatically generated top level index files in a manner that was new and not well-known, routine, or conventional at the time of the '061 patent.

147. Claims 2-13 of the '061 patent depend from claim 1, and each of claims 2-13 further describe how to perform the invention's improved method for adaptive bitrate streaming of protected video content using automatically generated top level index files that maintains startup speed and takes advantage of the full range of device capabilities on the market. The ordered combination of elements in each of claims 2-13, in conjunction with the elements of the claims from which they depend, therefore recite unconventional new and improved computer processes and top level index file structures that were not well-known at the time of the '061 inventions.

148. Claim 14 of the '061 patent recites a set of server computer systems configured to perform the invention's improved method for adaptive bitrate streaming of protected video content using automatically generated top level index files, reciting:

> 14. A set of server computer systems, comprising:
> an encrypting system configured to obtain common cryptographic information associated with a piece of content;
> an encoding system configured to encode multiple

streams of video content associated with a piece of content;

wherein the encrypting system is further configured to encrypt the encoded multiple streams of video content into multiple streams of protected video content using the common cryptographic information;

a server system configured to store the multiple streams of protected video content in memory; and

a digital rights management system configured to store the common cryptographic information;

wherein the server system is further configured to:

receive a request for a top level index file from a playback device, where the request identifies the piece of content and includes information describing the playback device;

automatically filter the multiple streams of protected video content into a plurality of alternative streams of protected video content based upon the received information describing the playback device;

automatically generate a top level index file in response to the request for a top level index file from the playback device, where the top level index file (i) identifies the location of each of the plurality of alternative streams of protected video content, where each of the plurality of alternative streams of protected video content encodes the piece of content at a different bitrate, (ii) describes at least a bitrate of each of a plurality of alternative streams of protected video content associated with the identified piece of content, and (iii) includes a reference to the

common cryptographic information for accessing the protected video content;

send the top level index file to the playback device;

receive a request for at least a portion of an initial stream of protected video content, selected from the plurality of alternative streams of protected video content, from the playback device; and

send the requested at least a portion of the initial stream of protected video content to the playback device; and

wherein the digital rights management system is further configured to:

receive a request from the playback device for common cryptographic information to access the initial stream of protected video content;

encrypt the common cryptographic information; and

send the encrypted common cryptographic information to the playback device.

Id. at 22:30-23:15.

149. The claim limitations of claim 14 achieve the benefits of maintaining startup speed and enabling a video streaming system to scale to optimize the user experience for a wide range of device capabilities on the market. In particular, the limitations of claim 14 include configuring a set of server computer systems to create a top level index file that references common cryptographic information and provides the top level index file and the encrypted common cryptographic information to the playback device. '061 patent, 8:45-60, 18:56-60, 18:66-19:2. Further, the limitations of claim 14 include configuring a set of server computer systems to filter the assets to exclude assets that are not desired to be played back. '061 patent, 13:1-10; 13:18-22. *See also* '061 patent, 1:39-61, 2:6-9, 2:33-50, 2:56-

61, 3:34-58, 4:24-34, 4:45-50, 5:27-36, 6:29-39, 6:49-54, 6:61-7:20; FIG. 1, 7:38-9:26; FIG. 2, 9:27-10:20; FIG. 3, 10:21-11:23; FIG. 4, 11:24-12:50; FIG. 5, 12:51-13:67; FIG. 6, 14:1-18:32; FIG. 7, 18:33-19:19; FIG. 8, 19:20-20:5; FIG. 9, 20:6-41. Claim 14 and its dependents, therefore, recite limitations that enable the technical and performance benefits of the invention described above in ¶¶ 136-143. Claim 14 recites a novel set of server computer systems for adaptive bitrate streaming of protected video content using automatically generated top level index files in a manner that was new and not well-known, routine, or conventional at the time of the '061 patent.

150. Claims 15-26 of the '061 patent depend from claim 14, and each of claims 15-26 further describe how to perform the invention's improved set of server computer systems for adaptive bitrate streaming of protected video content using automatically generated top level index files that maintains startup speed and takes advantage of the full range of device capabilities on the market. The ordered combination of elements in each of claims 15-26, in conjunction with the elements of the claims from which they depend, therefore recite unconventional new and improved computer systems and top level index file structures that were not well-known at the time of the '061 inventions.

VI. The '987 Patent

151. The '987 patent, entitled "Systems and Methods for Encoding Alternative Streams of Video for Use in Adaptive Bitrate Streaming," duly and legally issued on June 18, 2019, with Auke Sjoerd van der Schaar as the named inventor. The '987 patent claims priority to Provisional Application No. 61/430,502, filed on January 6, 2011.

Summary of the '987 Inventions

152. The inventions claimed in the '987 patent provides higher-quality video and fewer video stalls for a more seamless user experience during adaptive bitrate streaming. The '987 invention achieves this benefit by selecting the video

stream based on the network data rate and the data stored in the buffer on the playback device.

153. Specifically, the '987 inventions are directed to playing back alternative video streams encoded with different maximum bitrates, where the maximum bitrate of a video stream represents the rate that "ensures no underflow will occur given a certain buffer size." '987 patent, 1:53-55. The video streams are selected based on "the amount of time media is buffered prior to commencing playback so that underflow does not occur during playback." '987 patent, 4:36-39. "When the amount of media in the buffer has a playback duration equal to the upper bound seek delay, the playback device can then freely switch to the optimal stream for the channel conditions from that point onward." '987 patent, 5:35-38. For example, the '987 inventions disclose a playback device that "measures (**104**) the channel data rate and determines (**106**) whether there has been a change in channel rate. If there has been a change, the playback device chooses (**102**) the stream that is optimally encoded for the new channel rate." '987 patent, 10:52-56; FIG. 4. An exemplary process is outlined in FIG. 4 of the patent below, illustrating buffering media based on a seek delay and adapting the stream selection based on a channel rate:

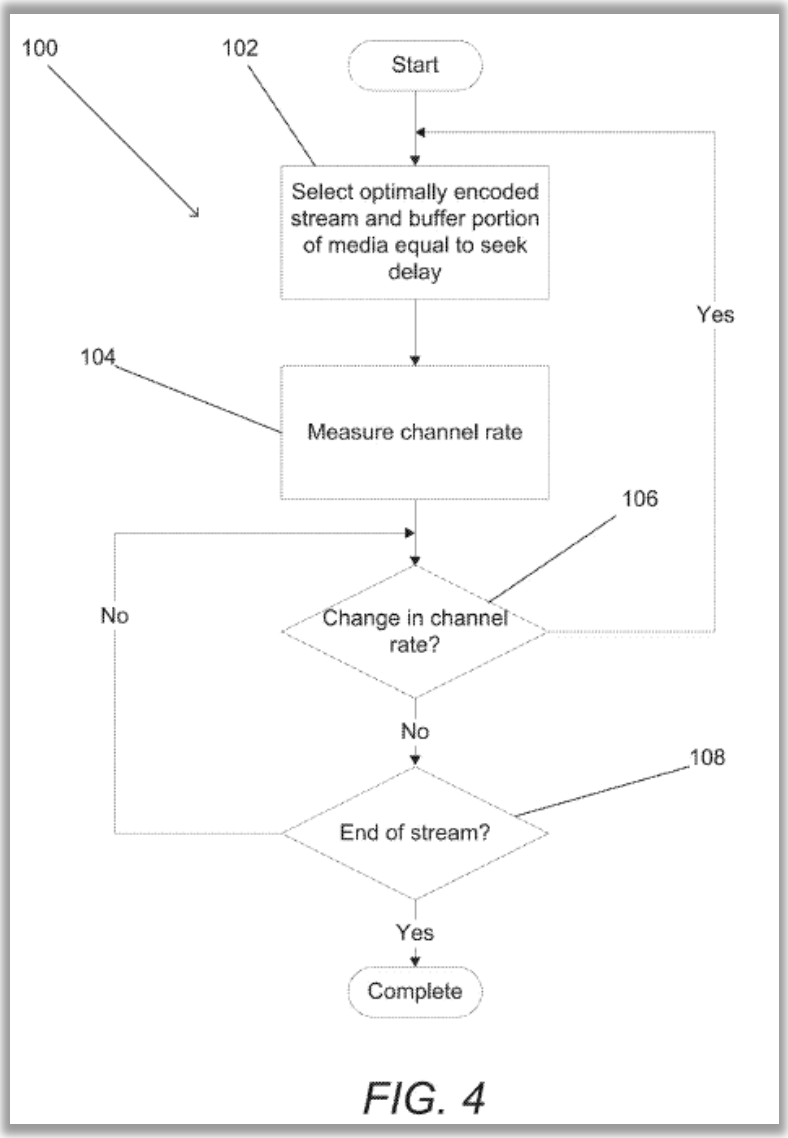


FIG. 4

'987 patent, FIG. 4.

154. These improvements provide a playback device "sufficient time to respond to a reduction in channel data rate so that the playback device can automatically switch to the optimal stream for the new channel conditions." '987 patent, 4:36-39. The inventions claimed in the '987 patent enable Hulu to deliver optimal video streams based on channel conditions and data stored in the buffer, allowing Hulu to provide a high-quality user experience during streaming of Hulu videos.

Technical Problems Addressed by the '987 Inventions

155. The '987 inventions address the problem of videos either stalling during streaming or playing back at low resolutions. Typically, "the playback device stores a sufficient quantity of media in a buffer at any given time during playback to prevent disruption of playback due to the playback device completing playback of all the buffered media prior to receipt of the next portion of media." '987 patent, 1:31-35. Video streams are usually selected based on "the present streaming conditions (e.g. the user's network bandwidth) in real time," not the buffer stored on the playback device. '987 patent, 1:36-39. Thus, disruption of playback occurs when the "playback device receives streaming media at a lower speed than the speed at which the media is played back," also known as underflow. '987 patent, 1:46-49.

156. Higher video quality requires streaming more video data, and streaming more video data increases the likelihood of stalls. Streaming higher quality video also slows down startup time because it requires transferring more data over the network to the device before playback can begin. The '987 patent addresses this technical problem, allowing a playback device "to stream the highest bitrate stream available given the streaming conditions experienced by the playback device without stalls in the playback of media due to underflow." *See, e.g.*, '987 patent, 1:43-46.

Technical Solutions and Benefits Provided by the '987 Inventions

157. The '987 patent claims specific ways to solve these technical problems by measuring network data rates and taking into account the amount of video data stored in a buffer on the playback device. The '987 claims are directed to a new adaptive bitrate streaming process to select optimally encoded video streams based on the network data rate and the buffer delay (claim 10 and dependents) and the improved playback device configured to perform the new adaptive bitrate streaming

process to select optimally encoded video streams based on the network data rate and the buffer delay (claim 1 and dependents).

158. The new adaptive bitrate streaming process in the '987 inventions includes several phases. At startup, the playback device selects a video stream based on the video stream's specified maximum bitrate and the streaming network's data rate. '987 patent, 15:7-11, 16:6-9. Based on the video stream selection, the playback device requests, stores, and plays back a portion of the stream while measuring the network data rate. '987 patent, 17:10-17. When the network data rate increases, the playback device then requests a portion of a video stream with a higher specified maximum bitrate. '987 patent, 17:19-26. When the network data rate drops, "the playback device can determine an appropriate lower bitrate stream based upon the amount of data that can be downloaded from the lower bitrate stream during the time it will take for the client application to playback the content stored within the buffer [] and the available data rate." '987 patent, 8:1-8, 17:32-50. This way, "[b]uffer underflow will not occur following the stream switch provided a sufficient amount of data is downloaded." '987 patent, 8:6-8.

159. Compared to the adaptive bitrate streaming process in prior systems, the '987 inventions allow the systems to stream the highest bitrate stream available without stalls by:

- during startup, commencing playback at a lower bitrate, '987 patent, 10:47-52;
- before the buffer reaches a certain threshold, choosing the video stream that is optimally encoded for the network data rate, '987 patent, 9:27-29, 10:44-47; and
- when the buffer reaches a certain threshold, selecting the optimal stream based on network data rate and buffer delay, '987 patent, 8:1-8.

See generally, '987 patent, 9:61-11:3. The new adaptive bitrate streaming process described in the '987 invention was new and not well-known, routine, or

conventional at the time of the '987 patent.

160. The '987 inventions are also directed to improvements to the functionality of a playback device that performs adaptive bitrate streaming, including:

- streaming system architecture, '987 patent, FIG. 1A, FIG. 4, 5:43-6:3, 9:61-11:3;
- source encoders, '987 patent, FIG. 1B, 6:4-7:28; and
- playback devices, '987 patent, FIG. 1A, FIG. 1C, 7:29-8:34.

The system configured to perform the new adaptive bitrate streaming process in the '987 inventions was new and not well-known, routine, or conventional at the time of the '987 patent.

161. The new adaptive bitrate streaming process to select optimally encoded video streams based on the network data rate and the buffer stored on the playback device as described in the '987 inventions, and system configured to perform the new process, provide technical benefits that improve the functionality and capabilities of a playback device performing these operations. *See, e.g.*, '987 patent at FIG. 1A, FIG. 4, 5:43-6:3, 9:61-11:3 (describing improvement to adaptive bitrate streaming architecture), 7:29-8:34, FIG. 1A, FIG. 1C (describing improvement to playback device that performs adaptive bitrate streaming). By measuring the network data rate and deciding that the buffer is sufficient to prevent buffer underflow, "[b]uffer underflow will not occur following the stream switch provided a sufficient amount of data is downloaded." '987 patent, 8:6-8; 15:31-46; 17:27-43.

