# PETITIONERS' PRESENTATION

## Oral Argument: March 12, 2026

Stingray Group, Inc. and Stingray Music USA, Inc.

v.

Edwin A. Hernandez-Mondragon

IPR2025-00349: U.S. Patent No. 10,123,074

IPR2025-00350: U.S. Patent No. 10,524,002

IPR2025-00351: U.S. Patent No. 11,140,441

Presented by Steven Jugle

Exhibit

1037

exhibitsticker.com

# OVERVIEW OF THE CHALLENGED PATENTS

# OVERVIEW OF THE CHALLENGED PATENTS

- The three Challenged Patents all share the same specification.
- Directed to obtaining content from a webpage and assembling the content into a media stream for delivery to a content provider.

An example computer-implemented method can receive, from a content provider such as a cable or satellite operator, a request for at least one media stream for playback on a broadcast media channel. The request for the at least one media stream can include a request for a plurality of multimedia items. Content corresponding to the plurality of multimedia items can be obtained from at least one website or cloud service that offers the content in at least one first format. Based on the obtained content, the plurality of multimedia files can be generated in a second format that is compatible with the content provider. The media stream can assembled using each of the multimedia items generated in the second format. The media stream can be provided to the content provider for broadcast on the broadcast media channel.

'074 Patent at 2:50-64.

IPR349: Pet. at 9.

# OVERVIEW OF THE CHALLENGED PATENTS

- Patent Owner amended the claims during prosecution to describe this process:

1. (Currently Amended) A computer-implemented method, comprising:

receiving, from a content provider, a request for at least one media stream for playback on a broadcast media channel, wherein the at least one media stream includes a plurality of multimedia items of different types;

obtaining content corresponding to the plurality of multimedia items from at least one source~~website~~ offering the content in at least one first format;

rendering a web page by a browser using the content;

generating a temporal sequence of~~each of the plurality of multimedia items in a second format based on the content obtained from the at least one website, wherein the second format~~screen captures of the rendered web page, where each screen capture defines all the content of the web page at a given time, and at least two adjacent screen captures illustrate a dynamic change of at least a portion of the content over time~~is compatible with the at least one content provider~~;

assembling the at least one media stream using the temporal sequence of screen captures~~each of the plurality of multimedia items generated~~; and

providing the at least one media stream to the content provider for broadcast on the broadcast media channel.

Prosecution History, EX1011 at 62.

# OVERVIEW OF THE CHALLENGED PATENTS

- According to the Challenged Patents, this "rendering" and "screen capture" process can be implemented using commonly-known software tools:

UI request can be used. Moreover, this custom UI can include JavaScript, CSS, and HTML content files that can be rendered using a browser using WebKit (e.g. PhantomJS or Safari) or any other browser-rendering engine compatible with HTML4/5 or any future HTML version. If a customi-

'074 Patent at 6:67-7:4.

Capturing screens **526** can be done using a tool similar to PhantomJS (Webkit-based headless engine for rendering images) or QtCapture, or simply capturing screen frame buffers from a machine running any software tool. Alternatively, the method can capture a video file from another rendering function, which can create a sequence of screens that can be used to generate a video file. Once all screens and

'074 Patent at 7:18-24.

IPR349: Pet. at 9-11.

# OVERVIEW OF THE CHALLENGED PATENTS

- Similarly, the "assembly" of the media stream can also be performed using commonly-known software tools and formats:

> Once the output file containing the screen shoot sequence of videos encoded in H.264 (for example, or MPEG2Video) is created, the audio can be multiplexed and added to one of the encoding channels of the resulting video output and saved into the caching unit storage disk **550**. The multiplexing can be done using tools such as FFMPEG or VLC or any proprietary tool that can multiplex audio and video files into a resulting MPEG Transport Stream (or MPEG-4 output, or any similar output).

'074 Patent at 7:35-43.

IPR349: Pet. at 9-11.

# OVERVIEW OF THE PRIMARY PRIOR ART

# OVERVIEW OF AVELLAN

- Avellan discloses the same process claimed in the Challenged Patents.
- Directed to a system that can "broadcast information from a web server" and can deliver "live streaming television feeds, movies, music downloads, ESPN" and other multimedia streams.

> less specifically-customized information. Accordingly, suitable sites for the broadcast mode are news sites like CNN, BBC, newspaper, live streaming television feeds, movies, music downloads, or ESPN. Those sites provide the same

Avellan, EX1003 at 8:18-21.

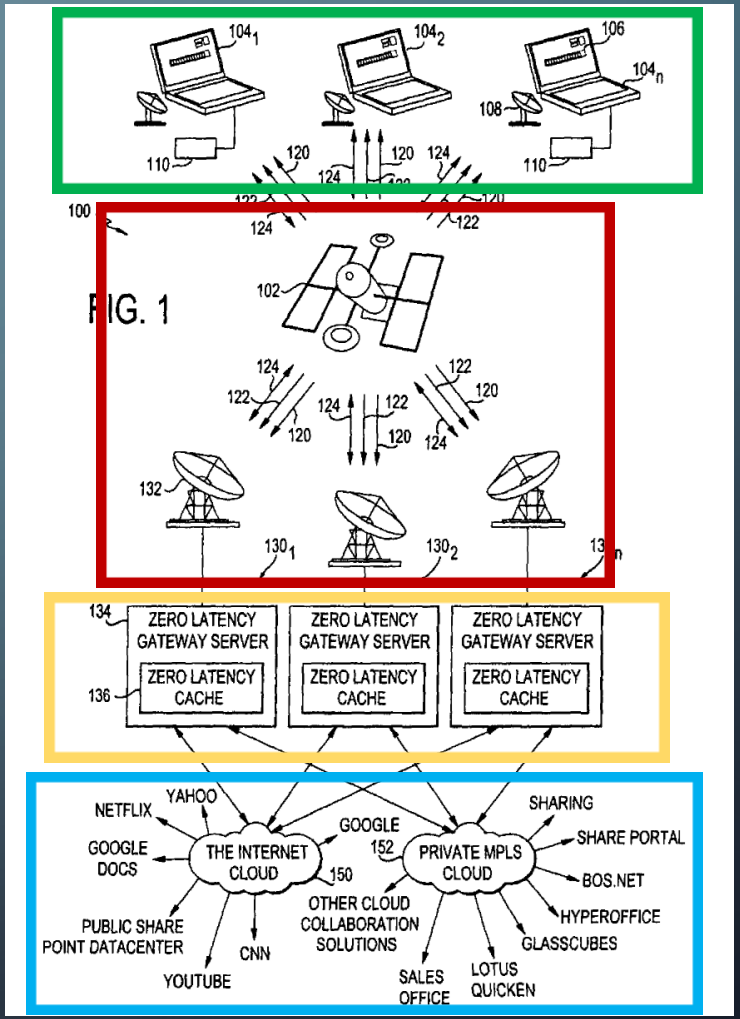
> information is not specific to the user (unlike a search site). In addition, there can be a number of broadcast channels and each channel can have a specific theme, such as country music, military television, etc. The user can then tune the browser **106** to one of the broadcast channels to receive that content. Thus, the broadcast channel can be configured here

Avellan, EX1003 at 8:24-29.

IPR349: Pet. at 11-12.

# OVERVIEW OF AVELLAN

- Avellan can deliver requested media streams to user computers 104.
- The request is transmitted via broadcast device 102 and broadcast receiver 132.
- The gateway server 134 receives request and obtains content from the Internet 150 or Private Cloud 152.

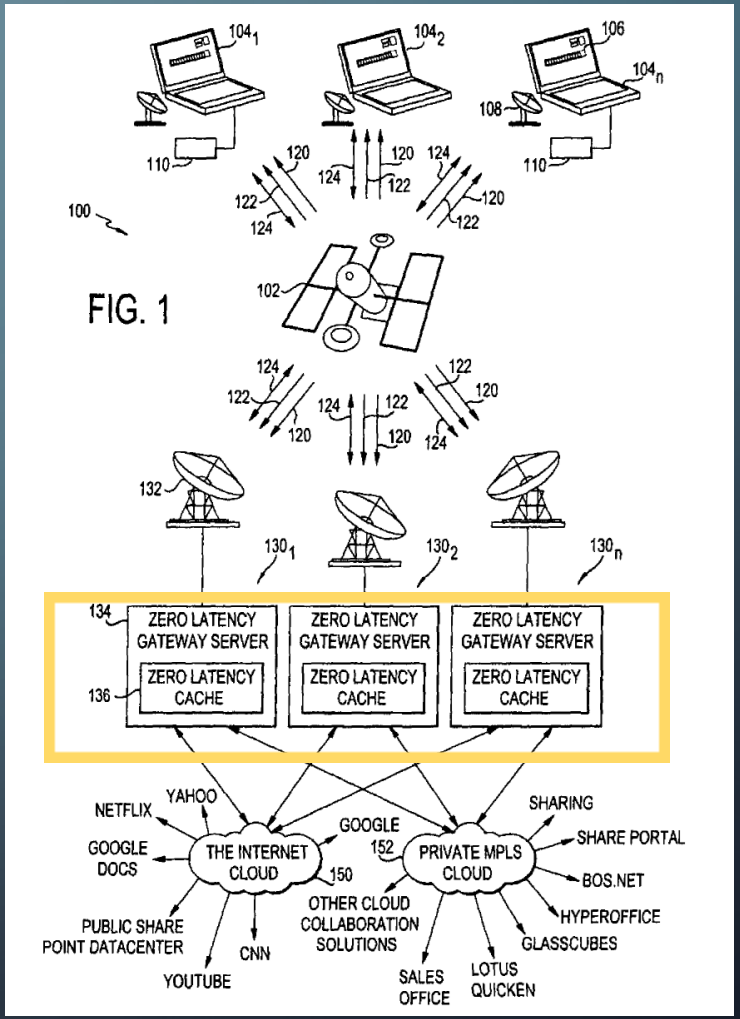


# OVERVIEW OF AVELLAN

- Gateway server 134 includes browser software 106 that renders the content.
- Gateway server 134 then generates screen captures of the rendered content.

As mentioned above, the browser software **106** resides at the gateway **134**, and is configured to run as a virtual browser as a remote client of the gateway server **134**. Because the browser **106** functions as a virtual browser, viruses that attack operating systems and threaten users' security are unable to infest the user's individual computing devices. In addition, the remote browser **106** may rely upon the computing power and very high-speed connection of the gateway server **134** to resolve and render pages, images, and documents from the networks **150**, **152**. The gateway server **134** then images those web pages (which are typically in an HTTP format) to a video or image frame format. This is preferably done by taking an image of the entire web page from top to bottom, including any portion that would normally be accessed by scrolling up/down or left/right but not immediately displayed on the screen because it is too long for a single screen.

Avellan, EX1003 at 4:56-5:4.

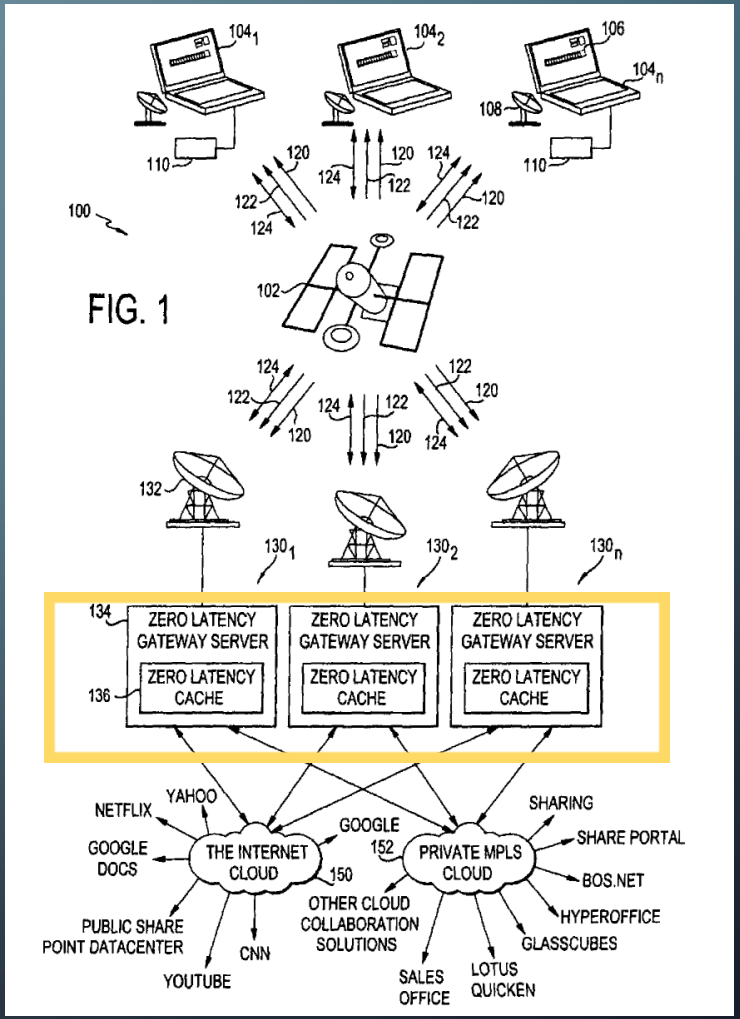


# OVERVIEW OF AVELLAN

- Avellan can "continuously capture" video/streaming content.
- Captures can then be assembled into a video frame format for delivery.

> In addition, web pages often have information which is rendered over a period of time, such as a video, streaming information, or an animated display. Accordingly, the gateway server **134** continuously captures images of the web page until it detects a repeating pattern of information in the web page. Thus, the gateway server **134** will be certain to capture all the information on the web page even if that information continues over a period of time. Once the page is imaged (including any audio on the web page), the gateway server **134** compresses the image or video frame format and sends it to the browser **106** via the satellite **102**.

Avellan, EX1003 at 5:5-15.



# OVERVIEW OF WANNAMAKER

- Wannamaker also discloses a process for delivering web content to users.
- The disclosed platform "converts Internet content and protocols into forms suitable for TV viewing" using the same rendering process.

**[0097]** Web Content Collection Process

**[0098]** The information platform converts Internet content and protocols into forms suitable for TV viewing. The content is fetched from Web sites, converted or rendered by the system, and then cached on application server The information platform supports content from standard Web sites, e.g. HTML source, GIF, animated GIF, JPEG, and PNG files. The majority of HTML 3.2 tags are supported, e.g. text, tables, forms, images, image maps, background images, and sounds.

Wannamaker, EX1006 at [0098].

IPR350: Pet. at 15-18.

# OVERVIEW OF WANNAMAKER

- Cache managers 37 accept requests for web-based content.