162. In sum, the inventions claimed in the '987 patent improve adaptive bitrate streaming. The claims recite new computing techniques that improve the performance of computing systems (servers and playback devices, connected by networks) performing ABS. These improvements make the ABS user experience better. For example, the inventions consistently deliver higher-resolution video than

prior technical approaches to ABS. The inventions also reduce stalls during ABS compared to prior technical approaches. The inventions also improve startup times compared to prior technical approaches. The inventions enable flexibility for streaming service providers to decide how to prioritize among improving startup times, smooth stream switching, and avoiding stalls, while improving all three. The inventions achieve these benefits with new computing techniques for determining when to switch among video streams having different maximum bitrates. For example, these techniques include comparing measured network data rates and amount of buffered video content to alternative streams' maximum bitrates to select the highest-resolution video available while avoiding stalls during playback, even if network data rates drop. *See, e.g.*, '987 patent at 4:22-29, 4:33-55, 4:56-5:3, 5:28-42, 6:33-49, 8:1-27, 8:36-45, 9:26-46, 9:47-60, 10:21-39, 11:4-20, 11:63-12:19, 12:47-13:20, 13:29-14:17, FIG. 4, FIG. 5.

Prosecution History of the '987 Inventions

163. During prosecution, the patent examiner did not reject any claims of the '987 patent for lack of subject matter eligibility under 35 U.S.C. § 101.

Claims Reciting the Technical Solutions of the '987 Inventions

164. The '987 claims recite methods and systems setting forth how to improve the functionality and performance of a playback device that performs adaptive bitrate streaming based on the network data rate and the buffer stored on the playback device. Claim 1 of the '987 patent recites a playback device configured to perform the invention's improved method for adaptive bitrate streaming based on the network data rate and the buffer stored on the playback device to stream the highest bitrate stream available without stalls, reciting:

> 1. A playback device for playing content from a plurality of alternative streams, the playback device comprising:
>
> a set of one or more processors; and

a non-volatile storage containing an application for causing the set of one or more processors to perform steps of:

obtaining a top level index file identifying a plurality of alternative video streams and specifying a maximum bitrate for each of the plurality of alternative video streams, where the plurality of alternative video streams comprises a first and a second alternative video stream and the specified maximum bitrate of the second alternative video stream is higher than the specified maximum bitrate of the first alternative video stream;

during an initial startup period:

obtaining at least one network data rate measurement;

selecting the first alternative video stream based upon a comparison between the specified maximum bitrates for each of the plurality of streams and the at least one network data rate measurement;

requesting at least one chunk of the first alternative video stream;

storing the at least one chunk of the first alternative video stream in a buffer of the playback device; and

playing back at least one chunk of the first alternative stream stored in the buffer;

obtaining at least one additional network data rate measurement;

determining that the network data rate is greater than the specified maximum bitrate for the second alternative video stream based upon the at least one additional

network data rate measurement;

when the network data rate is determined to be greater than the specified maximum bitrate for the second alternative video stream, requesting at least one chunk of the second alternative video stream;

when a minimum buffer level criterion is satisfied based upon a playback duration of chunks of video content stored in the buffer of the playback device:

obtaining at least one further network data rate measurement;

selecting a stream from the plurality of alternative video streams based upon a playback duration of chunks of video content stored in the buffer of the playback device by selecting a stream from the plurality of alternative video streams such that the playback duration of chunks of video content stored in the buffer of the playback device is sufficient to prevent buffer underflow during downloading and playback of at least one chunk of the selected video stream based upon the at least one further network data rate measurement;

requesting at least one chunk of the selected stream from the plurality of alternative video streams;

storing the at least one chunk of the selected stream from the plurality of alternative video streams in the buffer of the playback device; and

playing back the at least one chunk of the selected stream from the plurality of alternative video streams stored in the buffer.

Id. at 14:56-15:54.

165. The claim limitations of claim 1 describe how to achieve the benefits of streaming the highest bitrate stream available without stalls. In particular, the limitations of claim 1 include a playback device configured to, during startup, commence playback at a lower bitrate; before the buffer reaches a certain threshold, choose the video stream that is optimally encoded for the network data rate; and when the buffer reaches a certain threshold, select the optimal stream based on network data rate and buffer delay. '987 patent, FIG. 1A, FIG. 4, FIG. 5, 4:22-29, 4:33-55, 4:56-5:3, 5:28-42, 5:43-6:3, 6:33-49, 8:1-27, 8:36-45, 9:26-46, 9:47-60, 9:61-11:3; 11:4-20, 11:63-12:19, 12:47-13:20, 13:29-14:17; FIG. 1B, 6:4-7:28; FIG. 1A, FIG. 1C, 7:29-8:34; 8:6-8; 15:31-46; 17:27-43. Claim 1 and its dependents, therefore, recite limitations that enable the technical and performance benefits of the invention described above in ¶¶ 157-162. Claim 1 recites a novel playback device for adaptive bitrate streaming based on the network data rate and the buffer stored on the playback device in a manner that was new and not well-known, routine, or conventional at the time of the '987 patent.

166. Claims 2-9 of the '987 patent depend from claim 1, and each of claims 2-9 further describe the invention's improved playback device for adaptive bitrate streaming based on the network data rate and the data stored in the buffer on the playback device. The ordered combination of elements in each of claims 2-9, in conjunction with the elements of the claims from which they depend, therefore recite unconventional new and improved playback devices that were not well-known at the time of the '987 inventions.

167. Claim 10 of the '987 patent recites how to perform the invention's improved method of adaptive bitrate streaming based on the network data rate and the buffer stored on the playback device:

> 10. A method of playing back media content from a

plurality of alternative streams on a playback device, comprising:

obtaining a top level index file identifying a plurality of alternative video streams, and specifying a maximum bitrate for each of the plurality of alternative video streams, where the plurality of alternative video streams comprises a first and a second alternative video stream and the specified maximum bitrate of the second alternative video stream is higher than the specified maximum bitrate of the first alternative video stream;

during an initial startup period:

obtaining at least one network data rate measurement;

selecting the first alternative video stream based upon a comparison between the specified maximum bitrates for each of the plurality of streams and the at least one network data rate measurement;

requesting at least one chunk of the first alternative video stream;

storing the at least one chunk of the first alternative video stream in a buffer of the playback device;

playing back at least one chunk of the first alternative stream stored in the buffer:

obtaining at least one additional network data rate measurement;

determining that the network data rate is greater than the specified maximum bitrate for the second alternative video stream based upon the at least one additional network data rate measurement; and

when the network data rate is determined to be greater than the specified maximum bitrate for the second alternative video stream, requesting at least one chunk of the second alternative video stream; and

when a minimum buffer level criterion is satisfied based upon a playback duration of chunks of video content stored in the buffer of the playback device:

obtaining at least one further network data rate measurement;

selecting a stream from the plurality of alternative video streams based upon a playback duration of chunks of video content stored in the buffer of the playback device by selecting a stream from the plurality of alternative video streams such that the playback duration of chunks of video content stored in the buffer of the playback device is sufficient to prevent buffer underflow during downloading and playback of at least one chunk of the selected video stream based upon the at least one further network data rate measurement;

requesting at least one chunk of the selected stream from the plurality of alternative video streams;

storing the at least one chunk of the selected stream from the plurality of alternative video streams in the buffer of the playback device; and

playing back the at least one chunk of the selected stream from the plurality of alternative video streams stored in the buffer.

'987 patent, 16:60-17:50.

168. The claim limitations of claim 10 explain how to achieve the benefits of streaming the highest bitrate stream available without stalls. In particular, the limitations of claim 10 include during startup, commencing playback at a lower bitrate; before the buffer reaches a certain threshold, choosing the video stream that is optimally encoded for the network data rate; and when the buffer reaches a certain threshold, selecting the optimal stream based on network data rate and buffer delay. '987 patent, FIG. 1A, FIG. 4, FIG. 5, 4:22-29, 4:33-55, 4:56-5:3, 5:28-42, 5:43-6:3, 6:33-49, 8:1-27, 8:36-45, 9:26-46, 9:47-60, 9:61-11:3; 11:4-20, 11:63-12:19, 12:47-13:20, 13:29-14:17; FIG. 1B, 6:4-7:28; FIG. 1A, FIG. 1C, 7:29-8:34; 8:6-8; 15:31-46; 17:27-43. Claim 10 and its dependents, therefore, recite limitations that enable the technical and performance benefits of the invention described above in ¶¶ 157-162. Claim 10 recites a novel solution of adaptive bitrate streaming based on the network data rate and data stored in the buffer on the playback device in a manner that was new and not well-known, routine, or conventional at the time of the '987 patent.

169. Claims 11-18 of the '987 patent depend from claim 10, and each of claims 11-18 further describe how to perform the invention's improved method for adaptive bitrate streaming based on the network data rate and the buffer stored on the playback device to stream the highest bitrate stream available without stalls. The ordered combination of elements in each of claims 11-18, in conjunction with the elements of the claims from which they depend, therefore recite unconventional new and improved computer processes that were not well-known at the time of the '987 inventions.

HULU'S INTERNAL TESTING

170. Upon information and belief, Hulu tests its software application and video streaming service on CE devices to confirm that the application and service work properly before releasing them to users.

171. Upon information and belief, device testing is important to Hulu's success. Device testing allows Hulu to ensure that its application and service operate seamlessly on Hulu-compatible devices—a large ecosystem. Hulu's testing further ensures that iterative versions, updates, and subsequent releases of the application and service remain compatible and operable with consumer devices.

172. Upon information and belief, Hulu employs engineers to test Hulu's software and video streaming services in the United States. For example, Hulu advertises its Quality Assurance Engineer position to include responsibilities such as "design[ing] test plans and creat[ing] test cases for exciting new features being introduced."[23]

173. Upon information and belief, Hulu directly infringes the DivX Patents during Hulu's internal testing of its application and video streaming service on consumer devices.

174. Upon information and belief, Hulu's internal testing enables Hulu to deliver its application and service in the United States and worldwide.

175. Protecting content has become more important for studios in today's high-resolution world.[24] Upon information and belief, Hulu tests the DRM technologies that it employs to protect the security of the video content that it licenses from third parties, including studios, and that it produces itself. Upon information and belief, Hulu's internal testing of the DRM technologies it employs, therefore, enables Hulu to obtain video content from third parties and to invest in its own production of original content, which leads to increased adoption of Hulu's service by paying members in the United States and worldwide.

[23] https://lensa.com/quality-assurance-engineer-android-jobs/seattle/jd/43262aac704de51d7a888b3fe0e43468.

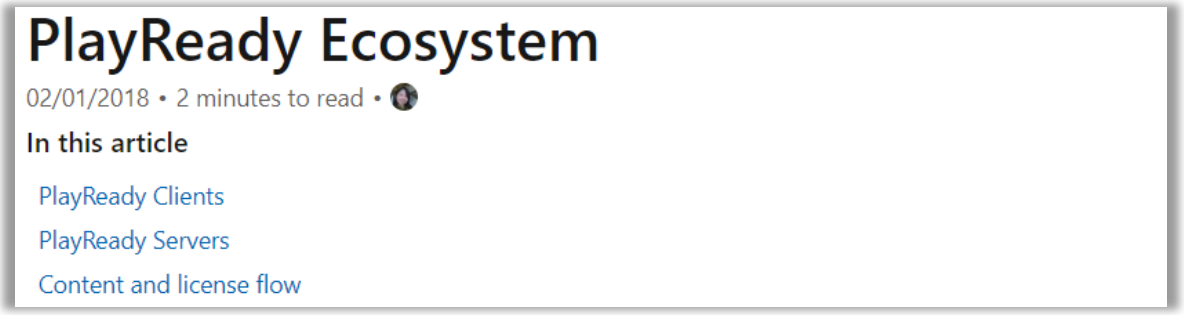
[24] https://www.streamingmedia.com/Articles/ReadArticle.aspx?ArticleID=127701.

HULU'S USE OF DRM SERVERS

176. By provisioning and operating its own DRM servers, Hulu directly infringes the asserted claims reciting a DRM server or a method performed by a DRM server.[25] Hulu's operates DRM servers, for example, based on Microsoft's PlayReady DRM architecture.[26]

> Hulu also began deploying DASH early for Chromecast devices and now uses it for all new deployments, strictly based on the ISO base media file format (BMFF) on-demand profile with demuxed essences and using CENC plus PlayReady and Widevine DRMs.

177. Streaming service providers such as Hulu "use the PlayReady Server Software Development Kit (SDK) to build Servers with service-specific business logic."[27]



[25] *See, e.g.*, '061 patent, claim 1, reciting "storing the common cryptographic information at a digital rights management system."

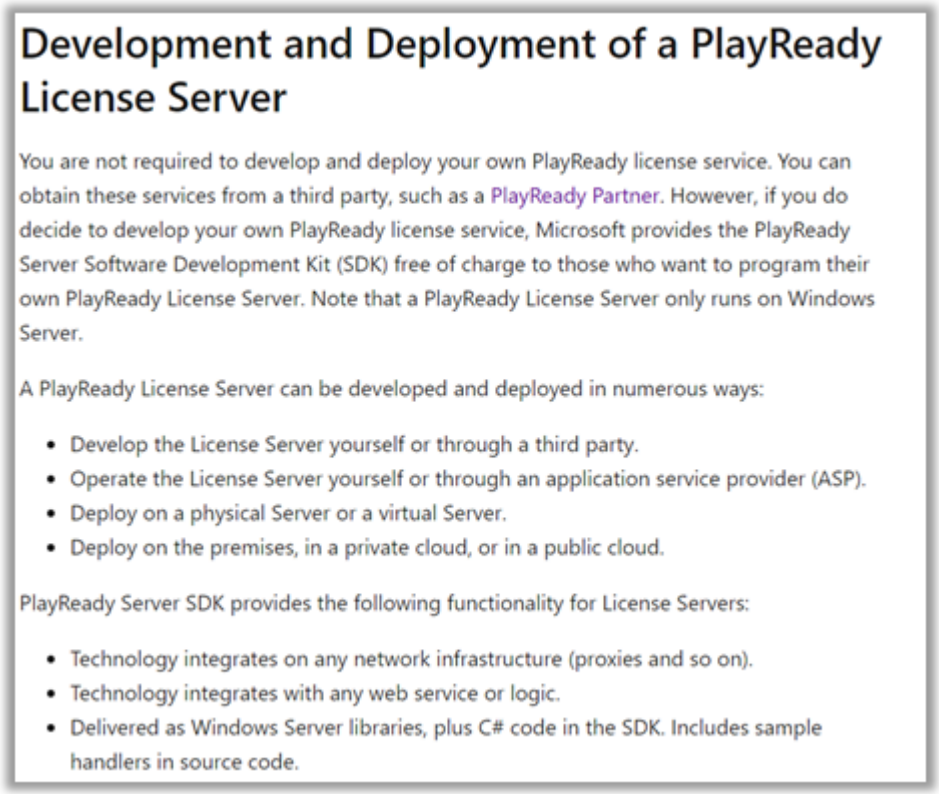
[26] *See, e.g.*, http://www.streamingmediaglobal.com/Articles/ReadArticle.aspx?ArticleID=96144&PageNum=1.

[27] https://docs.microsoft.com/en-us/playready/overview/ecosystem.

PlayReady Servers

Customized application Servers enable interoperation with the clients. Service providers use the PlayReady Server Software Development Kit (SDK) to build Servers with service-specific business logic. For example, a subscription service would customize Servers to have a service-specific license. The license might include expiration times and license issuance restrictions that tie to a specific data backend that has subscriber information. By using the PlayReady Server SDK, the customized Server builder can be confident the service will protect content and issue licenses that work with PlayReady Clients.

178. Upon information and belief, Hulu has developed and deployed its DRM servers according to the PlayReady SDK provided by Microsoft.[28]



Development and Deployment of a PlayReady License Server

You are not required to develop and deploy your own PlayReady license service. You can obtain these services from a third party, such as a PlayReady Partner. However, if you do decide to develop your own PlayReady license service, Microsoft provides the PlayReady Server Software Development Kit (SDK) free of charge to those who want to program their own PlayReady License Server. Note that a PlayReady License Server only runs on Windows Server.

A PlayReady License Server can be developed and deployed in numerous ways:

- Develop the License Server yourself or through a third party.
- Operate the License Server yourself or through an application service provider (ASP).
- Deploy on a physical Server or a virtual Server.
- Deploy on the premises, in a private cloud, or in a public cloud.

PlayReady Server SDK provides the following functionality for License Servers:

- Technology integrates on any network infrastructure (proxies and so on).
- Technology integrates with any web service or logic.
- Delivered as Windows Server libraries, plus C# code in the SDK. Includes sample handlers in source code.

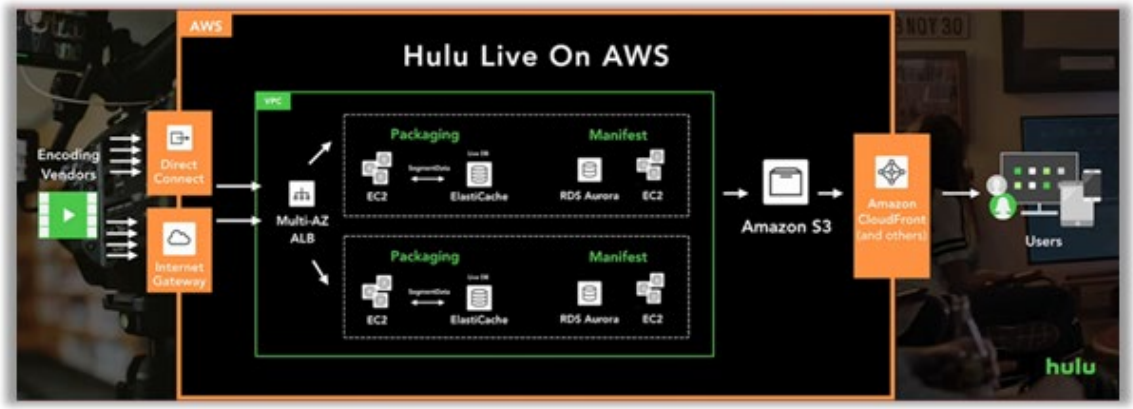
179. Thus, as demonstrated by publicly available information regarding Microsoft's PlayReady DRM architecture, Hulu directly infringes the asserted

[28] https://docs.microsoft.com/en-us/playready/overview/license-server.

claims reciting a DRM server by designing, deploying, and using a DRM server to support Hulu's streaming services.

HULU'S USE OF CONTENT PACKAGING OR MANIFEST SERVERS

180. By provisioning and operating its own content packaging and manifest servers, Hulu directly infringes the asserted claims reciting a content packaging or manifest server.[29] Hulu's content packaging or manifest server, for example, is based on Amazon's Virtual Private Cloud ("VPC").[30]

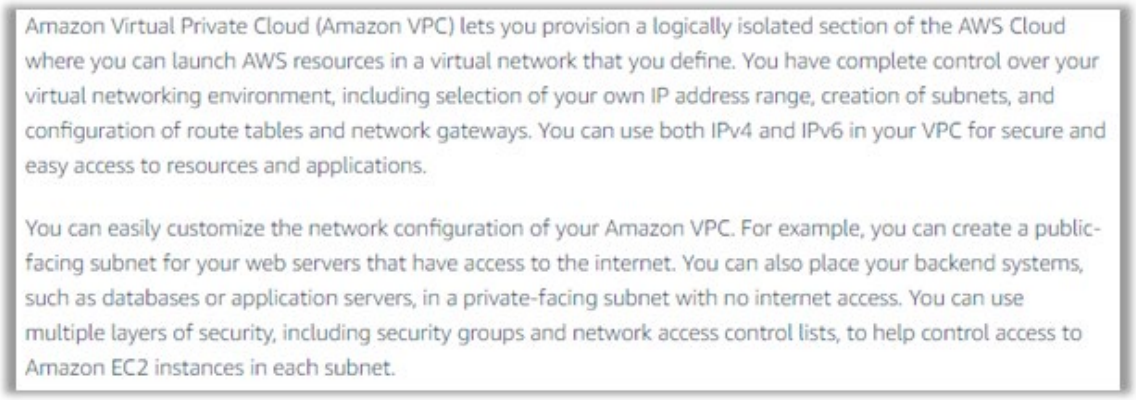


181. Hulu has "complete control over [its] virtual networking environment" and can "easily customize the network configuration" using the Amazon VPC.[31]

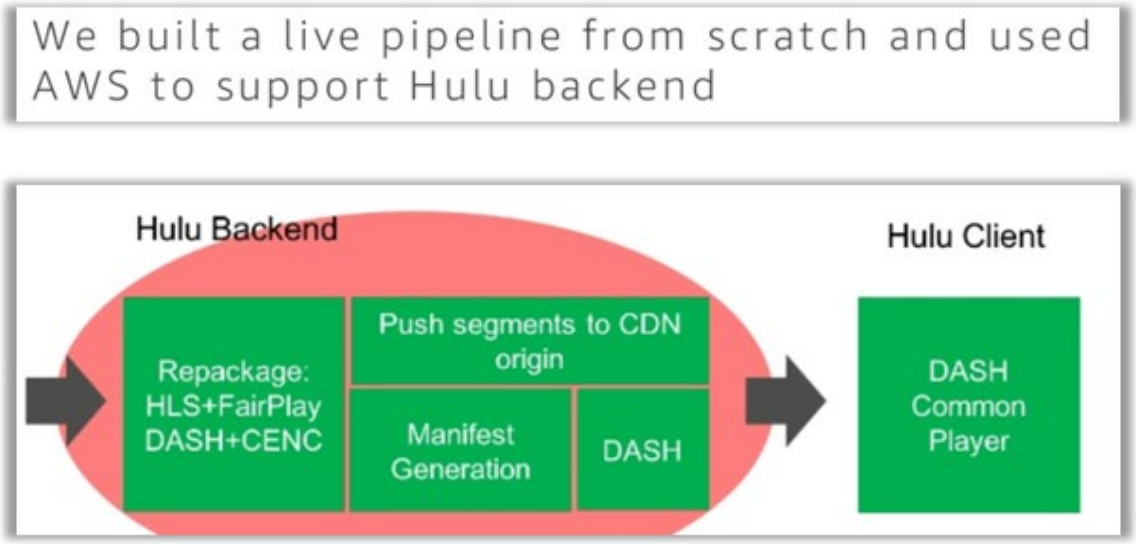
[29] *See, e.g.*, '061 patent, claim 1, reciting "storing the multiple streams of protected video content in memory at a server system" and "automatically generating a top level index file in response to the request for a top level index file from the playback device using the server system."

[30] https://aws.amazon.com/solutions/case-studies/hulu/.

[31] https://aws.amazon.com/vpc/.



182. Hulu "built a [Hulu] live pipeline from scratch and used AWS to support Hulu backend."[32]



183. Hulu's live streaming service architecture shares a "substantial part" of its architecture with its video-on-demand (VOD) service.[33]

[32] https://www.slideshare.net/AmazonWebServices/case-study-how-hulu-reinvented-television-using-the-aws-cloud-ctd302-reinvent-2017 (slide 13).

[33] https://medium.com/hulu-tech-blog/introducing-the-hulu-technical-landscape-93f4c136c568.

> On the functional side, we have the existing VOD architecture, which has recently been augmented by a live streaming architecture. To do this we recreated a substantial part of our system to add the concepts of live assets & program availability, in addition to rewriting all of our clients. This was a significant company-wide effort which resulted in a major new product for us. A key part of this effort was establishing a new metadata catalog with all live and VOD shows, populated by various sources and augmented by the out-of-band (SCTE224) and in-band (SCTE35) markers for program start and end. This is a significant and complex effort as data quality is paramount. Further, our services run at great scale due to the large number of subscribers.

184. Thus, as demonstrated by publicly available information regarding Amazon's AWS and VPC, Hulu directly infringes the asserted claims reciting a content packaging or manifest server by designing, deploying, and using a packaging or manifest server to support Hulu's streaming services.

HULU'S INDIRECT INFRINGEMENT

185. Hulu has indirectly infringed and continues to indirectly infringe at least the '318 patent, the '141 patent, the '061 patent, and the '987 patents (collectively, the "Indirectly Infringed DivX Patents") by inducing third parties to directly infringe those patents.

186. Hulu has induced, and continues to induce, direct infringement of the Indirectly Infringed DivX Patents by customers, importers, sellers, resellers, and/or end users of infringing playback devices enabled with the Hulu application and service.

I. Hulu's Knowledge of the DivX Patents

187. At the very latest, Hulu had actual knowledge of the DivX Patents and of its infringement as of the date of this Complaint.

188. Hulu also had actual knowledge of the DivX Patents and of its infringement as of March 5, 2019, the date of DivX's Complaint in *DivX, LLC v. Hulu, LLC*, 2:19-cv-01606-PSG-DFM (C.D. Cal.).

189. Hulu also had actual knowledge of the DivX Patents and of its infringement as of May 2019, when DivX sat down with Hulu to engage in good faith negotiations for Hulu to take a license to DivX's patents.

II. Hulu's Knowledge of Third-Party Actions Infringing DivX's Patents

190. Hulu is a known market leader and one of the dominant players in internet digital video streaming.

191. Hulu possesses the technical expertise required to understand the scope of the inventions claimed in the DivX Patents.