> **[0236]** Cache managers **37** (see **FIG. 3**) are responsible for accepting requests for content and handling the steps required to deliver the content to the necessary location. Once content is pulled from the Internet, the crawler **35** deposits the unrendered content (HTML, JPEG) into source cache. The rendered **36** cache is a storage area for content

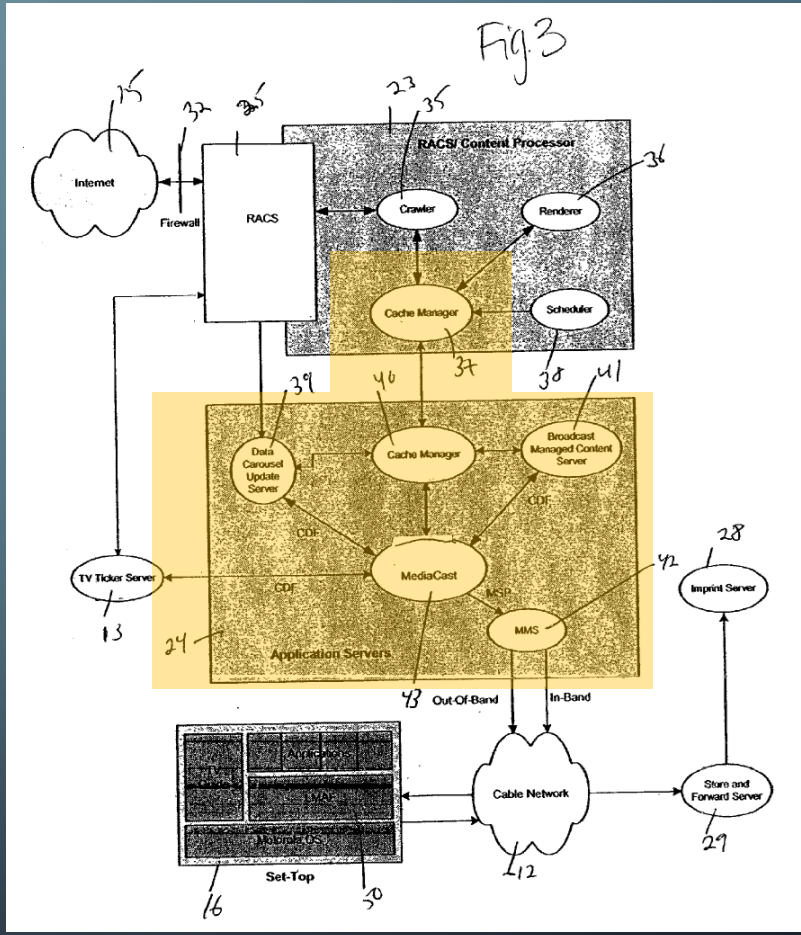
Wannamaker, EX1006 at [0236].

- Content processor 23 renders the web content for insertion in an MPEG stream.

> **[0244]** A content processor (CP) cluster produces digital cable system compatible (rendered) content from standard Web content retrieved from the Internet. Multiple content

> **[0271]** 2. The server continuously transmits the Web data to the client in an MPEG video stream. The server puts images of the rendered Web page in MPEG I-frames and MPEG P-frames. In addition, the server transmits some

Wannamaker, EX1006 at [0244] and [0271].



# OVERVIEW OF FARBER

- The ’441 Patent also includes certain claims that require creating custom HTML user interfaces (see Claims 1-9).
- Farber discloses a music broadcast system “designed for transmitting various music channels over . . . a cable television network.”

> An audio broadcast system **40** is designed for transmitting various music channel broadcasts over a network such as a cable television network or a satellite network for distribution to a plurality of subscribers. It should be understood, how-

Farber, EX1003 at 5:9-12.

IPR351: Pet. at 11-12.

# OVERVIEW OF FARBER

- Each music channel includes audio data and an accompanying user interface.

FIG. **1** shows a screen **1** which is displayed coincident with the transmission of a broadcast music channel. The screen **1** contains various assets including a logo **10**, a song add button **12**, a download button **14**, a personalized channel choice button **16**, a channel name **18**, album art **20**, title track label and artist information **22**, promotional/advertising panels **24**, **28** and a buy button **26**.

Farber, EX1003 at 3:38-44.

- Farber's user interfaces can be encoded using HTML.

sources **42**, **44**. The data carousel subsystem **46** is capable of managing/creating the templates, which may be coded utilizing HTML, XML or other suitable protocols for creating templates/screens having text and other assets such as those described above.

Farber, EX1003 at 5:50-54.

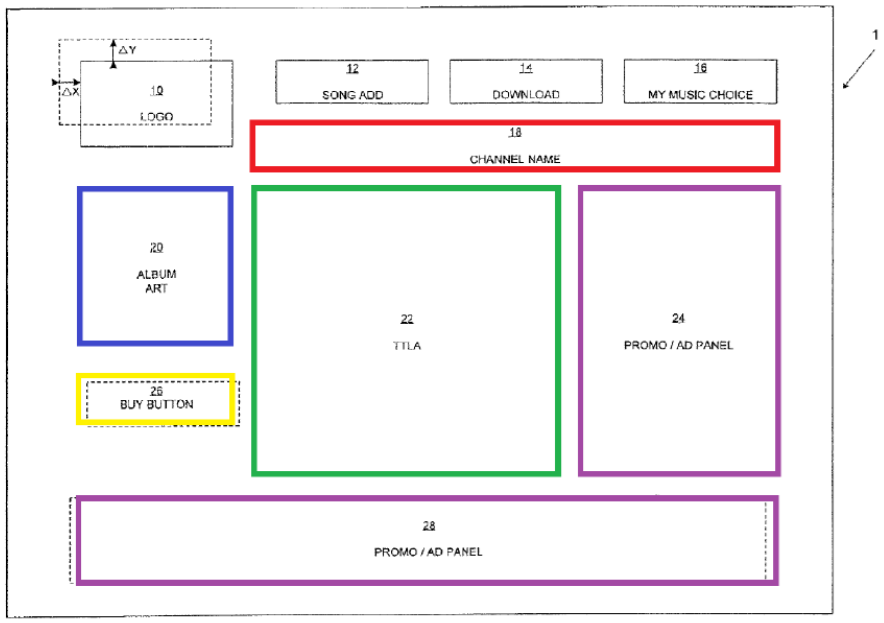



Fig. 1

# OVERVIEW OF FARBER

- Farber generates an image of the UI and encodes it into an MPEG frame.

which may be an MPEG encoder **47**. In response to the trigger, the encoder **47** pulls template information, images, and text from the data carousel subsystem **46** and creates, based on the template information, a screen having embedded assets such as those shown in screens **1**, **2** or **3** which are associated with the currently queued song. The screen may be a video frame, such as an MPEG video frame.

Farber, EX1003 at 6:2-8.

- Each channel's song and video data is then combined into a transport stream.

It should be understood that this process may be executed for a plurality of channels simultaneously. The MPEG encoder **47** then creates an MPEG transport stream for all channels with the embedded MPEG video frames. The MPEG encoder **47** may be implemented utilizing a commercially available encoder or a general-purpose computer.

Farber, EX1003 at 6:9-14.

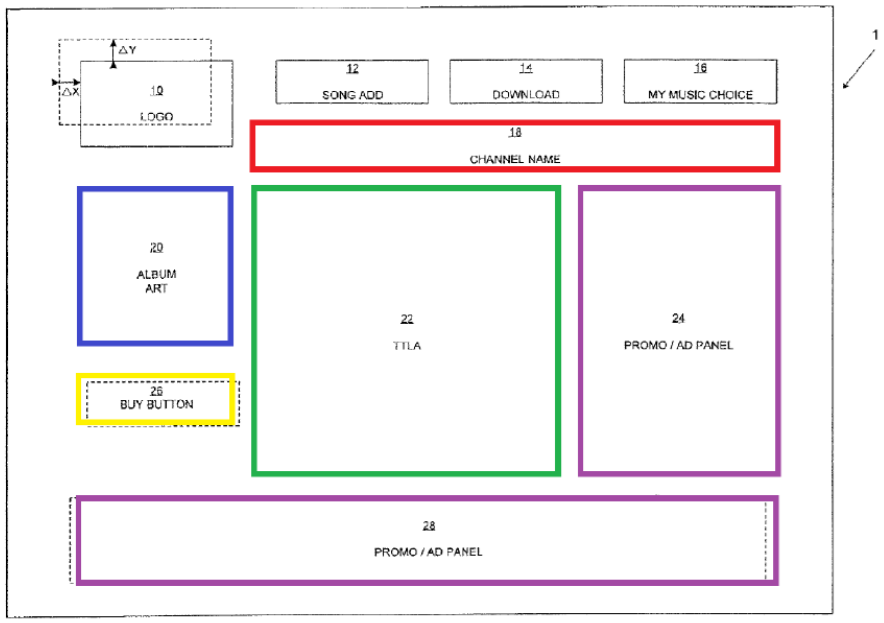



Fig. 1

Source: '351 Petition at 22-24; EX1003 at 3:38-44 and 5:50-54.

# INVALIDITY ANALYSIS

# AVELLAN

**U.S. Patent No. 8,954,600**

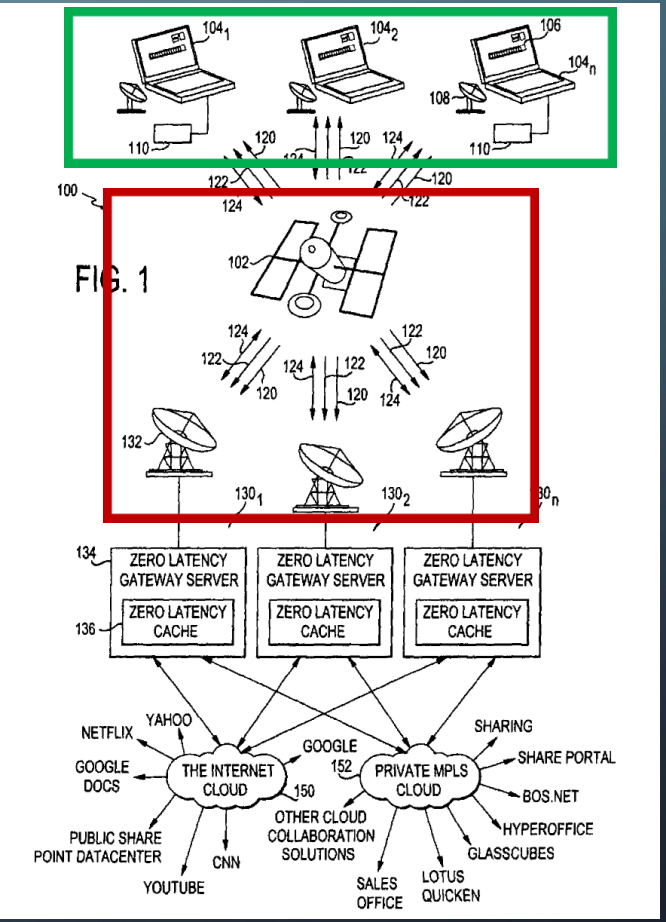
## THE ’074 PATENT: CLAIM 1

| | |
|---|---|
| **Preamble 1** | A computer-implemented method comprising: |
| **Element 1a** | receiving, from a content provider, a request for at least one media stream for playback on a broadcast media channel, wherein the at least one media stream includes a plurality of multimedia items of different types; |

- Avellan's user computers 104 can request media streams for delivery via broadcast device 102.

MPLS (MultiProtocol Label Switching) cloud. Generally, the system **100** provides for the user computers **104**$_1$-**104**$_n$ to access the Internet **150** and/or private network cloud **152** through a virtual browser **106** of the gateway **130**. The user operates the browser **106** which sends a signal over the satellite **102** to the gateway **130**. The gateway **130** accesses the networks **150**, **152** and returns the results to the virtual browser **106** via the satellite **102**.

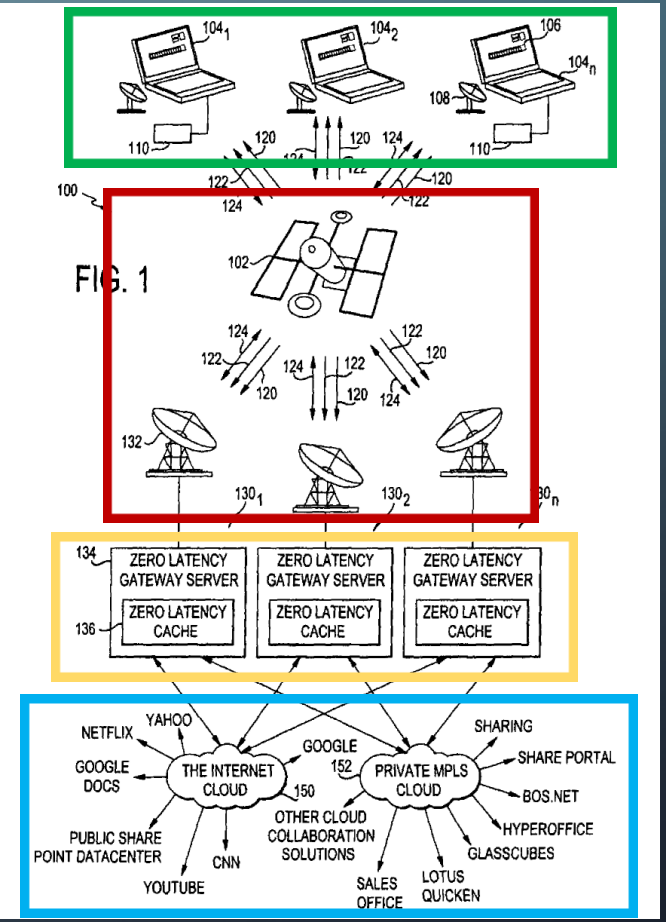
Avellan at 4:18-25.




Avellan at Fig. 1.

| | |
|---|---|
| **Preamble 1** | A computer-implemented method comprising: |
| **Element 1a** | receiving, from a content provider, a request for at least one media stream for playback on a broadcast media channel, wherein the at least one media stream includes a plurality of multimedia items of different types; |

- The gateway receives the request from broadcast device 102 and broadcast receiver 132.
- Request is processed by gateway server 134, which obtains content for the stream from the Internet 150 or Private Cloud 152.




Avellan at Fig. 1.

| | |
|---|---|
| **Preamble 1** | A computer-implemented method comprising: |
| **Element 1a** | receiving, from a content provider, a request for at least one media stream for playback on a broadcast media channel, wherein the at least one media stream includes a plurality of multimedia items of different types; |

- Avellan can broadcast "live streaming television feeds" and other live content.

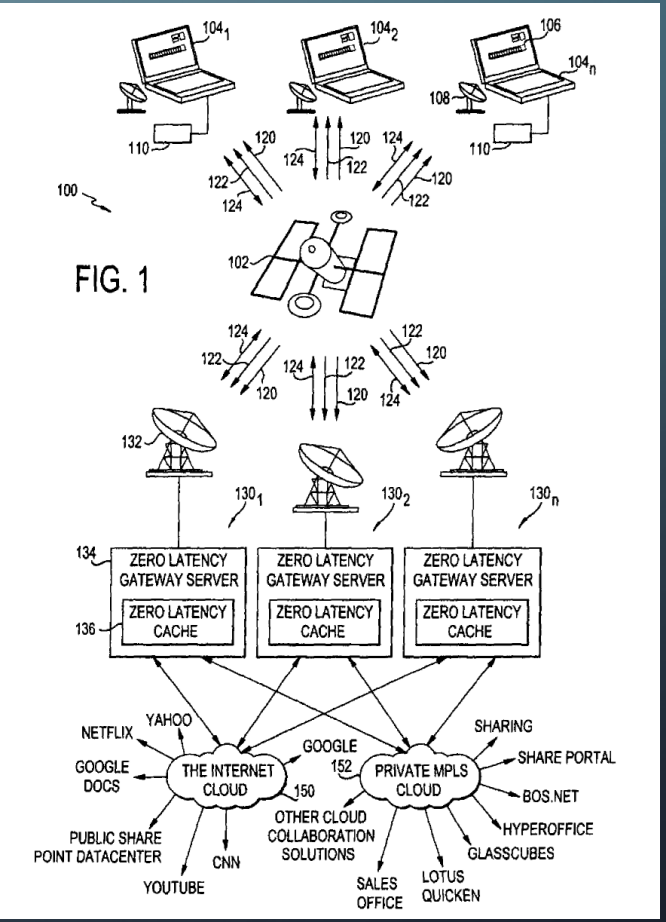
> less specifically-customized information. Accordingly, suitable sites for the broadcast mode are news sites like CNN, BBC, newspaper, live streaming television feeds, movies, music downloads, or ESPN. Those sites provide the same

Avellan at 8:18-21.

- Streams include a plurality of multimedia items of different types.

> information is not specific to the user (unlike a search site). In addition, there can be a number of broadcast channels and each channel can have a specific theme, such as country music, military television, etc. The user can then tune the browser **106** to one of the broadcast channels to receive that content. Thus, the broadcast channel can be configured here

Avellan at 8:24-29.




Avellan at Fig. 1.

| | |
|---|---|
| **Element 1b** | obtaining content corresponding to the plurality of multimedia items from at least one source offering the content in at least one first format; |

- Avellan obtains content from the Internet (e.g., Netflix, CNN, YouTube) or a Private Cloud.

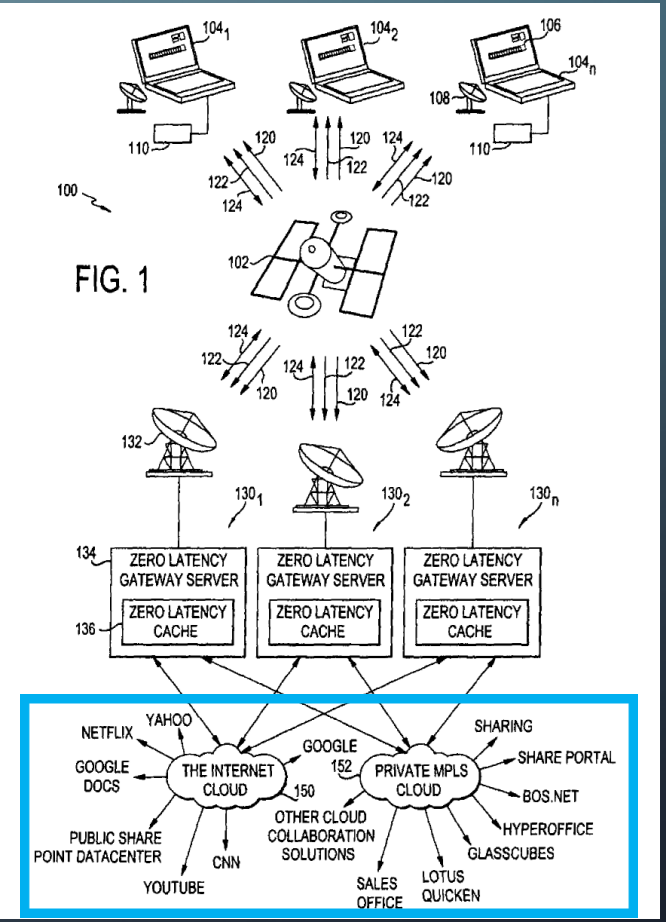
> The gateway server **134** may be configured to communicate with the public computers, servers, networks or clouds such as the Internet cloud **150**, and/or one or more private computers, servers, networks or clouds **152** such as a private MPLS (MultiProtocol Label Switching) cloud. Generally, the

Avellan at 4:14-18.

- Challenged Patents similarly obtain content from "websites or cloud based servers."

> At **604**, contents corresponding to the multimedia items can be retrieved, downloaded, or otherwise obtained from one or more websites or cloud based servers. The contents can include data corresponding to either or both of the audio and video components of the multimedia items. For

'074 Patent at 7:55-57.




Avellan at Fig. 1.

| Element 1c | rendering a web page by a browser using the content; |
|---|---|

- Avellan discloses "browser software 106" residing on the gateway server.

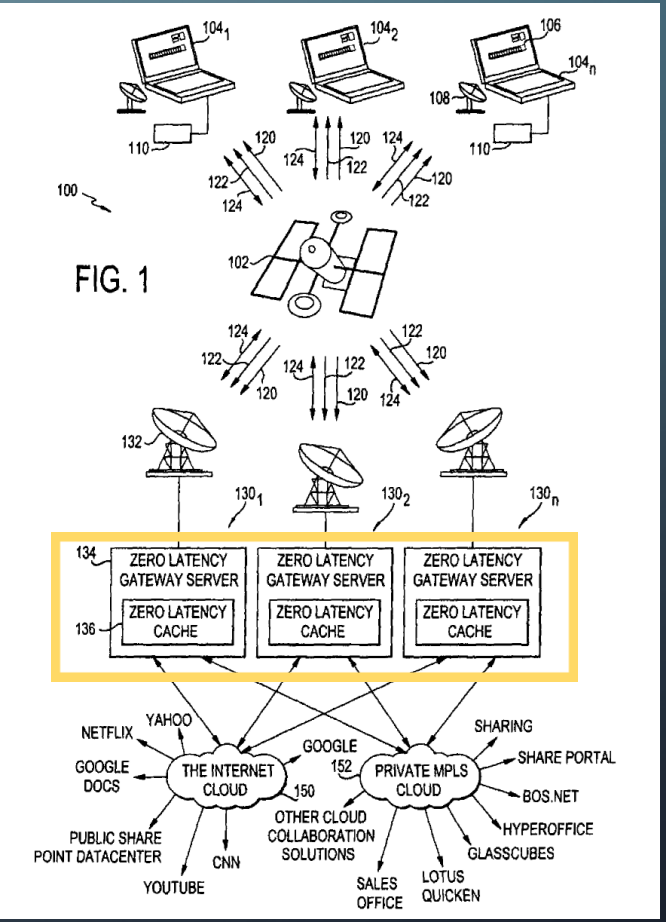
> As mentioned above, the browser software **106** resides at the gateway **134**, and is configured to run as a virtual browser as a remote client of the gateway server **134**. Because the

Avellan at 4:56-58.

- This browser can "render pages" from the Internet 150 and Private Cloud 152.

> infest the user's individual computing devices. In addition, the remote browser **106** may rely upon the computing power and very high-speed connection of the gateway server **134** to resolve and render pages, images, and documents from the networks **150**, **152**. The gateway server **134** then images those web pages (which are typically in an HTTP format) to a video or image frame format. This is preferably done by taking an

Avellan at 4:61-67.



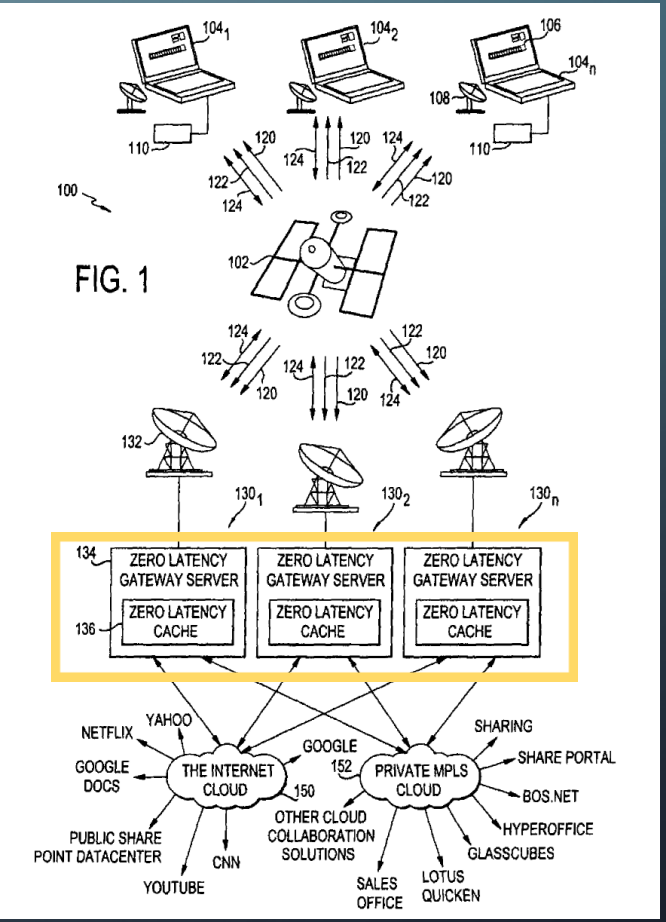

Avellan at Fig. 1.

| | |
|---|---|
| **Element 1d** | generating a temporal sequence of screen captures of the rendered web page, where each screen capture defines all the content of the web page at a given time, and at least two adjacent screen captures illustrate a dynamic change of at least a portion of the content over time; |

- Avellan discloses capturing all the content of the rendered web page at a given time.

networks **150**, **152**. The gateway server **134** then images those web pages (which are typically in an HTTP format) to a video or image frame format. This is preferably done by taking an image of the entire web page from top to bottom, including any portion that would normally be accessed by scrolling up/down or left/right but not immediately displayed on the screen because it is too long for a single screen.

Avellan at 4:65-5:4.



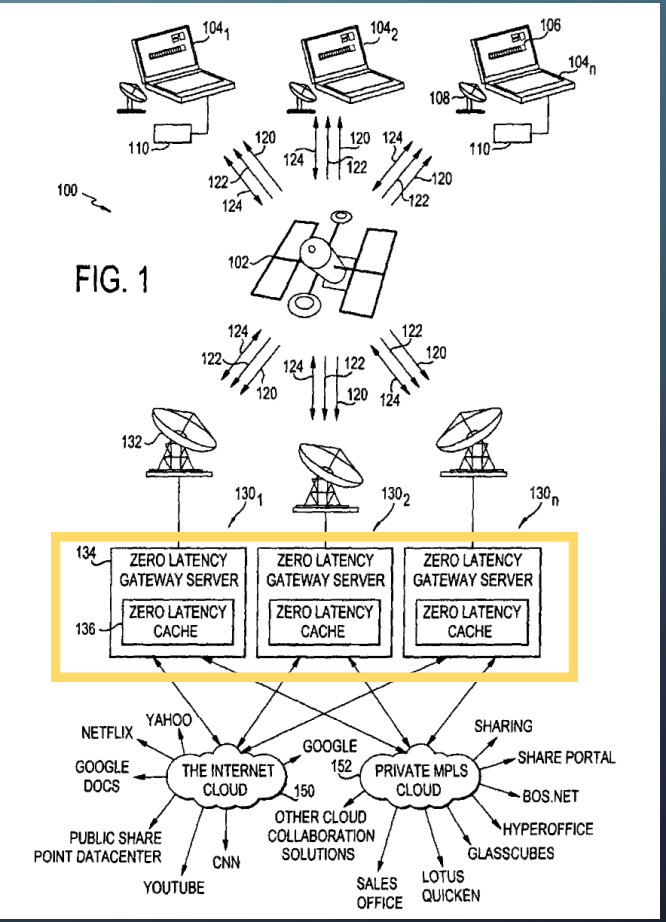

Avellan at Fig. 1.

| | |
|---|---|
| **Element 1d** | generating a temporal sequence of screen captures of the rendered web page, where each screen capture defines all the content of the web page at a given time, and at least two adjacent screen captures illustrate a dynamic change of at least a portion of the content over time; |

- Avellan generates a temporal sequence of screen captures.

In addition, web pages often have information which is rendered over a period of time, such as a video, streaming information, or an animated display. Accordingly, the gateway server **134** continuously captures images of the web page until it detects a repeating pattern of information in the web page. Thus, the gateway server **134** will be certain to capture all the information on the web page even if that information continues over a period of time. Once the page is imaged

Avellan at 5:5-12.



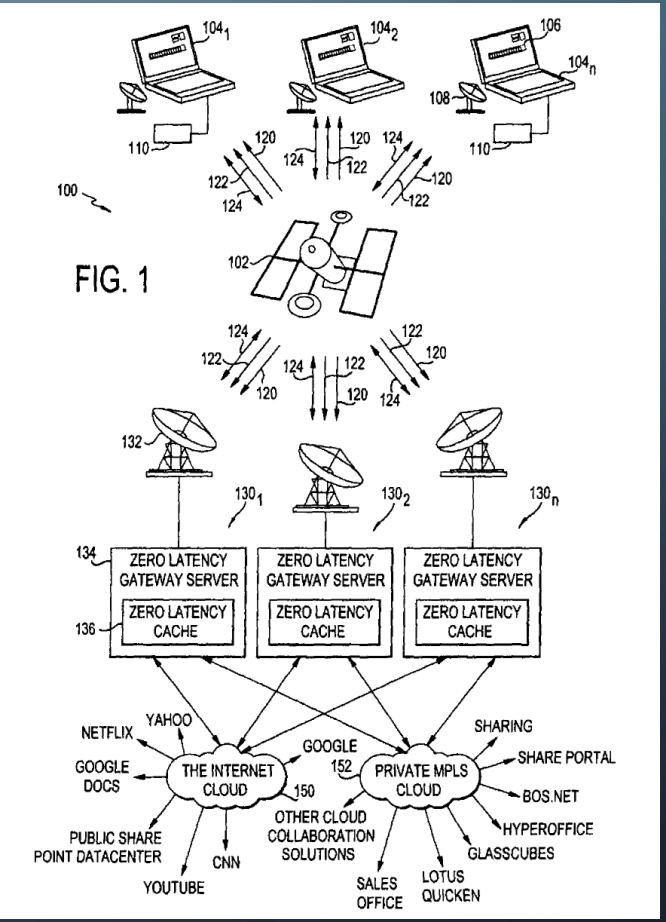

Avellan at Fig. 1.

| | |
|---|---|
| **Element 1d** | generating a temporal sequence of screen captures of the rendered web page, where each screen capture defines all the content of the web page at a given time, and at least two adjacent screen captures illustrate a dynamic change of at least a portion of the content over time; |

- Avellan's "continuous captures" illustrate a dynamic change.

In addition, web pages often have information which is rendered over a period of time, such as a video, streaming information, or an animated display. Accordingly, the gateway server **134** continuously captures images of the web page until it detects a repeating pattern of information in the web page. Thus, the gateway server **134** will be certain to capture all the information on the web page even if that information continues over a period of time. Once the page is imaged

Avellan at 5:5-12.