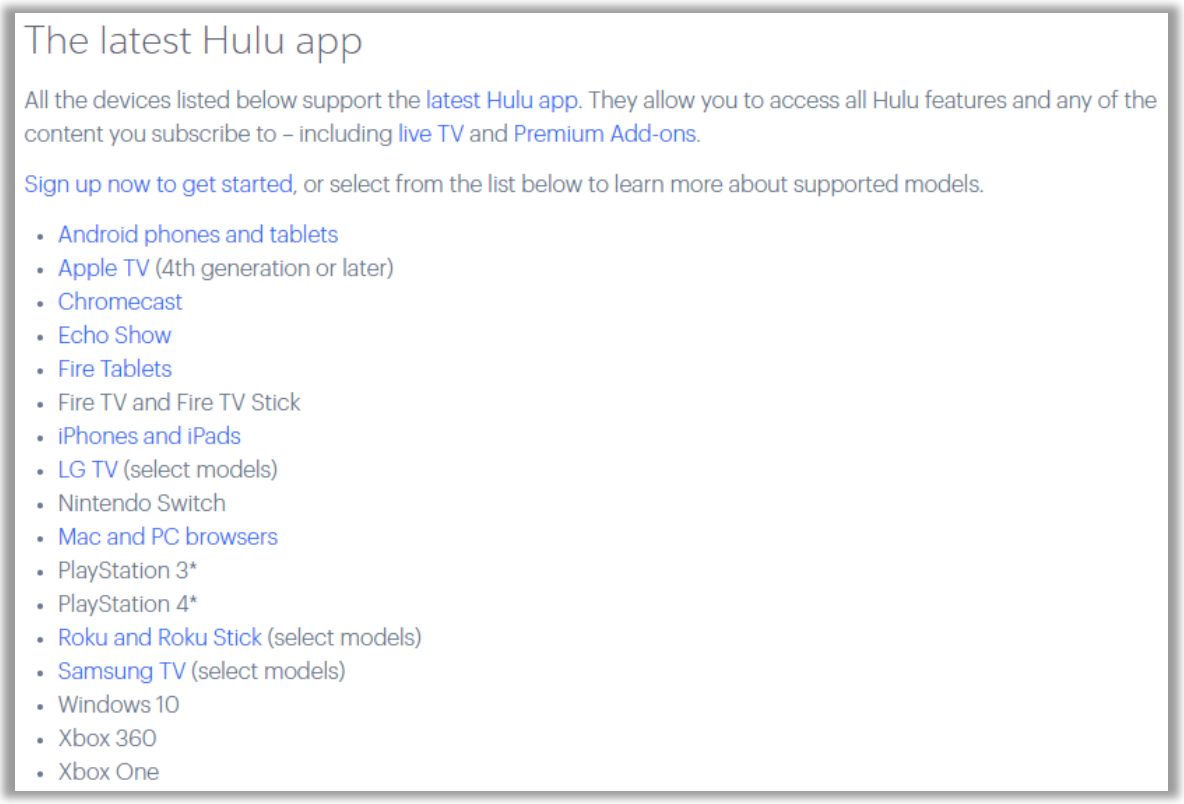
192. Hulu knows that it provides and markets an application, through its website,[34] the Apple App Store,[35] and the Google Play Store,[36] for use on playback devices that causes the playback devices and their users, importers, sellers, resellers, and customers to directly infringe Indirectly Infringed DivX Patents, when used as intended with Hulu's internet video streaming service. Indeed, as discussed, Hulu broadcasts that, "[a]s a Hulu subscriber, you'll be able to stream your favorite content from the comfort of your very own couch to the confines of public transportation using any one of our supported devices."[37]

[34] https://www.hulu.com/press/hulu-app-now-available-for-windows-10/.

[35] https://itunes.apple.com/us/app/hulu-watch-tv-shows-movies/id376510438.

[36] https://play.google.com/store/apps/details?id=com.hulu.plus&hl=en_US.

[37] https://help.hulu.com/en-us/supported-devices.



The latest Hulu app

All the devices listed below support the latest Hulu app. They allow you to access all Hulu features and any of the content you subscribe to – including live TV and Premium Add-ons.

Sign up now to get started, or select from the list below to learn more about supported models.

- Android phones and tablets
- Apple TV (4th generation or later)
- Chromecast
- Echo Show
- Fire Tablets
- Fire TV and Fire TV Stick
- iPhones and iPads
- LG TV (select models)
- Nintendo Switch
- Mac and PC browsers
- PlayStation 3*
- PlayStation 4*
- Roku and Roku Stick (select models)
- Samsung TV (select models)
- Windows 10
- Xbox 360
- Xbox One

193. Hulu actively encourages the installation and use of its application and service on consumer devices. Hulu has successfully pursued agreements with telecommunication network providers such as Sprint and AT&T to make Hulu's service available through consumer devices.[38] Hulu provides directions to consumers on how to download and install its application and service on different consumer devices.[39]

194. Hulu knows that its application is enabled in infringing playback devices used by over 25 million subscribers.[40]

[38] https://www.sprint.com/en/support/solutions/services/hulu.html; https://about.att.com/story/att_to_offer_hulu_subscription_streaming_service_to_customers.html.

[39] https://help.hulu.com/s/all-devices?language=en_US.

[40] https://www.pcmag.com/article/348989/netflix-vs-hulu-streaming-service-showdown.

195. Hulu knows that third parties—including playback device users, importers, sellers, resellers, and customers—make, use, offer to sell, sell, and/or import into the United States playback devices and other products that incorporate and enable the Hulu application. Indeed, Hulu encourages use of its application on "your favorite devices."[41]

196. Hulu knows that a number of devices come with the app "pre-installed."[42] Hulu announces when it rolls out user interfaces for CE devices. For example, it announced when it "roll[ed] out the updated UI [user interface] and access to Hulu with Live TV to even more devices across platforms including LG, Samsung and Roku."[43] And it announced when it rolled out its application for PlayStation3 & PlayStation4.[44]

197. Upon information and belief, Hulu has designed its application such that, when third party CE playback devices incorporate and/or enable the Hulu application and such third party devices with the Hulu application are used as intended, the third-party products with the application directly infringe one or more claims of the Indirectly Infringed DivX Patents when made, used, offered for sale, or sold in the United States, or when imported into the United States, as set forth in exemplary detail in the Counts herein.

198. At least as of the date of this Complaint, and based on its knowledge of the scope of the DivX Patents, its application, and products enabling that application, Hulu knows that third party sellers, resellers, importers, customer end-users, and other third parties have directly infringed and continue to directly

[41] https://www.hulu.com/welcome.

[42] https://help.hulu.com/en-us/download-hulu ("The Hulu app may come pre-installed . . . .").

[43] https://www.hulu.com/press/hulu-update/lg-samsung-roku/.

[44] https://www.hulu.com/press/hulu-update/playstation-new-hulu/.

infringe at least one claim of each of the Indirectly Infringed DivX Patents, as set forth in exemplary detail in the Counts herein.

III. Hulu's Specific Intent to Cause Third-Party Actions Infringing DivX's Patents

199. Upon information and belief, Hulu has designed, marketed, and sold its application and service to third parties with knowledge and the specific intent to cause the third parties to make, use, offer to sell, or sell in the United States, and/or import into the United States products incorporating and enabling the Hulu application and service.

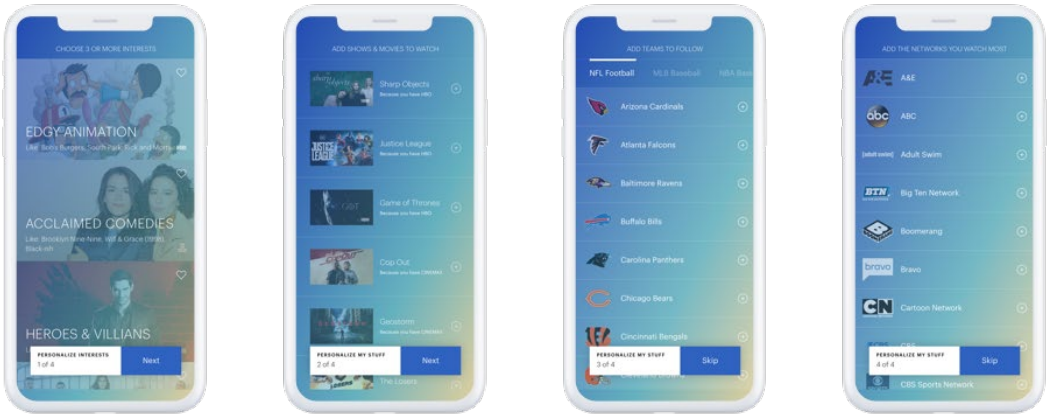
200. Upon information and belief, Hulu actively encourages its customers and end users to directly infringe the Indirectly Infringed DivX Patents by encouraging them to use the Hulu application as intended on various playback devices.

201. Hulu specifically encourages its customers to download its application onto a number of CE devices and provides detailed instructions for its users to download the app to particular CE devices.[45]

202. Hulu offers "mobile onboarding" image instructions, shown here.[46]

[45] https://help.hulu.com/en-us/download-hulu.

[46] https://www.hulu.com/press/products-assets/.



203. Upon information and belief, at least as of the date of this Complaint, Hulu intends and continues to intend to induce patent infringement by these third parties, has actual knowledge that the inducing acts would cause infringement, or is willfully blind to the possibility that its inducing acts would cause infringement.

204. Upon information and belief, Hulu indirectly infringes one or more claims of the Indirectly Infringed DivX Patents by inducing numerous third-parties to make, have made, use, sell, offer to sell, and/or import into the United States playback devices with the Hulu application installed and/or enabled.

COUNT I: INFRINGEMENT OF U.S. PATENT NO. 10,257,443

205. The allegations of ¶¶ 1-204 of this Complaint are incorporated by reference as though fully set forth herein.

206. Hulu directly infringes at least claim 7 of the '443 patent under 35 U.S.C. § 271(a), as set forth in ¶¶ 207-220.

207. Hulu uses "[a] system for encoding multimedia files." Hulu provides a streaming library of more than 80,000 TV episodes and movies.[47] '443 patent, claim 7. Hulu uses an encoder for its streaming service. "Regardless of the origin of the content, Hulu maintains control of its streaming digital video players and

[47] https://www.hulu.com/welcome.

content pipelines, and encodes all video content and advertisements prior to distribution within Hulu's proprietary video players on Web and within Hulu applications."[48]

208. Hulu's system for encoding multimedia files comprises "a network interface." '443 patent, claim 7. For example, Hulu's network interface facilitates receiving video content from content providers and distributing encoded video content.[49] On information and belief, Hulu obtains video content via a network interface (i.e., on information and belief, Hulu does not receive video content via physical media).

209. Hulu's system for encoding multimedia files comprises "at least one processing unit." '443 patent, claim 7. Any computing system that encodes multimedia files includes at least one processing unit.

210. Hulu's system for encoding multimedia files comprises "a non-transitory memory storing an encoding application, wherein the encoding application causes the at least one processing unit to encode multimedia files." '443 patent, claim 7. Hulu encodes video files for its streaming service and, on information and belief, stores an encoding application in non-transitory memory, to preserve the application.

211. Hulu encodes multimedia files by "obtaining source media using the network interface, wherein the source media comprises video." '443 patent, claim 7. Hulu's system for encoding multimedia files obtains source media—video content—using its network interface.[50]

---

[48] https://advertising.hulu.com/dom/ (copyright 2018 Hulu, LLC).

[49] *Id.*; https://www.slideshare.net/AmazonWebServices/case-study-how-hulu-reinvented-television-using-the-aws-cloud-ctd302-reinvent-2017, at slide 14.

[50] https://advertising.hulu.com/dom/ (copyright 2018 Hulu, LLC); https://medium.com/hulu-tech-blog/the-anatomy-of-a-live-ott-service-c8f6078b24d3 (depicting Hulu's Live TV service architecture, which shares a

212. Hulu encodes multimedia files by "encoding at least one video track as a plurality of video chunks, the video chunks being portions of the at least one video track, the video track comprising a series of encoded video frames." '443 patent, claim 7. Hulu encodes and packages video files, for example, mp4, fmp4, or other container files, containing at least one video track. Hulu encodes video tracks as a plurality of video chunks, using container structures consistent with the ISO BMFF Standard and ISO DASH Standard.[51] Hulu video tracks contain movie fragments, comprising a movie fragment, or 'moof,' box preceding a media data, or 'mdat,' box.[52] An 'mdat' box contains "actual media data"—video content or frames.[53] A 'moof' box contains data related to the video content—frames or "samples"—stored in its corresponding 'mdat' box.[54] Thus, the 'mdat' boxes, or video chunks, are portions of the video track, and the video track comprises a series of encoded video frames.

213. Hulu encodes multimedia files by "partially encrypting at least some of the encoded frames of video so that only portions of the encoded frames of video are encrypted." '443 patent, claim 7. Hulu encodes and encrypts video files using Microsoft PIFF Specification and ISO Common Encryption Specification extensions to the ISO BMFF container file format. The *Portable encoding of audio-*

"substantial part" of its architecture with Hulu's video-on-demand (VOD) service, see https://medium.com/hulu-tech-blog/introducing-the-hulu-technical-landscape-93f4c136c568).

[51] http://www.streamingmediaglobal.com/Articles/ReadArticle.aspx?ArticleID=96144&PageNum=1.

[52] ISO/IEC 23009-1 *Information technology – Dynamic adaptive streaming over HTTP (DASH) – Part 1: Media presentation description and segment formats*, at 85, 87 (2d ed. 2014) ("ISO/IEC 23009-1" or "ISO DASH Standard").

[53] ISO/IEC 14496-12 *Information technology — Coding of audio-visual objects — Part 12: ISO base media file format*, at 4, 20 (5th ed. 2015) ("ISO/IEC 14496-12" or "ISO BMFF Standard").

[54] ISO/IEC 14496-12 at 56.

video objects: The Protected Interoperable File Format (PIFF), Ver. 1.1 (2010) (the "Microsoft PIFF Specification") defines a standard multimedia file format (i.e., PIFF) and builds on the ISO BMFF Standard. Hulu encodes and encrypts video files containing PIFF sample encryption boxes. The ISO/IEC 23001-7 *Information technology – MPEG systems technologies – Part 7: Common encryption in ISO base media file format files* (3d ed. 2016) ("ISO Common Encryption Specification" or "ISO/IEC 23001-7") also builds on the ISO BMFF Standard, including standard encryption schemes that support multiple DRM systems. Hulu encodes and encrypts video files containing a track encryption box, consistent with the ISO Common Encryption Specification. The ISO Common Encryption Specification defines the 'cenc' encryption scheme,[55] and Hulu encodes and encrypts video files using that encryption scheme. Those video files are protected using subsample encryption.[56]

214. Hulu encodes multimedia files by "encoding the DRM information as a set of DRM chunks." '443 patent, claim 7. The ISO Common Encryption Specification teaches that an encrypted video file includes a key ID ("KID"), initialization vectors ("IVs"), and information specifying encrypted portions of the video frames to be decrypted (i.e., NumClearBytes and NumEncryptedBytes). The KID attribute value identifies a key to decrypt the video file's encrypted frames.[57] The KID is stored in the track encryption box. The ISO Common Encryption Specification also teaches that each encrypted sample includes an associated IV, i.e., a per-sample IV.[58] The Microsoft PIFF Specification also teaches that frame

[55] ISO/IEC 23001-7 at 3.