Avellan at Fig. 1.

| | |
|---|---|
| **Element 1d** | generating a temporal sequence of screen captures of the rendered web page, where each screen capture defines all the content of the web page at a given time, and at least two adjacent screen captures illustrate a dynamic change of at least a portion of the content over time; |

- Avellan's "continuous captures" illustrate a dynamic change.

hockey games and basketball games. If a large number of viewers choose to select a certain hockey game, that page of the website can also be sent in the broadcast mode. On the

Avellan at 8:34-36.

less specifically-customized information. Accordingly, suitable sites for the broadcast mode are news sites like CNN, BBC, newspaper, live streaming television feeds, movies, music downloads, or ESPN. Those sites provide the same

Avellan at 8:18-21.

of that content over time. For example, if a streaming hockey game is being rendered, a screen capture at Time 1 followed by a screen capture at Time 2 will illustrate a dynamic change in that game, such as the movement of players on the ice between Time 1 and Time 2—*i.e.*, a dynamic change of at least a portion of the content over time.

Lipoff Declaration, IPR349 EX1002 at Paragraph 111.

| | |
|---|---|
| **Element 1e** | assembling the at least one media stream using the temporal sequence of screen captures; and |
| **Element 1f** | providing the at least one media stream to the content provider for broadcast on the broadcast media channel. |

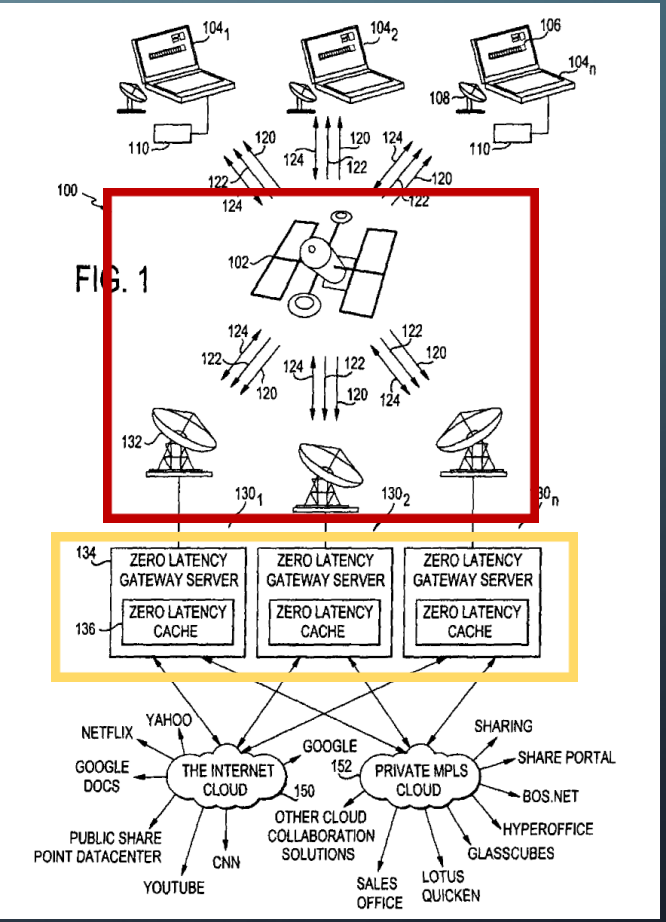
- Avellan's gateway can assemble the captured images into a digital video format.
- The stream is then broadcast via broadcast transmitter 132 and broadcast device 102.

continues over a period of time. Once the page is imaged (including any audio on the web page), the gateway server **134** compresses the image or video frame format and sends it to the browser **106** via the satellite **102**.

Avellan at 5:12-15.

In a non-limiting aspect of the invention, a web-to-HD video protocol may be used to capture and transmit images/ audio of a web page, including a full multimedia web page, regardless of its size or original format (e.g., HTTP). The protocol may be configured to compress the data into a compressed digital video format that can be sent to the user computer **104** for display in the browser **106**. Since HTTP is

Avellan at 5:16-22.




Avellan at Fig. 1.

# AVELLAN

**U.S. Patent No. 8,954,600**

THE ’074 PATENT: CLAIMS 2-21

THE ’002 PATENT: CLAIMS 1-6, 13

THE ’441 PATENT: CLAIMS 10-26

# AVELLAN AND THE ’074 PATENT

- Analysis of Claim 1 also applies to Independent Claims 12 and 17.
- Claim 8 requires detecting a change in a website and updating the screen captures (see also Claim 13 of the ’441 Patent).

| | |
|---|---|
| **Element 8a** | The method of claim 3, further comprising: detecting a change at the at least one webpage corresponding to at least one of the plurality of multimedia items; |
| **Element 8b** | in response to detecting the change, retrieving a new plurality of screen captures from the at least one webpage for the at least one of the plurality of multimedia items; and |
| **Element 8c** | reassembling the at least one of the plurality of multimedia items using the new plurality of screen captures to yield an updated multimedia item. |

As the broadcast sites are updated, the updated sites are cached **136** and sent to the user computers **104**. The gateway server **134** can track the updates in several ways. For instance, the server **134** can continuously capture the broadcast data at the networks **150**, **152**. Or, the gateway server **134** can continuously or periodically compare the broadcast data stored in the cache **136** with the broadcast data at the networks **150**, **152** and re-capture any broadcast data which is updated. As the information is updated, the updated broadcast data is compressed, saved to cache **136**, broadcast to the user computers **104**, and stored in the storage device **110**.

Avellan at 8:63-9:6.

IPR349: Pet. at 28-30; Reply at 17-18. IPR351: Pet. at 53; Reply at 24.

# AVELLAN AND THE ’074 PATENT

- In general, the Challenged Claims add well-known and common features.
  - The use of audio and video multimedia items (claims 2, 3, 4, 13, 18)
  - The use of metadata (claims 5, 21)
  - The use of channel identifiers, music genres, or multiple streams (claims 6, 14, 15)
  - The use of common standards, such as MPEG, HLS, or RTSP (claims 7, 10, 19)
  - The content provider is a satellite television operator (claim 9).
  - The use of virtual machines (claim 11)
  - The use of broadcast IP or unicast addresses (claim 16)
  - The use of recorded streams (claim 20)

# AVELLAN AND THE ’002 PATENT

- In general, the Challenged Claims add well-known and common features.
  - The use of common standards, such as MPEG, HLS, or RTSP (claim 1)
  - The use of metadata or MPEG two-way control messages (claims 2, 4, 5)
  - The use of unicast addresses (claims 3, 6)
  - The use of rendering in parallel (claims 4, 13)
  - The use of virtual machines (claim 13)

# AVELLAN AND THE '002 PATENT

- For example, Claim 1 adds the separate elements of the use of the HTTP Live Streaming (HLS) standard and rendering in parallel using multiple threads.

| | |
|---|---|
| [1g] | wherein the at least one media stream corresponds to at least one of an HTTP live stream (HLS), an HTTP playlist, and a Real-time Streaming Transport (RTSP) stream; and |
| [1h] | wherein the web page is rendered in parallel in multiple threads. |

with the gateway server **134**. The cache **136** is shown in FIG. **1** as part of the server **134**, but can be separate from the server **134**, and can be memory or other storage device. The gateway cache **136** preferably stores the web pages in the compressed video format, but can store the data in any suitable format(s) including HTTP format.

Avellan at 18:48-53.

- Examiner concluded that the use of HLS was not new (see EX1002 at 39).
- Petition cites the Ma prior art reference:

3GP DASH segments, etc.). In one embodiment, the output files, both segmented and un-segmented, are encrypted using standard encryption protocols (e.g., AES-128, HC-128, RC4, etc.). In one embodiment, the live stream processor **102** generates manifest files indicating segment retrieval locations for each encoding (e.g., HLS m3u8, DASH MPD, etc.).

Ma at [0033].

IPR349: Pet. at 30-31 and 40-44.

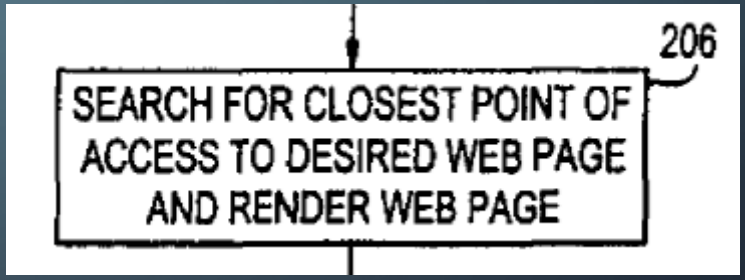
# AVELLAN AND THE ’002 PATENT

- For example, Claim 1 adds the separate elements of the use of the HTTP Live Streaming (HLS) standard and rendering in parallel using multiple threads.

| | |
|---|---|
| **[1g]** | wherein the at least one media stream corresponds to at least one of an HTTP live stream (HLS), an HTTP playlist, and a Real-time Streaming Transport (RTSP) stream; and |
| **[1h]** | wherein the web page is rendered in parallel in multiple threads. |

Although the steps in FIG. **2** have been described in sequential order for ease of understanding, it is important to note that many, if not all, of these steps may be performed in varying orders and/or in parallel. For instance, the server **134**

Avellan at 14:4-7.




Avellan at Fig. 2.

IPR349: Pet. at 30-31 and 40-44.

# AVELLAN AND THE '002 PATENT

- For example, Claim 1 adds the separate elements of the use of the HTTP Live Streaming (HLS) standard and rendering in parallel using multiple threads.

| | |
|---|---|
| **[1g]** | wherein the at least one media stream corresponds to at least one of an HTTP live stream (HLS), an HTTP playlist, and a Real-time Streaming Transport (RTSP) stream; and |
| **[1h]** | wherein the web page is rendered in parallel in multiple threads. |

Although the steps in FIG. **2** have been described in sequential order for ease of understanding, it is important to note that many, if not all, of these steps may be performed in varying orders and/or in parallel. For instance, the server **134**

Avellan at 14:4-7.

- Fain discloses that web-browsers support multiple threads.

A Java program can start multiple threads of execution that will run in parallel. Even if you have only one processor in your computer, it still could be a good idea to run some tasks in parallel. Think of a web browser that allows you to download a file and perform page brows-ing at the same time. Web browsers maintain two or more connections to any website, and can download multiple resources (text, images, videos, music) at the same time. Any web browser works in a multi-threaded mode.

Fain at 209 (PDF page 245 of 506).

IPR349: Pet. at 30-31 and 40-44.

# AVELLAN AND THE ’441 PATENT

- In general, the Challenged Claims add well-known and common features.
  - The use of virtualization or virtual machines (claims 10, 20, 21, 22, 23)
  - The use of audio and video multimedia items (claims 11, 12, 14)
  - The use of metadata (claim 15)
  - The use of channel identifiers, music genres, or multiple streams (claims 16)
  - The use of common standards, such as MPEG, HLS, or RTSP (claims 17, 19)
  - The content provider is a satellite television operator (claim 18).
  - The use of JSON API (claim 24)
  - The use of recorded streams and failure monitoring (claims 25, 26)

# AVELLAN AND THE CHALLENGED CLAIMS

- In general, the Challenged Claims add well-known and common features.
- Patent Owner does not contend he invented any of these features, which were all routinely implemented by POSITAs.
- Those features are either directly disclosed in Avellan and/or plainly disclosed in the cited secondary references.

# PATENT OWNER'S ATTEMPTS TO DISTINGUISH AVELLAN ARE UNAVAILING

# ARGUMENT: AVELLAN'S CONTENT PROVIDER DOES NOT "INITIATE" THE REQUEST?

- Patent Owner argues that the request for a media stream in Avellan is "initiated" at the user computers, not at the content provider.

"request" capability. The broadcast device described in Avellan is merely a passive intermediary in the communication path. It does not itself "request" a media stream, but only transmits a computer user-generated request. As such, it cannot satisfy the active method step of "receiving, from a content provider, a request for at least one media stream . . ." required by the challenged claims. Accordingly, the cited

IPR349: PO's Response at 50.

LEXIS 1033. A POSITA would understand that the use of term "content provider" as used intrinsically in the Specification refers to entities such as cable or satellite operators, and that the inventor describes his invention in the specification as requiring that a content provider initiate a request for at least one media stream.

IPR349: PO's Sur-Reply at 10-11.

# ARGUMENT: AVELLAN'S CONTENT PROVIDER DOES NOT "INITIATE" THE REQUEST?

- The Challenged Claims merely recite receiving a request from a content provider – the claims do not require that the request originates at or is initiated by the content provider.
- The Challenged Patents never describe the request as being "initiated" by or "originating" at the content provider – Patent Owner is adding a limitation unsupported by the specification.
- Patent Owner fails to identify any basis for rewriting the plain claim language.
  - Did not pursue claim construction.
  - Did not seek to amend the claim language.

IPR349: Reply at 7-8. IPR350: Reply at 6-7. IPR351: Reply at 17-18.

# ARGUMENT: AVELLAN TEACHES AWAY FROM USING A BROWSER?

- Patent Owner argues that Avellan does not disclose a browser that renders a web page.

> Avellan teaches away from *rendering a web page by a browser using the content.* Avellan is specifically structured to **NOT** RENDER A WEB PAGE BY A BROWSER. Avellan teaches, "the browser software 106 resides at the gateway 134,

IPR349: PO's Response at 57.

# ARGUMENT: AVELLAN TEACHES AWAY FROM USING A BROWSER?

- Avellan explicitly discloses "browser software 106" for rendering web pages.

As mentioned above, the browser software **106** resides at the gateway **134**, and is configured to run as a virtual browser as a remote client of the gateway server **134**. Because the

Avellan at 4:56-58.

infest the user's individual computing devices. In addition, the remote browser **106** may rely upon the computing power and very high-speed connection of the gateway server **134** to resolve and render pages, images, and documents from the networks **150**, **152**. The gateway server **134** then images those web pages (which are typically in an HTTP format) to a video or image frame format. This is preferably done by taking an

Avellan at 4:61-67.

IPR349: Pet. at 21; Reply at 9-10. IPR350: Pet. at 26; Reply at 8-9. IPR351: Pet. at 48; Reply at 19.

# ARGUMENT: PETITIONERS TREAT WEB PAGES AS MEDIA STREAMS?

- Patent Owner argues that a web page "cannot be a media stream for playback."

a recipient of a "media stream." EX2039¶¶98, 115. Petitioners treat web pages as media streams, which is clearly incorrect, as this method is constrained to show only to media visually appearing on the webpage.

IPR349: PO's Response at 25.

(*See* EX1003). A POSITA would understand that a webpage is not and cannot be a "media stream" for playback.

IPR349: PO's Sur-Reply at 12.

# ARGUMENT: PETITIONERS TREAT WEB PAGES AS MEDIA STREAMS?

- Patent Owner misunderstands the plain and ordinary claim language – requires assembling a media stream using screen captures of a rendered web page.

rendering a web page by a browser using the content;
generating a temporal sequence of screen captures of the rendered web page, where each screen capture defines all the content of the web page at a given time, and at least two adjacent screen captures illustrate a dynamic change of at least a portion of the content over time;
assembling the at least one media stream using the temporal sequence of screen captures; and

'074 Patent at Claim 1.

IPR349: Reply at 7-8. IPR350: Reply at 6-7. IPR351: Reply at 17-18.

# ARGUMENT: PETITIONERS TREAT WEB PAGES AS MEDIA STREAMS?

- Patent Owner misunderstands the plain and ordinary claim language – requires assembling a media stream using screen captures of a rendered web page.
- Challenged Patents state that the content can come from websites.

timedia items. Content corresponding to the plurality of multimedia items can be obtained from at least one website or cloud service that offers the content in at least one first format. Based on the obtained content, the plurality of

'074 Patent at 2:55-58.

IPR349: Reply at 7-8. IPR350: Reply at 6-7. IPR351: Reply at 17-18.

# ARGUMENT: PETITIONERS TREAT WEB PAGES AS MEDIA STREAMS?

- Patent Owner misunderstands the plain and ordinary claim language – requires assembling a media stream using screen captures of a rendered web page.
- Challenged Patents state that the content can come from websites.
- Avellan discloses this same process for viewing live content like live streaming television.

> less specifically-customized information. Accordingly, suitable sites for the broadcast mode are news sites like CNN, BBC, newspaper, live streaming television feeds, movies, music downloads, or ESPN. Those sites provide the same

Avellan at 8:18-21.

> information is not specific to the user (unlike a search site). In addition, there can be a number of broadcast channels and each channel can have a specific theme, such as country music, military television, etc. The user can then tune the browser **106** to one of the broadcast channels to receive that content. Thus, the broadcast channel can be configured here

Avellan at 4:24-29.

IPR349: Reply at 7-8. IPR350: Reply at 6-7. IPR351: Reply at 17-18.

# ARGUMENT: AVELLAN CAN ONLY TRANSMIT STATIC IMAGES, NOT MEDIA STREAMS?

- Patent Owner argues that Avellan is limited to sending compressed images.

> Avellan solely teaches imaging websites, compressing those images, and then broadcasting those compressed images to user computers. A POSITA would not recognize "broadcasting compressed website images" as a "media stream" because they lack time-synchronized playback properties, they are static data objects delivered in bulk, and they do not conform to the well-established technical usage of "media stream" in IETF, MPEG, or W3C standards. In fact, POSITA will not

IPR349: PO's Response at 56.

> Avellan merely discloses identifying a list of media items, such as image files and tag files, but does not obtain the underlying media content corresponding to those items. A person of ordinary skill in the art would understand that "obtaining content" requires retrieval of the substantive audio/video data for each multimedia item, not simply referencing or displaying descriptive information. Therefore, the

IPR349: PO's Response at 57.

# ARGUMENT: AVELLAN CAN ONLY TRANSMIT STATIC IMAGES, NOT MEDIA STREAMS?

- Avellan's gateway server can generate and transmit video data, not just static images.

In addition, web pages often have information which is rendered over a period of time, such as a video, streaming information, or an animated display. Accordingly, the gateway server **134** continuously captures images of the web page until it detects a repeating pattern of information in the web page. Thus, the gateway server **134** will be certain to capture all the information on the web page even if that information continues over a period of time. Once the page is imaged (including any audio on the web page), the gateway server **134** compresses the image or video frame format and sends it to the browser **106** via the satellite **102**.

Avellan at 5:5-15.

# ARGUMENT: AVELLAN CAN ONLY TRANSMIT STATIC IMAGES, NOT MEDIA STREAMS?

- Avellan explicitly discloses that it can stream video data, such as live streaming television.

> less specifically-customized information. Accordingly, suitable sites for the broadcast mode are news sites like CNN, BBC, newspaper, live streaming television feeds, movies, music downloads, or ESPN. Those sites provide the same

Avellan at 8:18-21.

> information is not specific to the user (unlike a search site). In addition, there can be a number of broadcast channels and each channel can have a specific theme, such as country music, military television, etc. The user can then tune the browser **106** to one of the broadcast channels to receive that content. Thus, the broadcast channel can be configured here

Avellan at 4:24-29.

IPR349: Pet. at 19; Reply at 6-8. IPR350: Pet. at 23-24; Reply at 5-6. IPR351: Pet. at 46; Reply at 21.

# ARGUMENT: THE CLAIMED BROWSER RUNS ON THE SET-TOP BOX?

- Patent Owner faults Avellan's browser for residing at the gateway.

> Avellan teaches away from a typical browser and suggests a virtual browser residing at a gateway, capturing images of a website, and then transmitting those images to a remote client (*See* EX1003) (***emphasis added***)

IPR349: PO's Sur-Reply at 13.

# ARGUMENT: THE CLAIMED BROWSER RUNS ON THE SET-TOP BOX?

- The claims do not require the rendering to be performed at the client device or set-top box.

rendering a web page by a browser using the content;
generating a temporal sequence of screen captures of the rendered web page, where each screen capture defines all the content of the web page at a given time, and at least two adjacent screen captures illustrate a dynamic change of at least a portion of the content over time;
assembling the at least one media stream using the temporal sequence of screen captures; and
providing the at least one media stream to the content provider for broadcast on the broadcast media channel.

'074 Patent at Claim 1.

IPR349: Pet. at 21; Reply at 9-10. IPR350: Pet. at 26; Reply at 8-9. IPR351: Pet. at 48; Reply at 19.

# ARGUMENT: THE CLAIMED BROWSER RUNS ON THE SET-TOP BOX?

- Patent Owner previously argued that the claimed system "allowed for simplification" of the set-top boxes.

> • The claimed solution provides a more efficient and less computationally intensive process for providing content to set-top boxes and allows for simplification of set-top boxes, unlike the prior art which required special set-top boxes for decoding and rendering

IPR349: PO's Response at 17.

- Avellan similarly states that the client device can be a "dumb terminal."

> One of the benefits of the present invention is that no local process on the computer **104** is required, other than the effective display of the web page images. The gateway **130** does all the processing, so that the computer **104** can be a dumb terminal.

Avellan at 14:11-15.

IPR349: Pet. at 21; Reply at 9-10. IPR350: Pet. at 26; Reply at 8-9. IPR351: Pet. at 48; Reply at 19.

# FARBER

**U.S. Patent No. 7,940,303**

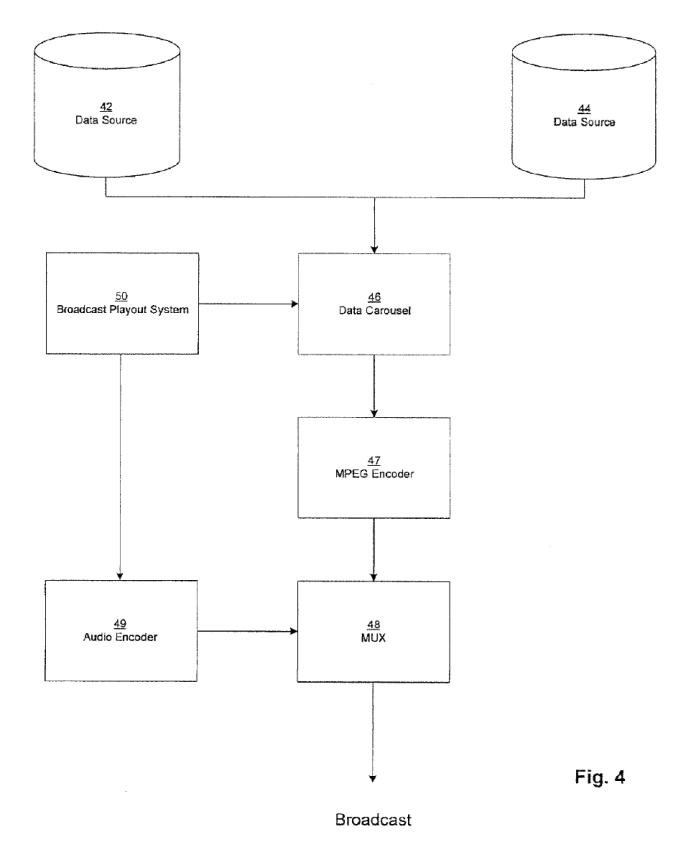
## THE ’441 PATENT: CLAIM 1

| | |
|---|---|
| **Preamble 1** | A computer-implemented method comprising: |
| **Element 1a** | creating a plurality of multimedia assets at a caching unit, for each one of the plurality of multimedia assets; |

- Farber discloses an "audio broadcast system" that performs the claimed method.
- Can cache and transmit music channel broadcasts over a network such as a cable television network or a satellite network.

> An audio broadcast system **40** is designed for transmitting various music channel broadcasts over a network such as a cable television network or a satellite network for distribution to a plurality of subscribers. It should be understood, however, that while the system described here is utilized for the broadcast of audio channels, the system is adaptable for use in broadcasting other content as well.

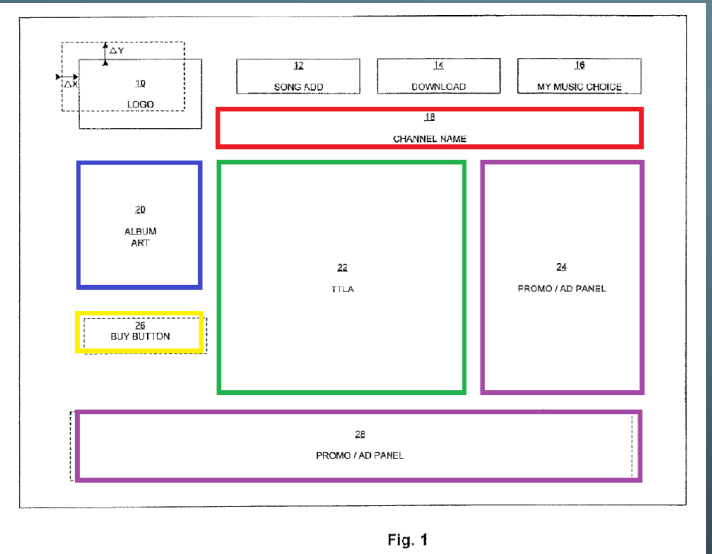
Farber at 5:9-15.



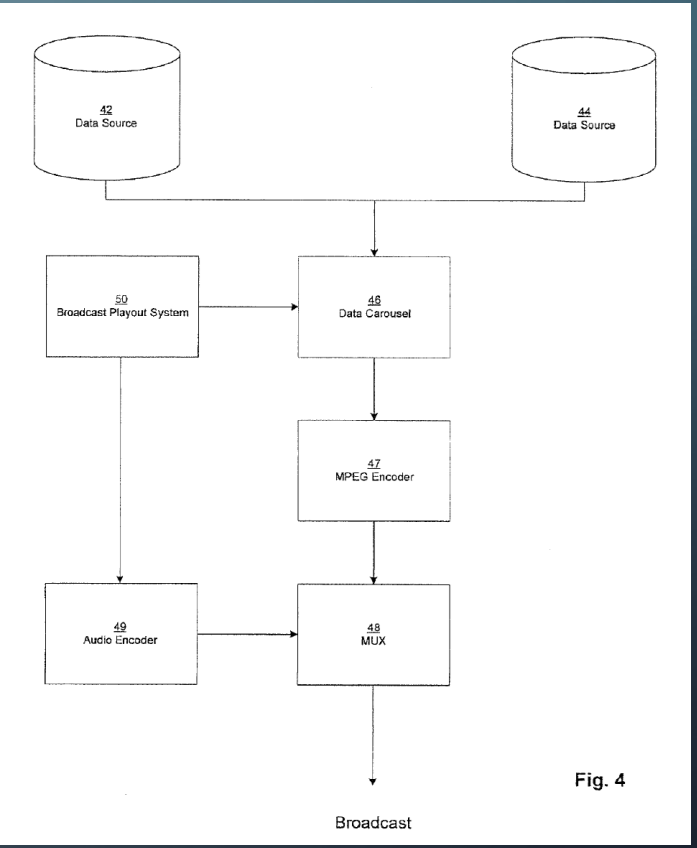

Farber at Fig. 4.

| | |
|---|---|
| **Preamble 1** | A computer-implemented method comprising: |
| **Element 1a** | creating a plurality of multimedia assets at a caching unit, for each one of the plurality of multimedia assets; |

- Each music channel includes audio data and an accompanying user interface that displays channel/song information.




Farber at Fig. 1.



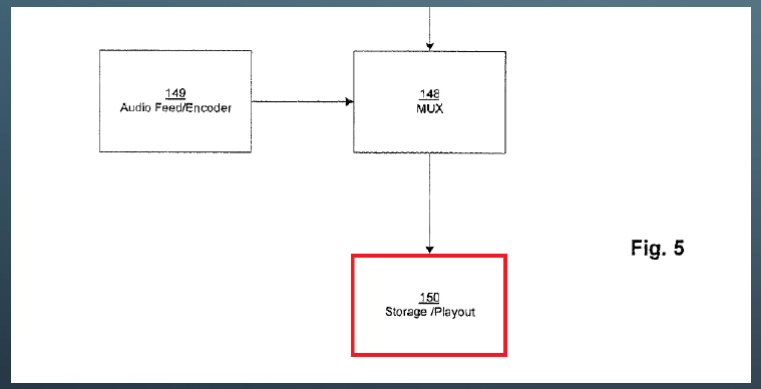

Farber at Fig. 4.

| | |
|---|---|
| **Element 1b** | determining that there is sufficient storage space for the multimedia asset at the caching unit; |

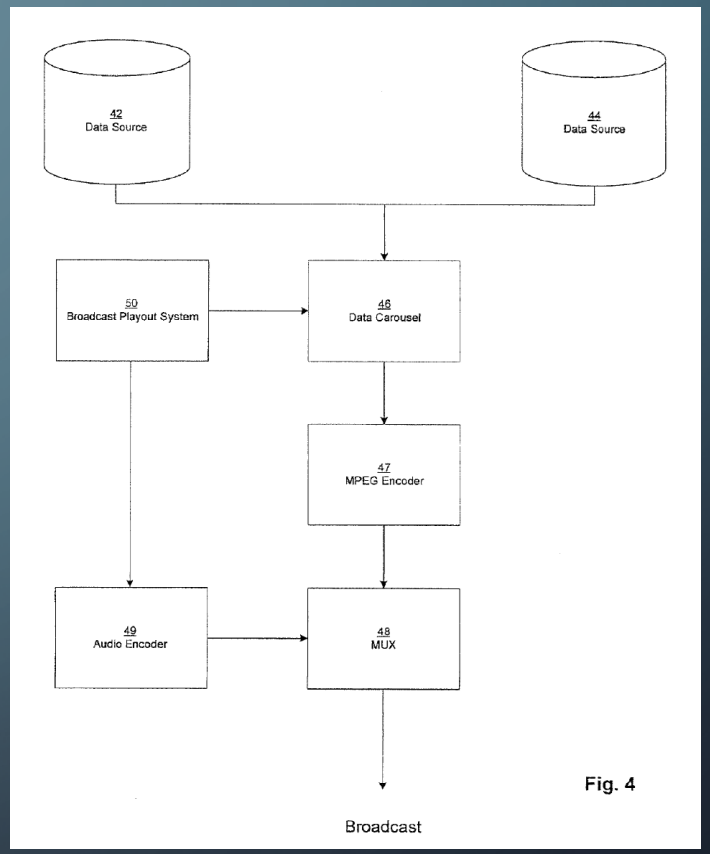
- Farber discloses an embodiment that includes storage/playout device 150.