[56] ISO/IEC 23001-7 at 19.

[57] ISO/IEC 23001-7 at 10, 24.

[58] ISO/IEC 23001-7 at 5, 10, 17.

decryption information includes a KID and a per-sample IV.[59] Hulu encodes / packages the DRM information—such as a flag identifying the relevant KID, the per-sample IVs, and information specifying encrypted portions of the video frames to be decrypted (i.e., NumClearBytes and NumEncryptedBytes)—as a set of DRM chunks, such as PIFF sample encryption boxes. PIFF sample encryption boxes—located within each track fragment, or 'traf,' box within each 'moof,' box—identify which parts of a sample are encrypted and which are unencrypted.[60] The ISO Common Encryption Specification also defines sample encryption boxes and related auxiliary data. A sample encryption box is located within each track fragment, or 'traf,' box within each 'moof,' box. Thus, Hulu encodes video files by encoding DRM information—such as a flag identifying the relevant KID, the sample-specific IVs, and information specifying encrypted portions of the video frames to be decrypted (i.e., NumClearBytes and NumEncryptedBytes)—as a set of DRM chunks, such as PIFF sample encryption boxes.

215. Hulu encodes multimedia files by "encoding the DRM information as a set of DRM chunks, wherein each DRM chunk of the set of DRM chunks comprises DRM information to decrypt at least one partially encrypted frame of video in at least one video chunk of the plurality of video chunks." '443 patent, claim 7. In Hulu encoded and encrypted video files, each DRM chunk in the set of DRM chunks—including, for example, each PIFF sample encryption box—contains DRM information to decrypt at least one partially encrypted frame of video in at least one video chunk of the plurality of video chunks. Each PIFF sample encryption box contains sample-specific encryption data. The box indicates the KID, the IV, and the clear (non-encrypted) and protected (encrypted) byte ranges of partially encrypted video samples. The KID attribute value identifies a

[59] Microsoft PIFF Specification at 17, 19, 22.

[60] Microsoft PIFF Specification at 23.

key to decrypt the video file's encrypted frames. The IV specifies the IV needed for decryption of a particular sample. Each 'mdat' box—or video chunk—is preceded by a 'moof' box containing at least one PIFF sample encryption box that contains KID-related information, IV information, and NumClearBytes / NumEncryptedBytes information for the video data in the corresponding 'mdat' box. The non-zero value of NumClearBytes of the first sample indicates a partially encrypted first sample in the corresponding 'mdat' box. The value of "NumClearBytes" for at least one subsample entry in each sample encryption box is non-zero, indicating that each DRM chunk comprises DRM information to decrypt at least one partially encrypted frame of video.

216. Hulu encodes multimedia files by "encoding the DRM information as a set of DRM chunks, wherein the DRM information comprises an offset value that points to the start of an encrypted block within an encoded frame and a number value that indicates the number of encrypted bytes in the encrypted block." '443 patent, claim 7. The "NumClearBytes," for example, provides an offset value that points to the start of an encrypted block within an encoded frame, and the "NumEncryptedBytes" provides a number value that indicates the number of encrypted bytes in each encrypted sample.

217. Hulu encodes multimedia files by "interleaving the video chunks and DRM chunks so that a DRM chunk for decrypting at least one partially encrypted frame within a particular video chunk is located before the particular video chunk." '443 patent, claim 7. The mp4 file format, in which each 'moof' box precedes its corresponding 'mdat' box throughout the file, illustrates that Hulu interleaves the video chunks and DRM chunks. The DRM chunks (e.g., PIFFSampleEncryptionBox) reside inside the 'traf' box located within the 'moof' box, preceding each 'mdat' box (video chunk).[61]

[61] *See, e.g.*, Microsoft PIFF Specification at 9; *Guidelines for Implementation:*

218. Hulu encodes multimedia files by "encoding at least one index chunk that includes information concerning the locations of video chunks." '443 patent, claim 7. Hulu encodes a segment index ('sidx') box. The 'sidx' is an index chunk that includes information concerning the locations of movie fragments (i.e., 'moof' box + 'mdat' box pairs), including video chunks, within the video file. The 'sidx' box "provides a compact index of one media stream within the media segment to which it applies" that "documents how a []segment is divided into one or more subsegments." A "segment" is a "portion of an ISO base media file format file, consisting of either (a) a movie box, with its associated media data (if any) and other associated boxes or (b) one or more movie fragment boxes ['moof' boxes], with their associated media data ['mdat' boxes], and other associated boxes." A "subsegment" is "a time interval of the containing []segment . . . correspond[ing] to a single range of bytes of the containing []segment." In short, the 'sidx' box references byte ranges within a video segment—'moof' box + 'mdat' box pairs—within a video file.[62] The 'sidx' box contains reference elements providing size information for subsegments / byte ranges—i.e., each 'moof' box + 'mdat' box pair—within the file. The 'sidx' includes size information that allows the device to identify the location of each 'moof' box + 'mdat' box pair and the beginning of each 'mdat' box, i.e., the first encoded video frame in each 'mdat' box. Thus, the 'sidx' includes information concerning the locations of video chunks.

219. Hulu encodes multimedia files by "writing the interleaved chunks to at least one multimedia file," for example, mp4, fmp4, or other container files. '443 patent, claim 7.

DASH-IF Interoperability Points, DASH Industry Forum, Ver. 3.0, at 104-05 (Apr. 7, 2015) (describing "common encryption in combination with ISO BMFF" and the ISO Common Encryption Specification).

[62] ISO/IEC 14496-12 at 5, 105-06, 228.

220. Hulu encodes multimedia files by "transmitting at least a portion of the at least one multimedia file using the network interface." '443 patent, claim 7. When a user requests playback of a Hulu-encoded video file, Hulu receives a request for a particular portion of that video file and transmits that portion using its network interface.

221. Hulu has infringed, and continues to infringe, at least claim 7 of the '443 patent in the United States by making, using, offering for sale, selling, and/or importing the Hulu application and service in violation of 35 U.S.C. § 271(a).

222. Hulu's infringement has caused and continues to cause damage to DivX, and DivX is entitled to recover damages sustained as a result of Hulu's wrongful acts in an amount subject to proof at trial.

COUNT II: INFRINGEMENT OF U.S. PATENT NO. 9,794,318

223. The allegations of ¶¶ 1-222 of this Complaint are incorporated by reference as though fully set forth herein.

224. Hulu directly infringes at least claim 1 of the '318 patent under 35 U.S.C. § 271(a), as set forth below in ¶¶ 225-230.

225. Hulu performs "[a] method for obtaining media from a media file for playback on a playback device from a remote server, the media formatted to represent media as chunks." '318 patent, claim 1. Hulu provides a streaming library of more than 80,000 TV episodes and movies.[63] The Hulu application obtains video content from Hulu-encoded video files from a remote server, for playback on a playback device.[64] Hulu video files represent media as chunks. Hulu encodes and packages video files, for example, mp4, fmp4, or other container files, using

[63] https://www.hulu.com/welcome.

[64] https://advertising.hulu.com/dom/ (copyright 2018 Hulu, LLC); https://help.hulu.com/s/article/watching-hulu-on-site?language=en_US.

container structures defined in and consistent with the ISO BMFF Standard and ISO DASH Standard.[65] Hulu video files contain movie fragments, comprising a movie fragment, or 'moof,' box preceding a media data, or 'mdat,' box.[66] An 'mdat' box contains "actual media data"—in video files, video content or "frames."[67] A 'moof' box contains data related to the video content—frames or "samples"—stored in its corresponding 'mdat' box.[68] Thus, the Hulu application obtains video content from a video file that represents media as chunks: 'mdat' boxes store video frames or "samples" and constitute "chunks," or a "contiguous set of samples for one track."[69]

226. Hulu "download[s] an index from a remote media file, using a playback engine implemented on the playback device, wherein the index is used to convert a chunk reference into a specific HTTP request for use in downloading." '318 patent, claim 1. The Hulu application is a playback engine implemented on a playback device. The Hulu application plays back video content based on a user's instructions.[70] The Hulu application downloads an index from a remote video file: the segment index, or 'sidx,' box. The 'sidx' box "provides a

---

[65] http://www.streamingmediaglobal.com/Articles/ReadArticle.aspx?ArticleID=96144&PageNum=1.

[66] ISO/IEC 23009-1 *Information technology – Dynamic adaptive streaming over HTTP (DASH) – Part 1: Media presentation description and segment formats*, at 85, 87 (2d ed. 2014) ("ISO/IEC 23009-1" or "ISO DASH Standard").

[67] ISO/IEC 14496-12 *Information technology — Coding of audio-visual objects — Part 12: ISO base media file format*, at 4, 20 (5th ed. 2015) ("ISO/IEC 14496-12" or "ISO BMFF Standard").

[68] ISO/IEC 14496-12 at 56.

[69] ISO/IEC 14496-12 at 3-5, 32 (defining "chunk," "track," and "sample").

[70] https://play.google.com/store/apps/details?id=com.hulu.plus&hl=en_US.

compact index of one media stream within the media segment to which it applies" that "documents how a []segment is divided into one or more subsegments." A "segment" is a "portion of an ISO base media file format file, consisting of either (a) a movie box, with its associated media data (if any) and other associated boxes or (b) one or more movie fragment boxes ['moof' boxes], with their associated media data ['mdat' boxes], and other associated boxes." A "subsegment" is "a time interval of the containing []segment . . . correspond[ing] to a single range of bytes of the containing []segment."[71] In short, the 'sidx' box references byte ranges for a video segment—'moof' box + 'mdat' box pairs—within a video file. The 'sidx' box contains reference elements that provide size information for subsegments / byte ranges. The 'sidx' includes size information that allows the device to identify the location of each 'moof' box + 'mdat' box pair and the beginning of each 'mdat' box. Thus, the 'sidx' box reference elements are chunk references. The Hulu application implemented on a playback device requests and receives the 'sidx' box to play back video from Hulu's servers and uses the 'sidx' box to convert a chunk reference into a specific HTTP request (byte range request) for use in downloading.

227. Hulu "request[s] chunks for downloading based upon a received instruction and maintain[s] a queue of the requested chunks, using the playback engine implemented on the playback device, wherein the requested chunks are downloaded by identifying media chunks corresponding to a sequence of key frames identified for the received instruction." '318 patent, claim 1. The Hulu application requests chunks for downloading as needed as playback progresses, based on the user's instructions. For example, the Hulu application requests non-sequential chunks for downloading if the Hulu application user seeks ahead during playback. The Hulu application requests chunks, using byte ranges, from the remote server, based on that seeking instruction. Then the Hulu application maintains a

[71] ISO/IEC 14496-12 at 5, 105-06, 228.

queue of the requested chunks. For example, when the Hulu application requests chunks, or byte ranges, the okhttp implements asynchronous HTTP requests and uses a "dispatch" process to maintain a request queue.[72] The Dispatcher class realizes the okhttp3 "dispatch" process.[73] And the Hulu application uses Dispatcher class for OkHttpClient class. The OkHttpClient.Builder function is used to create PlaybackHttpClient, which contains a Dispatcher method so that the system maintains a queue of requested chunks (i.e., HTTP requests to the Hulu remote servers). The system also uses the Dispatcher method if it needs to cancel a queued call. Cancelling a queued call further indicates that the Hulu application maintains a queue of requested chunks, using okhttp3. The Hulu application requests and downloads chunks by identifying media chunks corresponding to a sequence of key frames identified for the received instruction. The Hulu application uses the 'sidx' box to request 'mdat' boxes and their associated 'moof' boxes, and each 'mdat' box begins with an I frame, otherwise known as a "key frame."[74]

228. Hulu "receive[s] a requested chunk and remove[s] the received chunk from the queue of requested chunks, using the playback engine implemented on the playback device." '318 patent, claim 1. For example, the Hulu application requests a chunk (in the form of a byte range) as a HTTP GET request and queues the requested chunks using the Dispatcher class with asynchronous calls. The Hulu application will receive the requested chunk within a HTTP response. The Dispatcher class in okhttp3 is designed to remove the queued requested chunks

[72] https://square.github.io/okhttp/calls/.

[73] https://square.github.io/okhttp/4.x/okhttp/okhttp3/-dispatcher/.

[74] http://www.streamingmediaglobal.com/Articles/ReadArticle.aspx?ArticleID=96144&PageNum=1; *Guidelines for Implementation: DASH-IF Interoperability Points*, DASH Industry Forum, Ver. 4.3 (Nov. 15, 2018), at 12.

once the corresponding HTTP response is received. The "finished" method used in the Dispatcher class on the okhttp3 Github site indicates that once the current request is removed from the queue of requested chunks, this.idleCallback will update the value of idleCallback to indicate if the Dispatcher is in idle status and is ready to process the next request.[75] A similar function exists in the Dispatcher function contained within the Hulu application. The way to remove the requested chunk from the queue of requested chunks is implemented in the Hulu application the same manner as the Github source code.

229. Hulu "maintain[s] a playback queue of received chunks for playback pending commencement of playback by the playback device, using the playback engine implemented on the playback device." '318 patent, claim 1. For example, when the Hulu application implemented on a playback device requests chunks, or byte ranges, the playback device maintains a queue of the received chunks. The Response class extracts the details of the received video chunks from the HTTP response to the Hulu application's HTTP request. The MediaBytes method then uses the Response in FragmentDataSourcePullerUtil to store (i.e., download) the video chunks. The MdatBox class uses MediaBytes to transfer the information related to the video chunks to the MdatBox. The downloadMdatData class uses the MdatBox to track the size and duration of the downloaded video chunks (contained in the MdatBox). MdatBox.writePos tracks the video chunks in a first buffer. The MdatBox.writePos value determines how the video chunks queue in a second buffer, the DecoderInputBufferHolder. The buildFragment method in the Fragment class transfers data stored in MdatBox.getData() to Fragment.mData. The value stored in Fragment.mData is further stored in DecoderInputBufferHolder.data. The

[75] https://github.com/square/okhttp/blob/master/okhttp/src/main/kotlin/okhttp3/Dispatcher.kt; https://square.github.io/okhttp/4.x/okhttp/okhttp3/-dispatcher/idle-callback/.

source code queues that data in the DecoderInputBufferHolder buffer. The DecoderInputBufferHolder is a queue of the received video chunks for playback.

230. Hulu "provide[s] chunks maintained in the playback queue to a decoder to enable playing of the media by the playback device, using the playback engine implemented on the playback device." '318 patent, claim 1. The DecoderInputBuffer queues chunks for playback and feeds them to the decoder.[76] The Hulu application uses "queueInputBuffer" to move input data into the DecoderInputBuffer queue.[77] The Hulu application then plays the video on the device.

231. Hulu has infringed, and continues to infringe, at least claim 1 of the '318 patent in the United States by making, using, offering for sale, selling, and/or importing the Hulu application and service in violation of 35 U.S.C. § 271(a).

232. At least as of the date of this Complaint, Hulu knows that it provides and specifically intends to provide an application and service for CE playback devices that, when used as intended, meets the limitations of claim 1, as described in ¶¶ 225-230. Hulu therefore has induced, and continues to induce, infringement of at least claim 1 of the '318 patent in violation of 35 U.S.C. § 271(b) in the exemplary manner described herein.

233. Hulu's infringement has caused and continues to cause damage to DivX, and DivX is entitled to recover damages sustained as a result of Hulu's wrongful acts in an amount subject to proof at trial.

[76] https://developer.android.com/reference/android/media/MediaCodec.

[77] https://developer.android.com/reference/android/media/MediaCodec#queueInputBuffer(int,%20int,%20int,%20long,%20int).

COUNT III: INFRINGEMENT OF U.S. PATENT NO. 10,412,141

234. The allegations of ¶¶ 1-233 of this Complaint are incorporated by reference as though fully set forth herein.

235. Hulu directly infringes at least claim 1 of the '141 patent under 35 U.S.C. § 271(a), as set forth below in ¶¶ 236-247.

236. Hulu, operating on a compatible consumer electronics device, provides a playback device,[78] comprising "a processor and a non-volatile storage containing an application for causing the processor to perform" the steps as claimed in the '141 patent. '141 patent, claim 1.

237. Hulu causes the processor to perform the step of "establishing at least one connection for communicating with a remote server system." '141 patent, claim 1. Hulu provides a streaming library of more than 80,000 TV episodes and movies, which Hulu encodes prior to distribution through Hulu's video players and applications.[79] The Hulu application establishes at least one connection for communication with a remote server system to access Hulu's streaming library, including, for example, manifest-dp.hulustream.com, http-a-darwin.hulustream.com, http-e-darwin.hulustream.com, http-v-darwin.hulustream.com, ads-v-darwin.hulustream.com, and play.hulu.com.[80]

238. Hulu causes the processor to perform the step of "obtaining information from a remote server system describing at least one video track, multiple audio tracks, and multiple subtitle tracks." '141 patent, claim 1. Hulu uses the MPEG-DASH streaming protocol to deliver content.[81] The MPEG-DASH

[78] https://help.hulu.com/s/article/supported-devices?language=en_US.

[79] https://www.hulu.com/welcome; https://advertising.hulu.com/dom/.

[80] https://www.slideshare.net/AmazonWebServices/case-study-how-hulu-reinvented-television-using-the-aws-cloud-ctd302-reinvent-2017.

[81] http://www.streamingmediaglobal.com/Articles/Editorial/Featured-

streaming protocol defines "several encoded versions" for each media content component.[82] For example, the Hulu application obtains from a Hulu remote server system manifest-dp.hulustream.com, a manifest describing each alternative video track, multiple audio tracks, and multiple subtitle tracks.

239. Hulu causes the processor to perform the step of "selecting a video track from the at least one video track." '141 patent, claim 1. The Hulu application, for example, selects and requests from a remote server system, such as http-e-darwin.hulustream.com, a particular encoded mp4 video track. The Hulu application can select, for example an H.264 video track and request segments of the selected video track.[83]

240. Hulu causes the processor to perform the step of "requesting a header describing the selected video track." '141 patent, claim 1. Hulu encodes, packages, and streams to its applications video files, for example, mp4, fmp4, or other container files, that contain headers describing the selected video track.[84] An mp4

Articles/Hulus-Move-to--DASH-105110.aspx.

[82] ISO/IEC 23009-1, Information technology – Dynamic adaptive streaming over HTTP (DASH) – Part 1: Media presentation description and segment formats, at 9 (2d ed. 2014) ("ISO/IEC 23009-1").

[83] ISO/IEC 23009-1 at 9.

[84] http://www.streamingmediaglobal.com/Articles/ReadArticle.aspx?ArticleID=96144&PageNum=1; ISO/IEC 23001-7 Information technology – MPEG systems technologies – Part 7: Common encryption in ISO base media file format files (3d ed. 2016); Pantos, R., HTTP Live Streaming (2d ed. Sept. 23, 2019), Apple Inc. (https://tools.ietf.org/pdf/draft-pantos-hls-rfc8216bis-05.pdf); ISO/IEC 14496-12 Information technology – Coding of audio-visual objects – Part 12: ISO base media file format (2d ed. (Corrected) 2005) (hereinafter "ISO BMFF Specification"); The Protected Interoperable File Format (PIFF), Ver. 1.1 (2010) (hereinafter "Microsoft PIFF Specification").

file, for example, contains a movie header box, among other header boxes.[85] The Hulu application selects a particular encoded video track and the video's header box, such as the movie header ('mvhd'), track header ('tkhd'), or media header ('mdhd').[86]

241. Hulu causes the processor to perform the step of "selecting an audio track from the multiple audio tracks." '141 patent, claim 1. The Hulu application, after obtaining information from a Hulu remote server system, such as http-e-darwin.hulustream.com, selects an audio track from multiple available audio tracks.[87]

242. Hulu causes the processor to perform the step of "obtaining index information indicating the locations of audio and video data within the selected audio and video tracks." '141 patent, claim 1. Hulu encodes video files as, for example, mp4 container files, that contain alternative encoded streams of content divided into segments.[88] For mp4 files, the Hulu application obtains from a Hulu remote server, such as http-e-darwin.hulustream.com, a top level index file and parses the top level index file to obtain information, such as mp4 segment index boxes ('sidx'), indicating the location of audio and video data within the selected audio and video tracks.

243. Hulu causes the processor to perform the steps of "determining byte ranges to request from the selected audio and video tracks using the index information" and "requesting byte ranges from the selected video track and the selected audio track from the remote server system." '141 patent, claim 1. For example, for mp4 files, the Hulu application parses and analyzes the 'sidx' box

[85] ISO IEC 14496-12 at 11, 15.

[86] ISO IEC 14496-12 at 11, 15; Microsoft PIFF Specification, at 9.

[87] ISO/IEC 23009-1 at 9.

[88] ISO/IEC 23009-1 at 10; ISO/IEC 14496-12 at 105.

information to determine the byte ranges from the selected video track and the selected audio track that need to be requested from the remote server system for playback.[89] The 'sidx' boxes for a selected video track and audio track include reference elements containing size information for each segment within the selected track. Using the size information in the 'sidx' box, the Hulu application determines the byte range to request in order to request a given segment of audio or video from the remote server. The Hulu application requests byte ranges from the selected audio and video track as an HTTP GET request from, for example, Hulu's remote server http-e-darwin.hulustream.com.

244. Hulu causes the processor to perform the step of "buffering received bytes of information comprising audio and video data." '141 patent, claim 1. For example, the Hulu application inputs the received bytes of information comprising audio and video data into a "chunkBuffer." Using the function "iMediaExtractor," and "mVideoExtractor" and "mAudioExtractor," the Hulu application extracts the video track and audio track data which will be read into the "ChunkSampleSource" function. The "chunkBuffer" is used by the "ChunkSampleSource" function to buffer the downloaded video and audio track segments.

245. Hulu causes the processor to perform the steps of "checking that sufficient data is buffered to commence playback and playing back the buffered audio and video data" and "responding to a received seek instruction by: pausing playback." '141 patent, claim 1.For example, the Hulu application uses a "minBufferTime" function to determine whether sufficient data has buffered to commence playback. The Hulu application subsequently plays back the buffered audio and video data on a playback device. The Hulu application, upon receiving a seek instruction from an end user, pauses playback using a "seekToInternal" and "renderer.stop ()" function.

[89] ISO/IEC 14496-12 at 105, 228.

246. Hulu causes the processor to perform the steps of "determining byte ranges to request from the selected audio and video tracks based upon a new playback location using the index information" and "requesting byte ranges required to play the selected audio and video tracks from the new playback location from the remote server." '141 patent, claim 1. For example, the Hulu application responds to a seek instruction from the end user by using the size information from the 'sidx' boxes of a selected audio and video track to determine the byte ranges to request based on the new playback location. The Hulu application requests byte ranges for the new playback location from the selected audio and video track as an HTTP GET request from, for example, Hulu's remote server http-e-darwin.hulustream.com.

247. Hulu causes the processor to perform the step of "buffering received bytes of information comprising audio and video data pending resumption of playback" and "checking that sufficient data is buffered to commence playback and playing back the buffered audio and video data." '141 patent, claim 1. For example, the Hulu application inputs the received bytes of information comprising audio and video data into a "chunkBuffer." Using the functions "iMediaExtractor," "mVideoExtractor," and "mAudioExtractor," the Hulu application extracts the video track and audio track data which will be read into the "ChunkSampleSource" function. The "chunkBuffer" is used by the "ChunkSampleSource" function to buffer the downloaded video and audio track segments pending resumption of playback. The Hulu application then uses a "minBufferTime" function to determine whether sufficient data has buffered to commence playback. The Hulu application subsequently plays back the buffered audio and video data on a playback device.

248. Hulu has infringed, and continues to infringe, at least claim 1 of the '141 patent in the United States by making, using, offering for sale, selling, and/or importing the Hulu application and service in violation of 35 U.S.C. § 271(a).

249. At least as of the date of this Complaint, Hulu knows that it provides and specifically intends to provide an application and service for CE playback devices that, when used as intended, meets the limitations of claim 1, as described in ¶¶ 236-247. Hulu therefore has induced, and continues to induce, infringement of at least claim 1 of the '141 patent in violation of 35 U.S.C. § 271(b) in the exemplary manner described herein.

250. Hulu's infringement has caused and continues to cause damage to DivX, and DivX is entitled to recover damages sustained as a result of Hulu's wrongful acts in an amount subject to proof at trial.

COUNT IV: INFRINGEMENT OF U.S. PATENT NO. 10,542,061

251. The allegations of ¶¶ 1-250 of this Complaint are incorporated by reference as though fully set forth herein.

252. Hulu directly infringes at least claim 1 of the '061 patent under 35 U.S.C. § 271(a), as set forth below in ¶¶ 253-270.

253. Hulu performs "[a] method for streaming a piece of content using a set of server computer systems." '061 patent, claim 1. For example, Hulu provides a streaming library of more than 80,000 TV episodes and movies.[90] Hulu's video streaming platform uses a set of server computer systems, including but not limited to hulustream.com, play.hulu.com, and vortex.hulu.com.

254. Hulu "obtain[s] common cryptographic information associated with a piece of content using an encrypting system." '061 patent, claim 1. For example, Hulu uses an encrypting system that utilizes the Common Encryption Scheme (CENC) to protect its content.[91] The CENC implements a key identifier (KID) and

[90] https://www.hulu.com/welcome.

[91] http://www.streamingmediaglobal.com/Articles/ReadArticle.aspx?ArticleID=96144&PageNum=1.

cryptographic information associated with a piece of content to be used in the encryption or decryption of the frames of the video content.[92] Content from the same container file (i.e., mp4 file) shares the same KID value.

255. Hulu "encod[es] multiple streams of video content associated with a piece of content using an encoding system." '061 patent, claim 1. For example, Hulu uses the MPEG-DASH streaming protocol to deliver content.[93] The MPEG-DASH streaming protocol defines "several encoded versions" for each media content component.[94] "Regardless of the origin of the content, Hulu maintains control of its streaming digital video players and content pipelines, and encodes all video content and advertisements prior to distribution within Hulu's proprietary video players on Web and within Hulu applications."[95]

256. Hulu "encrypt[s] the encoded multiple streams of video content into multiple streams of protected video content using the common cryptographic information using the encrypting system." '061 patent, claim 1. For example, Hulu uses the MPEG-DASH streaming protocol to deliver content.[96] The MPEG-DASH streaming protocol defines "several encoded versions," or "media streams," for each media content component.[97] Each media file (i.e., Representation) contains

[92] ISO/IEC 23001-7, *Information technology — MPEG systems technologies — Part 7: Common encryption in ISO base media file format files*, Third Edition, at 10 (Feb. 15, 2016) ("ISO/IEC 23001-7:2016").