Alternatively, in applications where playout is desired at a later time, the storage/playout device **150** may be implemented utilizing a recorder for creating or writing to any suitable storage medium such as DVD, CD ROM, hard disk, or any other suitable read only or rewritable storage medium.

Farber at 7:66-8:3.




Farber at Fig. 5.




Farber at Fig. 4.

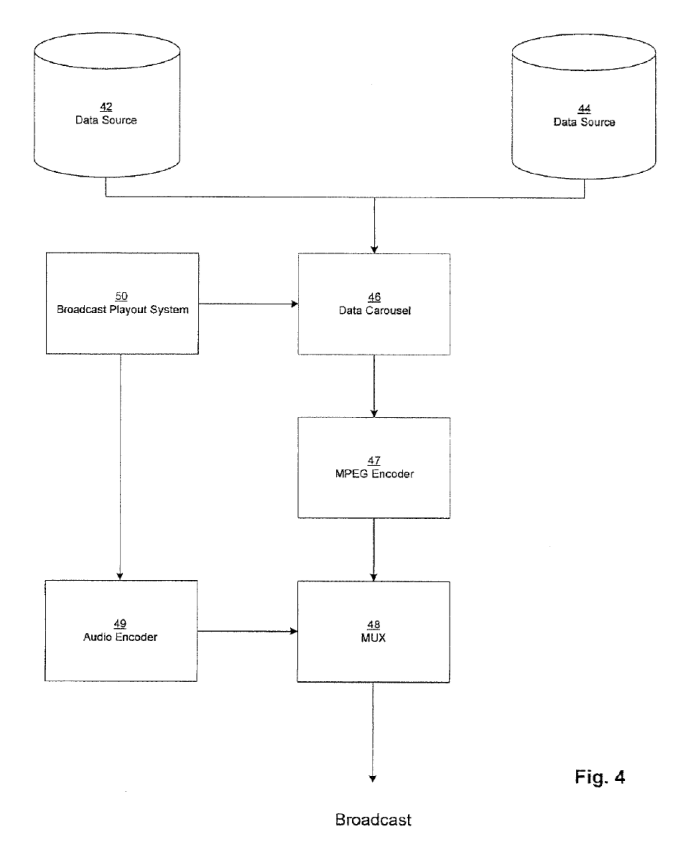
| | |
|---|---|
| **Element 1b** | determining that there is sufficient storage space for the multimedia asset at the caching unit; |

- Farber discloses an embodiment that includes storage/playout device 150.

> Alternatively, in applications where playout is desired at a later time, the storage/playout device **150** may be implemented utilizing a recorder for creating or writing to any suitable storage medium such as DVD, CD ROM, hard disk, or any other suitable read only or rewritable storage medium.

Farber at 7:66-8:3.

- By retrieving at a later time, there must have been sufficient storage space – inherent.
- The "determination" does not need to be made prior to storage.



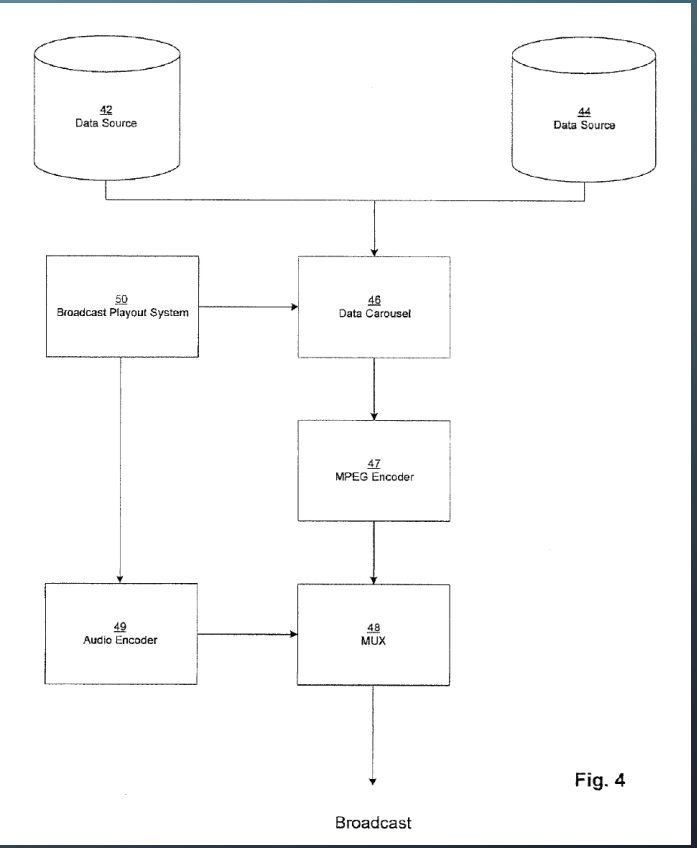

Farber at Fig. 4.

| | |
|---|---|
| **Element 1b** | determining that there is sufficient storage space for the multimedia asset at the caching unit; |

- It is obvious to combine Farber with Vermeulen.
- Vermeulen discloses determining that there is sufficient storage space prior to storage.

storage policies are satisfied. For example, in the case of writing an object **30** as outlined above, nodepicker **130** may operate to develop a write plan, or a particular sequence of nodes **160** to which the object **30** should be written. In developing a particular write plan, nodepicker **130** may be configured to ensure that the write plan has a reasonable chance of succeeding—for example, that the nodes **160** specified in the write plan are in fact operational and are expected to have sufficient storage resources available to accept the object **30**—and that the write plan, if completed, would satisfy all storage policies pertinent to write operations. Example write

Vermeulen at 14:64-15:7.

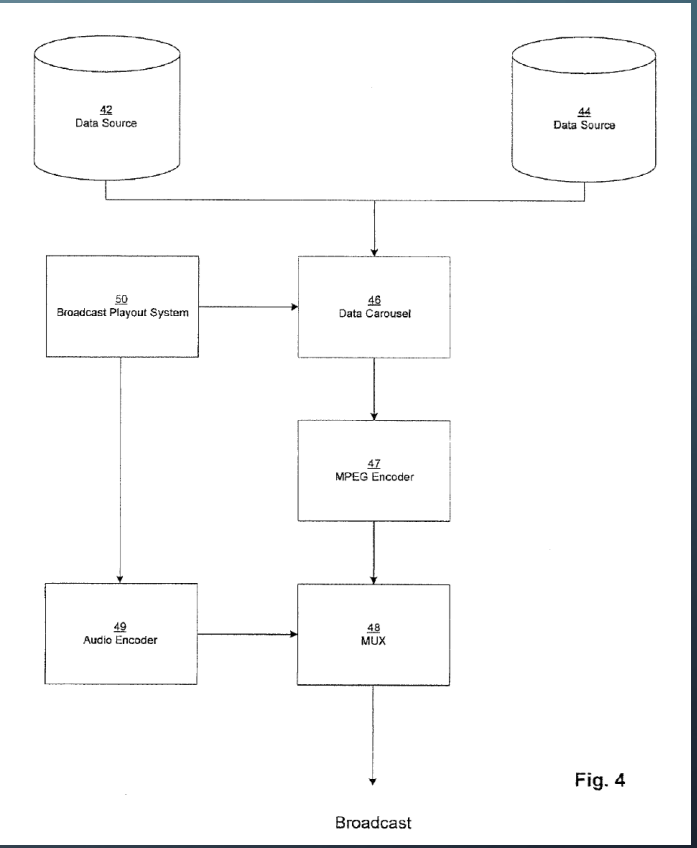



Farber at Fig. 4.

IPR351: Pet. at 29-30; Reply at 13.

| | |
|---|---|
| **Element 1b** | determining that there is sufficient storage space for the multimedia asset at the caching unit; |

- It is obvious to combine Farber with Vermeulen.
- Vermeulen discloses determining that there is sufficient storage space prior to storage.
- Verifying that there is sufficient storage space is well-known and common in the art.
- POSITAs would readily recognize that it is advantageous to check for available space.




Farber at Fig. 4.

IPR351: Pet. at 29-30; Reply at 13.

| Element 1c | generating a channel identifier for the multimedia asset; |
|---|---|

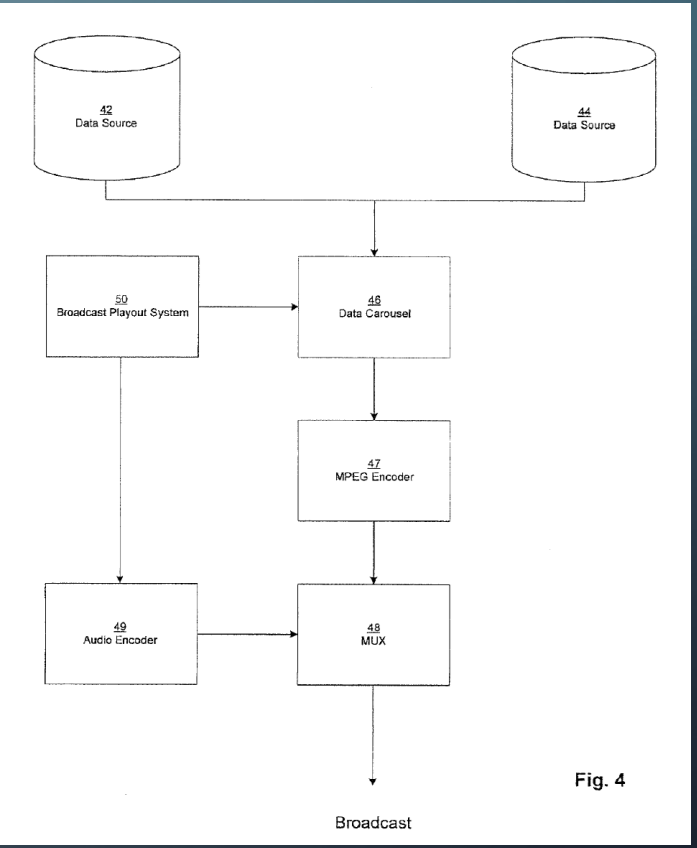
- Farber discloses generating numerical channel identifiers associated with music genre.

object tags "<Grobj>". The screen tag may include a number of attributes, including height ("hgt"), which may be used to identify the height of the screen, width ("wid"), which may be used to identify the width of the screen, and channel ("chan"), which may be used to identify the name of the broadcast music channel with which the template data is associated.

Farber at 6:39-44.

```
<?xml version="1.0"?>
<!DOCTYPE Scrn SYSTEM "file:///data/mc/MusicChoice.dtd">
<Scrn hgt="480" wid="704"
chan="001" chName="Soft Rock" Seq="02262">
<IntWin xPos="200" yPos="90" xSz="390" ySz="236"
BgCl="0x1E1E1E" FgCl="0xC4C3C8" InCl="0xB71E16"
BgSl="0xB71E16" BtSl="0xB71E16" BdSl="0xC8C800" />
```

Farber at Fig. 6.




Farber at Fig. 4.

| Element 1c | generating a channel identifier for the multimedia asset; |
|---|---|

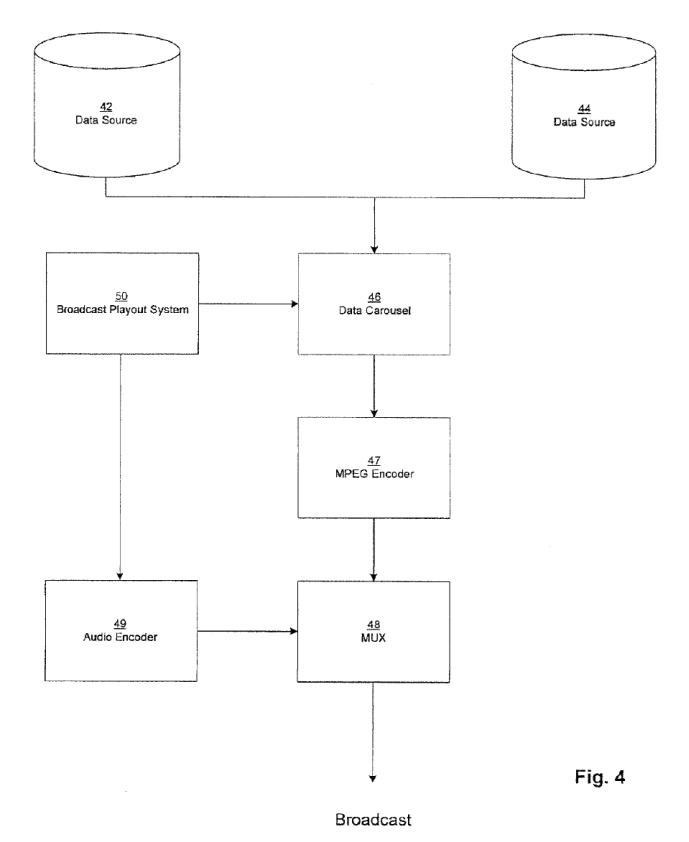
- The ’441 Patent also uses numerical channel identifiers associated with music genre.

video multimedia items. In some instances, the request can include a channel identifier that is associated with a particular category of multimedia items, such as a music genre corresponding to rock, classical, jazz, etc.

’441 Patent at 7:56-59.

Several indexes or identifiers by the name “ChID,” (these integers preferably no more than an integer value from 1 to the maximum number of channels) begin (e.g. MAX could be 50) to be generated by the caching unit **510**. This method

’441 Patent at 6:52-55.




Farber at Fig. 4.

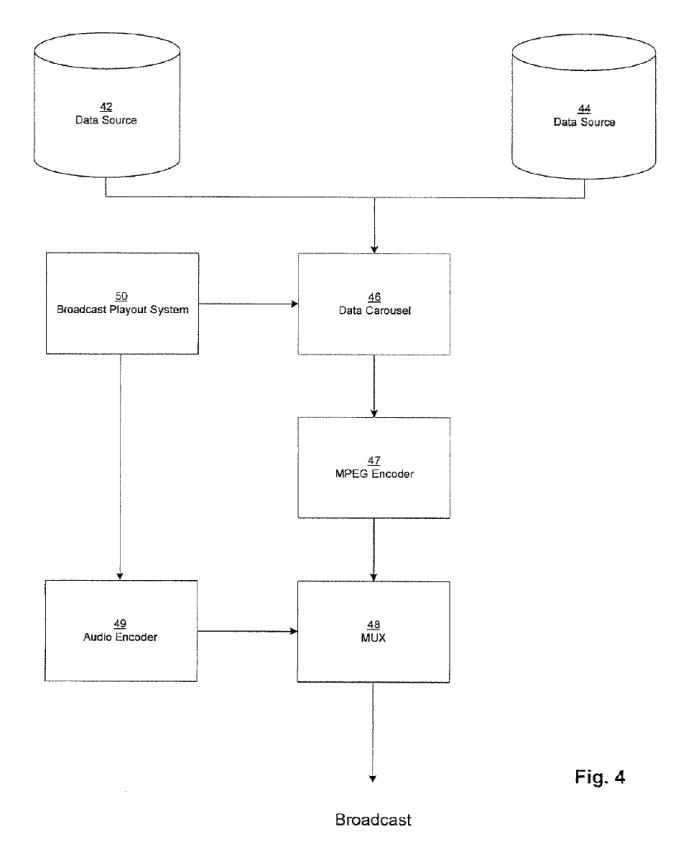
IPR351: Pet. at 18-19; Reply at 7.

| | |
|---|---|
| **Element 1d** | retrieving from a cloud service a plurality of media files to be included in the multimedia asset including a media identifier for each one of the media files; |

- Each of Farber's music channels include a plurality of songs.

by an event. For example, in an embodiment for transmitting a broadcast music channel, screen arrangements may change from screen **1** to screen **2** to screen **3** and other subsequent screens at the beginning of each new song being broadcast.