[93] http://www.streamingmediaglobal.com/Articles/Editorial/Featured-Articles/Hulus-Move-to--DASH-105110.aspx.

[94] ISO/IEC 23009-1, *Information technology – Dynamic adaptive streaming over HTTP (DASH) – Part 1: Media presentation description and segment formats*, at 9 (2d ed. 2014) ("ISO/IEC 23009-1:2014").

[95] https://advertising.hulu.com/dom/ (copyright 2018 Hulu, LLC).

[96] http://www.streamingmediaglobal.com/Articles/ReadArticle.aspx?ArticleID=96144&PageNum=1.

[97] ISO/IEC 23009-1:2014 at 9.

"one or more media streams."[98] Hulu also uses an encrypting system that includes the Common Encryption Scheme (CENC) to protect its content.[99] The CENC implements a key identifier (KID) and cryptographic information associated with a piece of content to be used in the encryption or decryption of the frames of the video content.[100] Content from the same container file (i.e., mp4 file) shares the same KID value.

257. Hulu "stor[es] the multiple streams of protected video content in memory at a server system." '061 patent, claim 1. For example, Hulu's video streaming platform uses a set of server computer systems, including but not limited to hulustream.com, play.hulu.com, and vortex.hulu.com. The hulustream.com server contains multiple streams of protected video content. The video content stored on Hulu's server computer systems contains multiple streams according to the MPEG-DASH streaming protocol.[101] The video content stored on Hulu's server computer systems is protected using the CENC.[102] The ContentProtection value for Hulu's video content is "cenc," further indicating that Hulu's video content is protected according to CENC.

258. Hulu "stor[es] the common cryptographic information at a digital rights management system." '061 patent, claim 1. For example, Hulu uses the

---

[98] ISO/IEC 23009-1:2014 at 9, 34.

[99] http://www.streamingmediaglobal.com/Articles/ReadArticle.aspx?ArticleID=96144&PageNum=1.

[100] ISO/IEC 23001-7, *Information technology — MPEG systems technologies — Part 7: Common encryption in ISO base media file format files*, Third Edition, at 10 (Feb. 15, 2016) ("ISO/IEC 23001-7:2016").

[101] http://www.streamingmediaglobal.com/Articles/ReadArticle.aspx?ArticleID=96144&PageNum=1; ISO/IEC 23009-1:2014 at 9.

[102] http://www.streamingmediaglobal.com/Articles/ReadArticle.aspx?ArticleID=96144&PageNum=1.

MPEG-DASH streaming protocol to deliver content. In some instances, Hulu also uses an encrypting system that includes Microsoft PlayReady DRM to protect its content.[103] Microsoft PlayReady supports CENC and stores the common cryptographic information (*e.g.*, the KID values) in its DRM system.[104] Further, the request for a DRM license during streaming using the Hulu application indicates that the common cryptographic information is stored on play.hulu.com, the DRM server.

259. Hulu "receiv[es] a request for a top level index file from a playback device at the server system, where the request identifies the piece of content and includes information describing the playback device." '061 patent, claim 1. The playback device may be, for example, a Hulu supported device such as the Motorola G7.[105] Hulu's server system receives a request for a top level index file, or manifest, from a playback device. The request identifies a piece of content, for example, by Hulu movieId. The request for content also includes information describing the playback device such as "version" and "device" numbers and "User-Agent." The playback device also sends requests to the server before the manifest request that describes information about the playback device.

260. Hulu "automatically filter[s] the multiple streams of protected video content into a plurality of alternative streams of protected video content based upon the received information describing the playback device using the server

---

[103] http://www.streamingmediaglobal.com/Articles/ReadArticle.aspx?ArticleID=96144&PageNum=1.

[104] Microsoft PlayReady, *Developing PlayReady Clients*, at 17 (April 2015) ("Developing PlayReady Clients"); Microsoft Corporation, *DASH Content Protection using Microsoft PlayReady: Implementing Content Protection for Live and On-Demand Profiles of Dynamic Adaptive Streaming over HTTP (ISO/IEC 23009-1) using Microsoft PlayReady*, at 2, 8 (February 2013) ("Microsoft PlayReady for DASH").

[105] https://help.hulu.com/s/article/watching-hulu-on-site?language=en_US; https://help.hulu.com/s/article/supported-devices?language=en_US.

system." '061 patent, claim 1. For example, different playback devices receive different video streams (i.e., filtered video streams) from the Hulu playback server system after requesting the same content based on information describing the playback device, as evidenced by different manifests obtained during playback. The Hulu server computer systems have a list of different streams associated with a particular piece of content. This list is different for different device types, indicating that the Hulu server retrieves a list of video streams and the list has been filtered based on information describing the playback device. Further, Hulu represents that its playback server system maintains a database (i.e., storage) of manifest attributes for manifest generation, indicating that the playback device information is stored on Hulu's server.[106]

261. Hulu "automatically generat[es] a top level index file in response to the request for a top level index file from the playback device using the server system." '061 patent, claim 1. For example, the Hulu server computer systems generate a DASH-compliant manifest (.mpd file) describing each video stream in the filtered list of video streams, which includes the location of the video file and its bitrate, as demonstrated by the exemplary MPD files and manifests sent during streaming of Hulu content on playback devices. Hulu indicates that it generates a "manifest with pre-set content and ads for each stream" on the server side.[107] A presentation by Hulu indicates that Hulu automatically generates top level index files, *i.e.*, manifests, "on-the-fly."[108] Hulu also uses an encoding system to encode content, as demonstrated by Hulu's live streaming service architecture, which

---

[106] *See* https://www.slideshare.net/AmazonWebServices/case-study-how-hulu-reinvented-television-using-the-aws-cloud-ctd302-reinvent-2017.

[107] https://advertising.hulu.com/dom/.

[108] *See* https://www.slideshare.net/AmazonWebServices/case-study-how-hulu-reinvented-television-using-the-aws-cloud-ctd302-reinvent-2017.

shares a "substantial part" of its architecture with its video-on-demand (VOD) service.[109]

262. Hulu automatically generates a top level index file in response to the request for a top level index file from the playback device using the server system, where the top level index file "identifies the location of each of the plurality of alternative streams of protected video content, where each of the plurality of alternative streams of protected video content encodes the piece of content at a different bitrate." '061 patent, claim 1. When a playback device requests a manifest file, the device performs a GET request to the Hulu manifest server (part of the Hulu server computer systems). In response to the GET Request, Hulu's manifest server returns the manifest (in some cases, an MPD file) to the playback device. This manifest file is then used to determine which video streams to request for playback.

263. Hulu automatically generates a top level index file in response to the request for a top level index file from the playback device using the server system, where the top level index file "describes at least a bitrate of each of a plurality of alternative streams of protected video content associated with the identified piece of content." '061 patent, claim 1. Hulu generates a top level index file—*e.g.*, a DASH-compliant manifest (.mpd file)—that describes at least a bitrate of each asset in the filtered list of video streams and identifies locations of the video streams in the filtered list of video streams. For example, the top level index file that each device receives from Hulu identifies the locations (BaseURL) and bitrates for different alternative streams.

[109] https://advertising.hulu.com/dom/ (copyright 2018 Hulu, LLC); https://medium.com/hulu-tech-blog/the-anatomy-of-a-live-ott-service-c8f6078b24d3; https://medium.com/hulu-tech-blog/introducing-the-hulu-technical-landscape-93f4c136c568.

264. Hulu automatically generates a top level index file in response to the request for a top level index file from the playback device using the server system, where the top level index file "includes a reference to the common cryptographic information for accessing the protected video content." '061 patent, claim 1. The Hulu application receives a top level index (i.e., an MPD file). The top-level index identifies common cryptographic information (e.g., KID) for accessing the protected video content for a plurality of alternative streams (i.e., Representations) of video. The presence of a TrackEncryptionBox in each representation indicates that these video frames are encrypted. Content from the same container file (i.e., mp4 file) shares the same KID value and the KID is referenced in the top-level index (i.e., MPD file).

265. Hulu "send[s] the top level index file to the playback device from the server system." '061 patent, claim 1. In response to the GET Request, Hulu's manifest server (e.g., manifest-dp.hulustream.com) returns the manifest (in this case, an MPD file) to the playback device.

266. Hulu "receiv[es] at the server system a request for at least a portion of an initial stream of protected video content, selected from the plurality of alternative streams of protected video content, from the playback device." '061 patent, claim 1. In some instances, the playback device uses the manifest file to determine which video streams to request for playback. For example, the Hulu Player operating on a playback devices receives a manifest specifying locations (BaseURL) for portions of a video stream. Using the specified locations, the playback device requests the header information and actual video content, identified as byte ranges.

267. Hulu "sends the requested at least a portion of the initial stream of protected video content to the playback device from the server system." '061 patent, claim 1. For example, the Hulu Player operating on a playback device receives header information and actual video content, identified as byte ranges.

268. Hulu "receiv[es] at the digital rights management system a request from the playback device for common cryptographic information to access the initial stream of protected video content." '061 patent, claim 1. The Hulu application issues a request to Hulu's DRM server. For example, the request for a DRM license during streaming using the Hulu application shows an encrypted block of data, indicated by the presence of a content decryption module (CDM). The request to Hulu's DRM server requests cryptographic information. Where the DRM server is PlayReady-based, the Hulu application obtains the key to unlock the encrypted media from the PlayReady License Server.[110] Content from the same container file (i.e., mp4 file) shares the same KID value.

269. Hulu "encrypt[s] the common cryptographic information using the digital rights management system." '061 patent, claim 1. For example, the Hulu application obtains encrypted keys from a DRM server, indicating that the DRM server encrypts the common cryptographic information (i.e., keys).

270. Hulu "send[s] the encrypted common cryptographic information to the playback device from the digital rights management system." '061 patent, claim 1. For example, the Hulu application obtains encrypted keys from a content decryption module. The request for a DRM license during streaming using the Hulu application shows an encrypted block of data (signature). The response for the DRM license during streaming using the Hulu application also shows an encrypted block of data. In some instances, Hulu's DRM server is PlayReady-based, and the Hulu application obtains the key to unlock the encrypted media from the PlayReady License Server.[111]

[110] https://docs.microsoft.com/en-us/playready/overview/license-server.

[111] https://docs.microsoft.com/en-us/playready/overview/license-server.

271. Hulu has infringed, and continues to infringe, at least claim 1 of the '318 patent in the United States by making, using, offering for sale, selling, and/or importing the Hulu application and service in violation of 35 U.S.C. § 271(a).

272. At least as of the date of this Complaint, Hulu knows that it provides and specifically intends to provide an application and service for CE playback devices that, when used as intended, meets the limitations of claim 1, as described in ¶¶ 253-270. Hulu therefore has induced, and continues to induce, infringement of at least claim 1 of the '061 patent in violation of 35 U.S.C. § 271(b) in the exemplary manner described herein.

273. Hulu's infringement has caused and continues to cause damage to DivX, and DivX is entitled to recover damages sustained as a result of Hulu's wrongful acts in an amount subject to proof at trial.

COUNT V: INFRINGEMENT OF U.S. PATENT NO. 10,326,987

274. The allegations of ¶¶ 1-273 of this Complaint are incorporated by reference as though fully set forth herein.

275. Hulu directly infringes at least claim 1 of the '987 patent under 35 U.S.C. § 271(a), as set forth below in ¶¶ 276-294.

276. Hulu's streaming service and application, operating on a compatible consumer electronics device, provides "[a] playback device for playing content from a plurality of alternative streams, the playback device comprising."[112] '987 patent, claim 1. For example, Hulu uses the MPEG-DASH streaming protocol to deliver content.[113] The MPEG-DASH streaming protocol is a protocol for performing adaptive bitrate streaming of media including a set of alternative video streams. The MPEG-DASH streaming protocol provides "several encoded

[112] https://help.hulu.com/s/article/supported-computers?language=en_US.

[113] http://www.streamingmediaglobal.com/Articles/Editorial/Featured-Articles/Hulus-Move-to--DASH-105110.aspx.

versions" for each media content component.[114] Each media file (e.g., Adaptation Set) contains Representations, which contain "one or more media streams."[115]

277. Hulu causes Hulu-compatible playback devices—which contain "a set of one or more processors" —to execute the software operations specified by Hulu. '987 patent, claim 1.

278. Hulu causes Hulu-compatible playback devices—which contain "a non-volatile storage containing an application"—to store the Hulu application. '987 patent, claim 1. The Hulu application installed on a playback device is "an application for causing the set of one or more processors to perform" the recited steps.

279. Hulu causes the processor to perform the step of "obtaining a top level index file identifying a plurality of alternative video streams." '987 patent, claim 1. For example, the MPEG-DASH streaming protocol requires each media file (e.g., Adaptation Set) to include Representations, which contain "one or more media streams."[116] The Media Presentation Description (MPD) is "a document that contains metadata required by a DASH Client to construct appropriate HTTP-URLs to access Segments and to provide the streaming service to the user."[117] The MPD is therefore a top level index file identifying a plurality of alternative video streams. The MPD received during the streaming of Hulu content includes alternative streams, as indicated by the "Representation id" variable in the MPD.

280. Hulu causes the processor to perform the step of "specifying a maximum bitrate for each of the plurality of alternative video streams." '987 patent,

[114] ISO/IEC 23009-1, *Information technology – Dynamic adaptive streaming over HTTP (DASH) – Part 1: Media presentation description and segment formats*, Second Edition, at 9 (May 15, 2014) ("ISO/IEC 23009-1:2014").

[115] ISO/IEC 23009-1:2014 at 10.

[116] ISO/IEC 23009-1:2014 at 10.

[117] ISO/IEC 23009-1:2014 at 9, 16.

claim 1. The MPEG-DASH streaming protocol provides that each video stream (*i.e.*, Representation) specify a maximum bandwidth. Hulu uses the MPEG-DASH streaming protocol and therefore specifies a maximum bitrate for each of the plurality of alternative video streams. For example, the MPD received during the streaming of Hulu content includes alternative streams and a maximum bitrate for each of the plurality of alternative video stream, as indicated by the "bandwidth" variable in the MPD.

281. Hulu causes the processor to perform the step of specifying a maximum bitrate for each of the plurality of alternative video streams, "where the plurality of alternative video streams comprises a first and a second alternative video stream." '987 patent, claim 1. For example, the MPD received during the streaming of Hulu content includes two video streams with different "Representation id" values.

282. Hulu causes the processor to perform the step of "during an initial startup period: obtaining at least one network data rate measurement." '987 patent, claim 1. For example, Hulu allows the playback device to measure the channel data rate of the network connection (e.g., bandwidth or bitrate).[118] For example, for the initial bitstream, when the Hulu application requests an MPD for Hulu content, the request includes the desirable bandwidth at the playback device, indicating that the playback device measures and reports channel data rate. Further, Hulu uses the Javascript Player, which contains source code for estimating network bandwidth during streaming. During startup, for example, Hulu assigns the value for "estimateBandwidth" as the value stored in "initBandwidth." The presence of "initBandwidth" and "estimateBandwidth" values during startup

[118] https://help.hulu.com/s/article/speed-recommendations?language=en_US; ISO/IEC 23009-1:2014 at 108.

("prototype.setup") indicates that Hulu obtains at least one network data rate measurement at startup.

283. Hulu causes the processor to perform the step of, during an initial startup period, "selecting the first alternative video stream based upon a comparison between the specified maximum bitrates for each of the plurality of streams and the at least one network data rate measurement." '987 patent, claim 1. For example, Hulu uses the MPEG-DASH streaming protocol and therefore selects a first video stream (*i.e.*, a Representation) from a set of alternative video streams (i.e., an Adaptation Set) provided by the Hulu server using a playback device.[119] This stream selection is a result based on a comparison of the specified maximum bitrates for each of the plurality of streams and the at least one network data rate measurement. For example, the Hulu Javascript Player selects the desired video stream ("selectAlgorithmQuality") based on "estimatedBandwidth" derived from the network data rate measurement "(this._RTTcoefficient - i) / this._RTTcoefficient." The Hulu Javascript Player also compares each video stream's specified maximum bitrate (represented by the "safeBandwidth" value) with the network data rate measurement "(this._RTTcoefficient - i) / this._RTTcoefficient." Based on this comparison, Hulu selects the video stream with "ProfileBandwidth" of 400, because "it is the maximum one with the profile: 400 bitrate: 476707 safeBandwidth: 698 that is small than the estimated bandwidth: 708."

284. Hulu causes the processor to perform the step of, during an initial startup period, "requesting at least one chunk of the first alternative video stream." '987 patent, claim 1. For example, for the selected video stream with "ProfileBandwidth" of 400, Hulu first requests the header information and then the actual video content, in byte ranges, for the same video stream.

[119] ISO/IEC 23009-1:2014 at 108.

285. Hulu causes the processor to perform the step of, during an initial startup period, "storing the at least one chunk of the first alternative video stream in a buffer of the playback device." '987 patent, claim 1. For example, Hulu uses the MPEG-DASH streaming protocol and therefore stores the requested chunks (e.g., entire Segments or byte ranges of Segments) in a buffer of the playback device.[120] Further, Hulu uses the Javascript Player, which tracks video segments saved in the buffer through the "videoBufferLength" variable, indicating that Hulu downloads the video segment (i.e., chunk of the video) and stores it in a buffer.

286. Hulu causes the processor to perform the step of, during an initial startup period, "playing back at least one chunk of the first alternative stream stored in the buffer." '987 patent, claim 1. For example, Hulu uses the MPEG-DASH streaming protocol and therefore allows the playback device to play back the buffered chunks (e.g., entire Segment or byte ranges of Segments).[121]

287. Hulu causes the processor to perform the step of, during an initial startup period, "obtaining at least one additional network data rate measurement." '987 patent, claim 1. For example, Hulu allows the playback device to continuously measure the channel data rate of the network connection (e.g., bandwidth or bitrate).[122] Further, Hulu uses the Javascript Player, which obtains at least one additional network data rate measurement using "updateNetworkTrackerLog."

288. Hulu causes the processor to cause the set of one or more processors to perform the step of "determining that the network data rate is greater than the specified maximum bitrate for the second alternative video stream based upon the

[120] ISO/IEC 23009-1:2014 at 10, 108.

[121] ISO/IEC 23009-1:2014 at 10, 109.

[122] https://help.hulu.com/s/article/speed-recommendations?language=en_US; ISO/IEC 23009-1:2014 at 108.

at least one additional network data rate measurement." '987 patent, claim 1. For example, Hulu uses the Javascript Player, which selects the desired video stream ("selectAlgorithmQuality") based on a determination that the newly measured network data rate value ("estimated bandwidth") is greater than the "safeBandwidth" value for the second alternative video stream.

289. Hulu causes the processor to perform the step of "when the network data rate is determined to be greater than the specified maximum bitrate for the second alternative video stream, requesting at least one chunk of the second alternative video stream." '987 patent, claim 1. For example, after determining that this network data rate is greater than the specified maximum bitrate for the second alternative video stream Hulu first requests the header information and then the actual video content, in byte ranges, for the same video stream.

290. Hulu causes the processor to perform the step of "when a minimum buffer level criterion is satisfied based upon a playback duration of chunks of video content stored in the buffer of the playback device: obtaining at least one further network data rate measurement." '987 patent, claim 1. For example, Hulu requires the implementation of minBufferTime, which "specifies a common duration" so that "a client can be assured of having enough data for continuous playout providing playout begins after @minBufferTime @bandwidth bits have been received."[123] Further, Hulu allows the playback device to continuously measure the channel data rate of the network connection (e.g., bandwidth or bitrate).[124] Hulu uses the Javascript Player, which contains source code for estimating network bandwidth during streaming. For example, at least one version of the "prototype.existCdnPerformanceIssue" function – specifically, "Te.prototype.existCdnPerformanceIssue" – determines when "buffer is in danger,

[123] ISO/IEC 23009-1:2014 at 18-19, 35, 108-109.

[124] https://help.hulu.com/s/article/speed-recommendations?language=en_US.

bandwidth is lower than the threshold," taking into account network data rate ("estimateBandwidth").

291. Hulu causes the processor to perform the step of, when a minimum buffer level criterion is satisfied based upon a playback duration of chunks of video content stored in the buffer of the playback device, "selecting a stream from the plurality of alternative video streams based upon a playback duration of chunks of video content stored in the buffer of the playback device by selecting a stream from the plurality of alternative video streams such that the playback duration of chunks of video content stored in the buffer of the playback device is sufficient to prevent buffer underflow during downloading and playback of at least one chunk of the selected video stream based upon the at least one further network data rate measurement." '987 patent, claim 1. For example, Hulu requires the implementation of minBufferTime, which "specifies a common duration" so that "a client can be assured of having enough data for continuous playout providing playout begins after @minBufferTime @bandwidth bits have been received."[125] Hulu uses the Javascript Player, which implements a function to determine buffer underflow. For example, the "Te.prototype.existCdnPerformanceIssue" function determines when "buffer is in danger, bandwidth is lower than the threshold," taking into account network data rate ("estimateBandwidth") and buffer ("bufferTrackerByType"). When "buffer is in danger, bandwidth is lower than the threshold," the "isCdnInIssue" value is set to "r," which is true ("!1"). When buffer underflow occurs – i.e., "isCdnInIssue" is true – the "getPlaybackCdn" function returns "break," meaning that the function is switching to an alternative video stream from a different content distribution network (CDN) server. The "AdaptQuality" function then selects an alternative video stream based on the decision from the "getPlayBackCdn" function.

[125] ISO/IEC 23009-1:2014 at 18-19, 35, 108-109.

292. Hulu causes the processor to perform the step of, when a minimum buffer level criterion is satisfied based upon a playback duration of chunks of video content stored in the buffer of the playback device, "requesting at least one chunk of the selected stream from the plurality of alternative video streams." '987 patent, claim 1. For example, Hulu selects a video stream (*i.e.*, a Representation) from a set of alternative video streams (i.e., an Adaptation Set) provided by the Hulu server using a playback device.[126] For example, the set of plurality of video streams are identified by "Representation id" values. For the selected video stream, Hulu first requests the header information and then the actual video content, in byte ranges, for the same video stream.

293. Hulu causes the processor to perform the step of, when a minimum buffer level criterion is satisfied based upon a playback duration of chunks of video content stored in the buffer of the playback device, "storing the at least one chunk of the selected stream from the plurality of alternative video streams in the buffer of the playback device." '987 patent, claim 1. For example, Hulu stores the requested chunks (e.g., entire Segments or byte ranges of Segments) in a buffer of the playback device.[127] Further, Hulu uses the Javascript Player, which tracks video segments saved in the buffer through the "videoBufferLength" variable, indicating that Hulu downloads the video segment (i.e., chunk of the video) and stores it in a buffer.

294. Hulu causes the processor to perform the step of, when a minimum buffer level criterion is satisfied based upon a playback duration of chunks of video content stored in the buffer of the playback device, "playing back the at least one chunk of the selected stream from the plurality of alternative video streams stored

[126] ISO/IEC 23009-1:2014 at 108.

[127] ISO/IEC 23009-1:2014 at 10, 108.

in the buffer." '987 patent, claim 1. Hulu allows the playback device to play back the buffered chunks (e.g., entire Segment or byte ranges of Segments).[128]

295. At least as of the date of this Complaint, Hulu knows that it provides and specifically intends to provide an application and service for CE playback devices that, when used as intended, meets the limitations of claim 1, as described in ¶¶ 276-294. Hulu therefore has induced, and continues to induce, infringement of at least claim 1 of the '987 patent in violation of 35 U.S.C. § 271(b) in the exemplary manner described herein.

296. Hulu's infringement has caused and continues to cause damage to DivX, and DivX is entitled to recover damages sustained as a result of Hulu's wrongful acts in an amount subject to proof at trial.

JURY TRIAL DEMANDED

DivX hereby demands a trial by jury on all claims and issues so triable.

PRAYER FOR RELIEF

WHEREFORE, DivX respectfully requests that the Court:

A. Enter judgment that Hulu has directly infringed one or more claims of one or more of the DivX Patents, either literally or under the doctrine of equivalents, in violation of 35 U.S.C. § 271(a);

B. Enter judgment that Hulu has induced infringement of one or more claims of the DivX Patents in violation of 35 U.S.C. § 271(b);

C. Enter an order, pursuant to 35 U.S.C. § 284, awarding to DivX damages adequate to compensate for Hulu's infringement of the DivX Patents (and, if necessary, related accountings), in an amount to be determined at trial, but not less than a reasonable royalty;

[128] ISO/IEC 23009-1:2014 at 10, 109.

D. Enter an order, pursuant to 35 U.S.C. § 285, deeming this to be an "exceptional case" and thereby awarding to DivX its reasonable attorneys' fees, costs, and expenses;

E. Enter an order that Hulu account for and pay to DivX the damages to which DivX is entitled as a consequence of the infringement;

F. Enter an order for a post-judgment equitable accounting of damages for the period of infringement of the DivX Patents following the period of damages established at trial;

G. Enter an order awarding to DivX pre- and post-judgment interest at the maximum rates allowable under the law; and

H. Enter an order awarding to DivX such other and further relief, whether at law or in equity, that this Court deems just and proper.

Dated: February 22, 2021

Respectfully submitted,

ROBINS KAPLAN LLP

By: */s/ Roman M. Silberfeld*

Roman M. Silberfeld, SBN 62783
RSilberfeld@RobinsKaplan.com
Daniel L. Allender, SBN 264651
DAllender@RobinsKaplan.com
ROBINS KAPLAN LLP
2049 Century Park East, Suite 3400
Los Angeles, CA 90067
Telephone: (310) 552-0130
Facsimile: (310) 229-5800

David M. Stein, SBN 198256
DStein@BrownRudnick.com
BROWN RUDNICK LLP
2211 Michelson Drive, 7th Floor
Irvine, CA 92612
Telephone: (949) 440-0231
Facsimile: (949) 486-3686

ATTORNEYS FOR PLAINTIFF DIVX, LLC

Pro hac vice motions to be filed:

Christopher A. Seidl
CSeidl@RobinsKaplan.com
Aaron R. Fahrenkrog
AFahrenkrog@RobinsKaplan.com
Bryan J. Mechell
BMechell@RobinsKaplan.com
Christine Yun Sauer
CYunSauer@RobinsKaplan.com
Emily J. Tremblay
ETremblay@RobinsKaplan.com
Shui Li
SLi@RobinsKaplan.com
Mary Pheng
MPheng@RobinsKaplan.com
Rajin Singh Olson
ROlson@RobinsKaplan.com
Deanna Thompson
DThompson@RobinsKaplan.com
ROBINS KAPLAN LLP
800 LaSalle Avenue, Suite 2800
Minneapolis, MN 55402
Telephone: (612) 349-8500
Facsimile: (612) 339-4181