Farber at 4:33-36.




Farber at Fig. 4.

| | |
|---|---|
| **Element 1d** | retrieving from a cloud service a plurality of media files to be included in the multimedia asset including a media identifier for each one of the media files; |

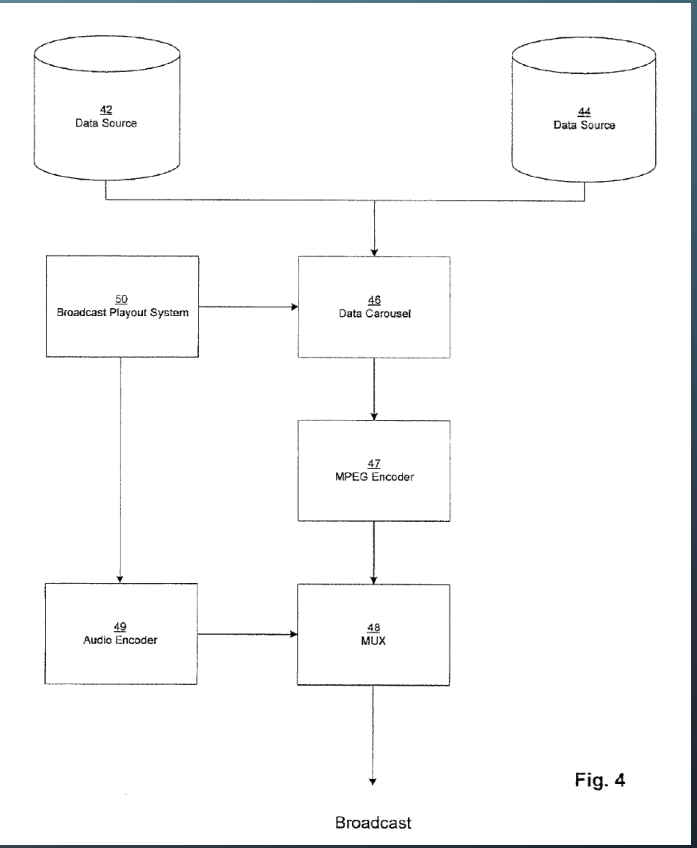
- Each of Farber's music channels include a plurality of songs.
- For each song, Farber retrieves audio content and the associated album art (e.g., ".jpg" file).

In the embodiment shown in FIG. **6**, the template data is associated with a given broadcast music channel and the audio work that is currently being broadcast on the music channel. The template data may consist of a number of tags

Farber at 6:33-36.

```
pos="2" />
<Grobj src="mcAA1730327.jpg" desc="Elton John&apos;s Greatest
Hits 1970-2002"
xPos="501" yPos="338" xSz="100" ySz="100" wid="100" hgt="100"
pos="3" />
<Grobj src="mcBA506412.jpg" Billable="N"
```

Farber at Fig. 6.



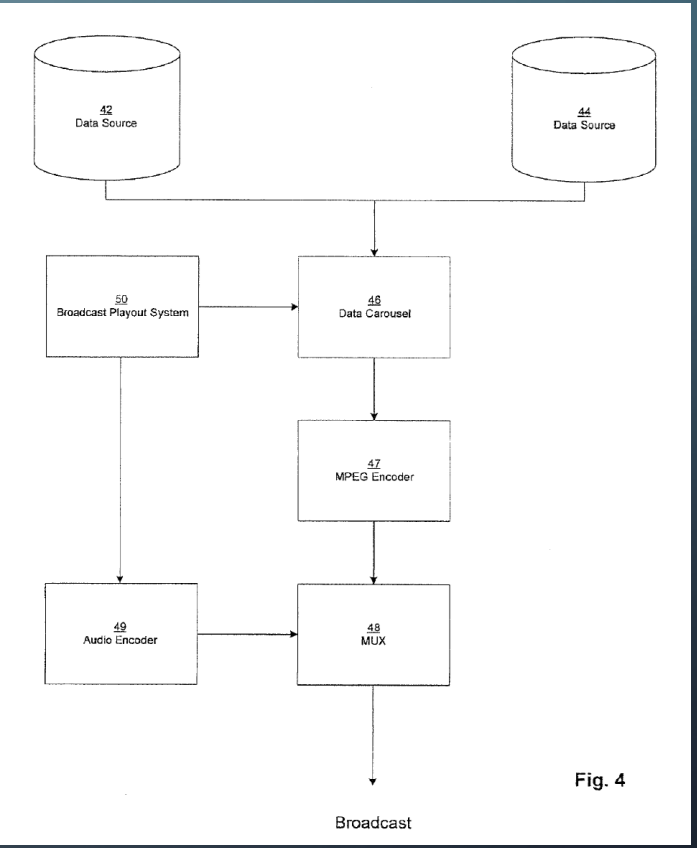

Farber at Fig. 4.

| | |
|---|---|
| **Element 1d** | retrieving from a cloud service a plurality of media files to be included in the multimedia asset including a media identifier for each one of the media files; |

- Farber discloses that the media identifiers can be numbers and/or simple strings.

```
<Grobj
xPos="214" yPos="92" xSz="214" ySz="251" wid="214" hgt="251"
pos="1" >
 <SongTitle value="Are You Ready For Love (2003)"/>
 <Artist value="Elton John"/>
 <RecordLabel value="ULTRA"/>
 <CD value="Elton John&apos;s Greatest Hits 1970-2002"/>
 <UniqID value="3310722"/>
</Grobj>
<Grobj src="mcBT_BuyPrevious_1.jpg" desc="Buy Previous Button"
xPos="209" yPos="314" xSz="129" ySz="37" wid="129" hgt="37"
pos="2" />
<Grobj src="mcAA1730327.jpg" desc="Elton John&apos;s Greatest
Hits 1970-2002"
xPos="501" yPos="338" xSz="100" ySz="100" wid="100" hgt="100"
pos="3" />
<Grobj src="mcBA506412.jpg" Billable="N"
xPos="336" yPos="128" xSz="320" ySz="192" wid="320" hgt="192"
pos="4" />
```

Farber at Fig. 6.




Farber at Fig. 4.

| | |
|---|---|
| **Element 1d** | retrieving from a cloud service a plurality of media files to be included in the multimedia asset including a media identifier for each one of the media files; |

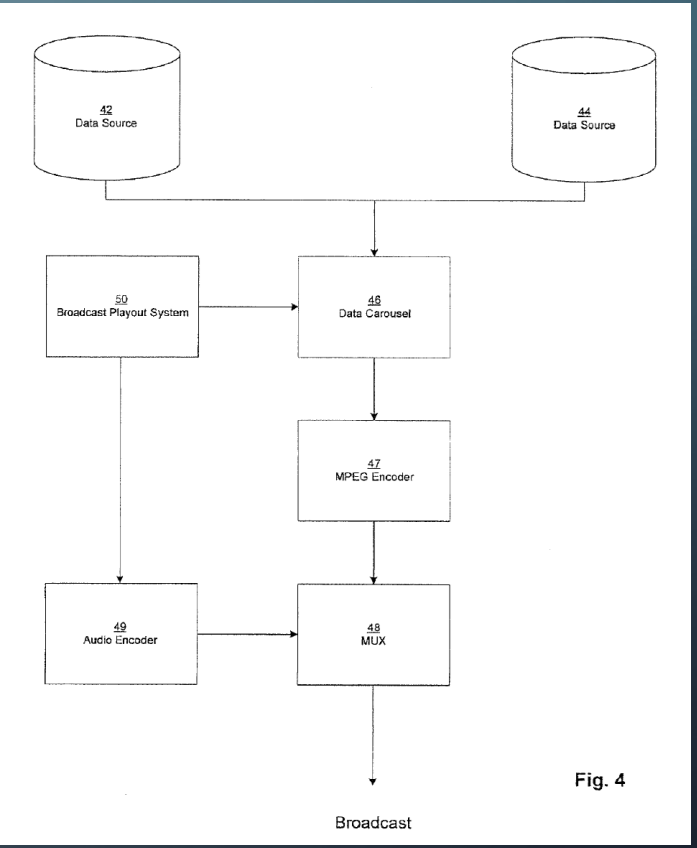
- The ’441 Patent also uses numbers or simple strings for the media identifiers.

Channel ID and a Playlist Name. Each new file synchronized with the Mediamplify cloud has a Media ID value which the cloud can assign, and include additional metadata information such as “Artist” and “Author.” The “Mediamplify

’441 Patent at 9:10-13.

{“song_encoded”: “ts”, “artist”: “Madonna”, “duration”: 0, “exists_on_disk”: True, “genre”: “Pop”, “title”: “Like a Virgin”, “music_id”: “1082863”}

’441 Patent at 9:27-29.




Farber at Fig. 4.

IPR351: Pet. at 21; Reply at 7.

| | |
|---|---|
| **Element 1d** | retrieving from a cloud service a plurality of media files to be included in the multimedia asset including a media identifier for each one of the media files; |

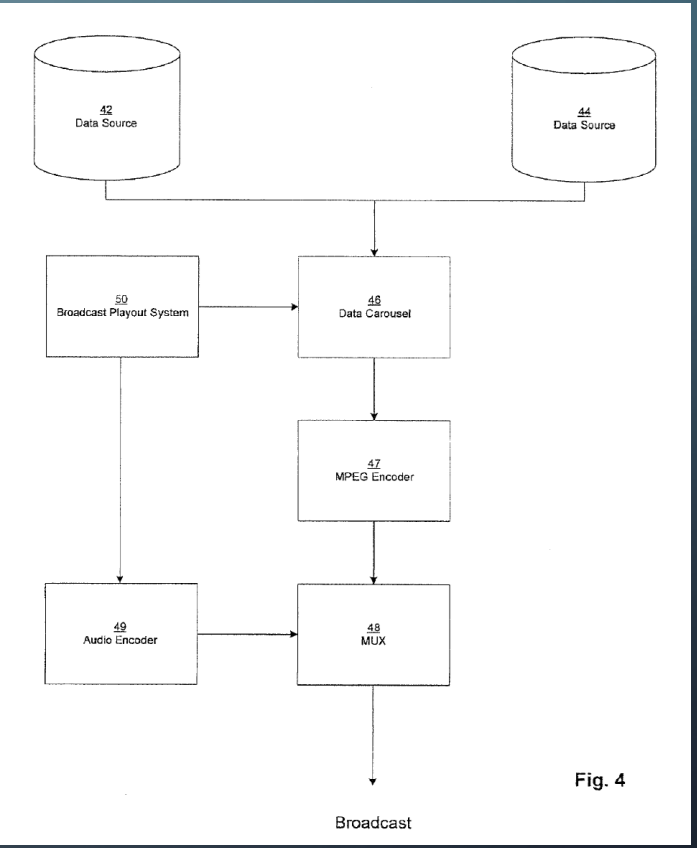
- Farber's data sources can be accessible remotely – i.e., cloud-based.

It should be understood that while two data sources are shown here, a single data source or a plurality of data sources may be connected to or in communication with the data carousel subsystem **46**. The data sources **42**, **44** may or may not be located in a single location and may or may not be co-located with the data carousel subsystem **46**.

Farber at 5:33-38.

The second data source **44** may include, for example, information to support the album art asset **20**, the title and track label and artist asset **22**, and information to support other assets. The second data source **44** may also be associated with and connected to a server or other computer for managing the database and/or facilitating communications with the data carousel subsystem **46**.

Farber at 5:26-32.




Farber at Fig. 4.

| | |
|---|---|
| **Element 1e** | for each one of the media files, creating a custom hypertext transfer markup language (HTML) user interface that includes video; |

- For each song, Farber discloses creating a custom HTML template.

which may be an MPEG encoder **47**. In response to the trigger, the encoder **47** pulls template information, images, and text from the data carousel subsystem **46** and creates, based on the template information, a screen having embedded assets such as those shown in screens **1**, **2** or **3** which are associated with the currently queued song. The screen may be a video frame, such as an MPEG video frame.

Farber at 6:2-8.

sources **42**, **44**. The data carousel subsystem **46** is capable of managing/creating the templates, which may be coded utilizing HTML, XML or other suitable protocols for creating templates/screens having text and other assets such as those described above.

Farber at 5:50-54.

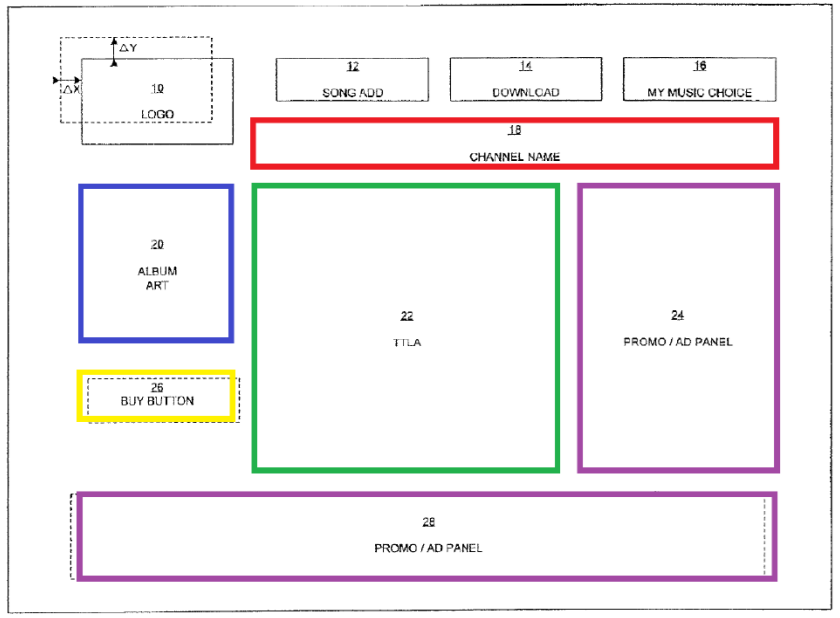



Fig. 1

Farber at Fig. 1.

IPR351: Pet. at 22-24; Reply at 9-10.

| | |
|---|---|
| **Element 1e** | for each one of the media files, creating a custom hypertext transfer markup language (HTML) user interface that includes video; |

- Farber's templates are user interfaces that include buttons and other assets.

FIG. **1** shows a screen **1** which is displayed coincident with the transmission of a broadcast music channel. The screen **1** contains various assets including a logo **10**, a song add button **12**, a download button **14**, a personalized channel choice button **16**, a channel name **18**, album art **20**, title track label and artist information **22**, promotional/advertising panels **24**, **28** and a buy button **26**.

Farber at 3:38-44.

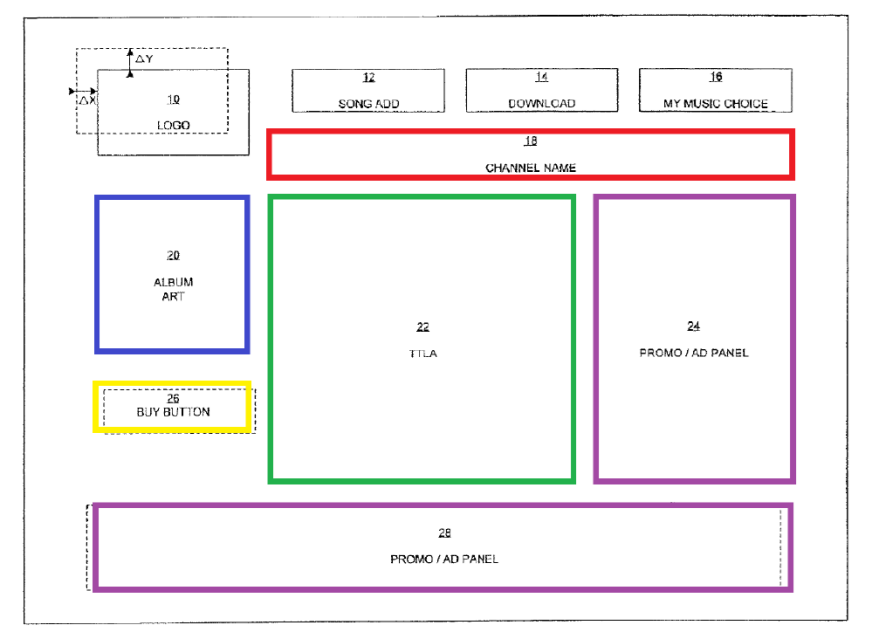



Fig. 1

Farber at Fig. 1.

| | |
|---|---|
| **Element 1e** | for each one of the media files, creating a custom hypertext transfer markup language (HTML) user interface that includes video; |

- Farber's templates can include video.

"graphic object". These assets may include, but are not limited to, textual information, motion picture video, graphics, control features such as buttons or pull down menus, promotional materials, or other control mechanisms. Referring first

Farber at 3:30-33.

contain graphical content. The promotion/advertising panels **24, 28** may contain a combination of graphic and textual content and may alternatively contain motion picture video content.

Farber at 3:48-51.

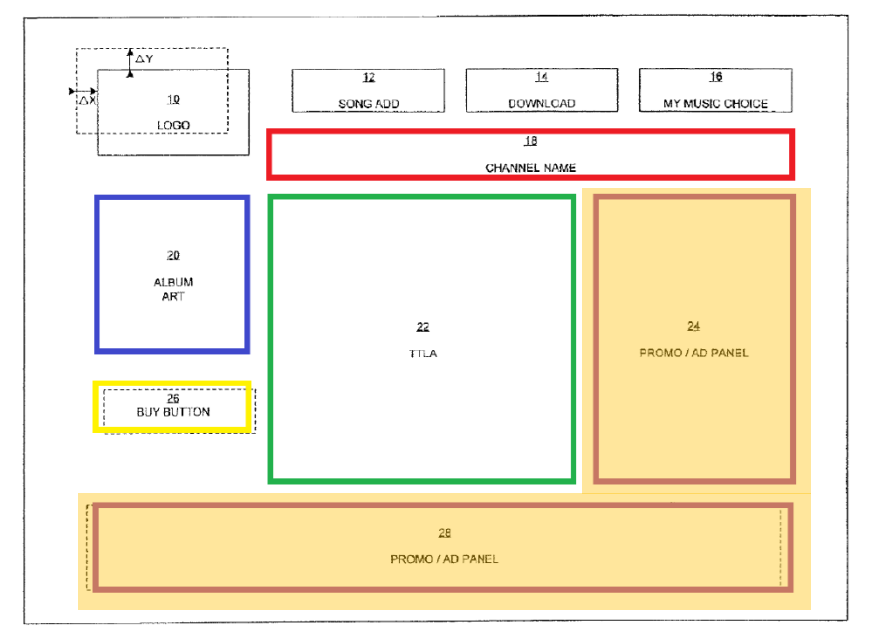



Fig. 1

Farber at Fig. 1.

IPR351: Pet. at 22-24; Reply at 9-10.

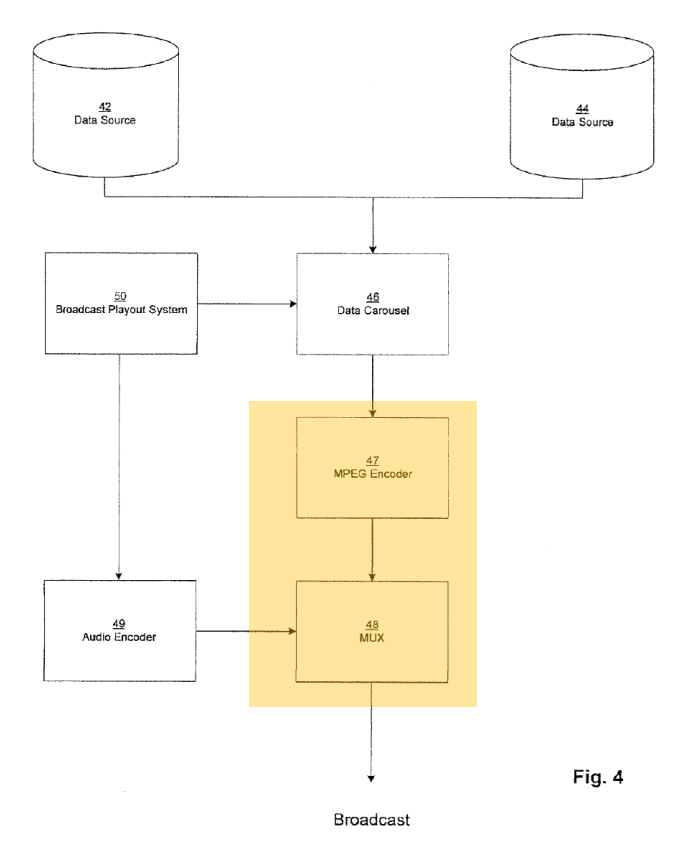
| | |
|---|---|
| **Element 1f** | encoding each of the media files and encapsulating them together using an MPEG transport stream format; |

- Farber encodes and encapsulates the media files using an MPEG transport stream.

It should be understood that this process may be executed for a plurality of channels simultaneously. The MPEG encoder **47** then creates an MPEG transport stream for all channels with the embedded MPEG video frames. The MPEG encoder **47** may be implemented utilizing a commercially available encoder or a general-purpose computer.

Based upon a trigger from the broadcast playout system **50**, a multiplexer **48** receives the MPEG transport stream from the MPEG encoder **47** and simultaneously receives an encoded audio feed from an audio encoder **49**. The multiplexer **48** serves to combine the audio feed from the audio encoder **49** and the associated encoded MPEG transport stream from the MPEG encoder **47**. The multiplexer **48** then feeds the multiplexed signal out for broadcast in the form of a video transport stream to a distribution system, such as the head end of service provider (e.g., a cable television network or a satellite network provider as is well known in the art).

Farber at 6:9-25.




Farber at Fig. 4.

| Element 1g | storing each one of the multimedia assets at the caching unit; |
|---|---|

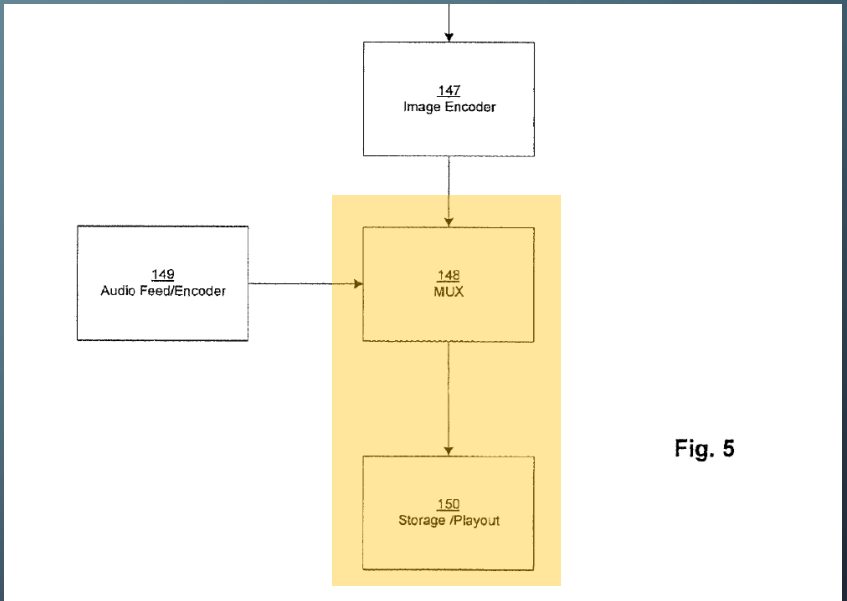
- Farber discloses storing the assets in storage/playout device 150.

A multiplexer **148** serves to combine the audio feed/encoder output **149** with the image encoder output **147** to create a transport stream at its output. It should be understood that

Farber at 7:51-53.

The output of the multiplexer **148** is fed to a storage/playout device **150**. The storage/playout device **150** may be implemented utilizing any display or transmission device which is capable of displaying or transmitting video images.

Farber at 7:62-65.



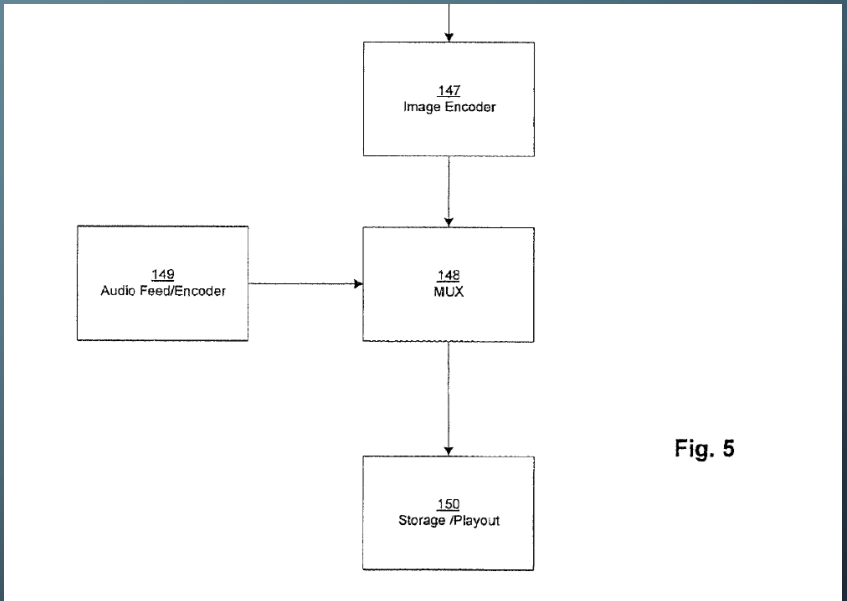

Farber at Fig. 5.

| | |
|---|---|
| **Element 1h** | receiving a request at the caching unit for one of the multimedia assets from a broadcasting unit; and |
| **Element 1i** | responsive to receiving the request, the caching unit providing the multimedia asset to the broadcasting unit in the MPEG transport stream format. |

- Farber discloses receiving a request from a broadcasting unit.

Alternatively, in applications where playout is desired at a later time, the storage/playout device **150** may be implemented utilizing a recorder for creating or writing to any suitable storage medium such as DVD, CD ROM, hard disk, or any other suitable read only or rewritable storage medium.

Farber at 7:66-8:3.




Farber at Fig. 5.

- Also obvious to combine Vermeulen.

Various embodiments of a distributed, web-services based storage system are disclosed. According to one embodiment, a system may include a web services interface configured to receive, according to a web services protocol, client requests for access to data objects. A given client request for access to

Vermeulen at 2:15-19.

IPR351: Pet. at 27-28 and 30; Reply at 11-12.

| | |
|---|---|
| **Element 1h** | receiving a request at the caching unit for one of the multimedia assets from a broadcasting unit; and |
| **Element 1i** | responsive to receiving the request, the caching unit providing the multimedia asset to the broadcasting unit in the MPEG transport stream format. |

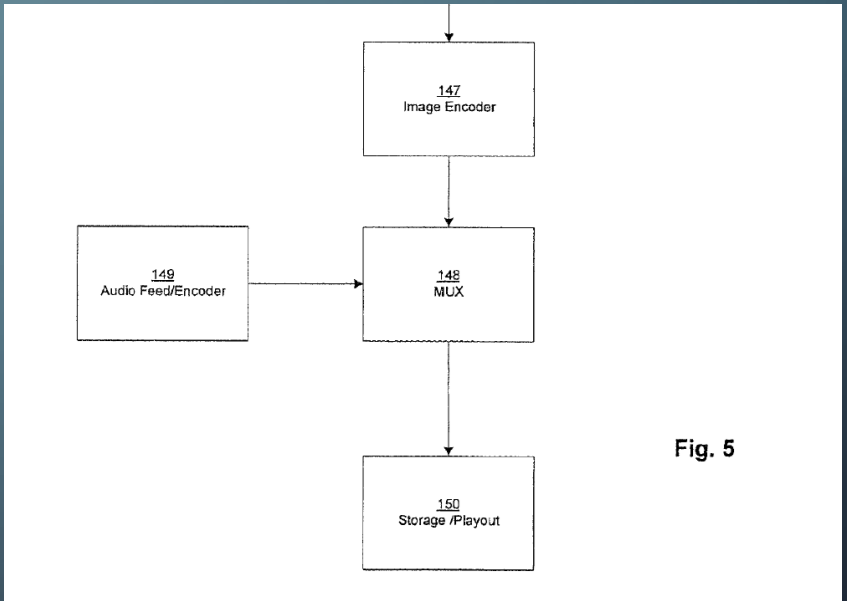
- Farber discloses providing the asset to the broadcasting unit in the MPEG transport stream format.

stored templates to reduce screen burn-in. It should also be understood that the system **140** is capable of simultaneously transmitting, playing out, or storing such content for later play back utilizing the methods disclosed above.

Farber at 8:22-8:25.

MPEG encoder **47**. The multiplexer **48** then feeds the multiplexed signal out for broadcast in the form of a video transport stream to a distribution system, such as the head end of service provider (e.g., a cable television network or a satellite network provider as is well known in the art).

Farber at 6:21-25.




Farber at Fig. 5.

# FARBER

## U.S. Patent No. 7,940,303

THE ’441 PATENT: CLAIMS 2-9

# FARBER AND THE ’441 PATENT

- In general, the Challenged Claims add well-known and common features.
  - The use of a REST API (claim 2)
  - The use of JavaScript or a WebKit browser or engine (claims 3, 6, 9)
  - Encoding at a “specified bit rate” (claim 4)
  - The use of common standards, such as H.264 or AC-3 (claims 5, 8)
  - The inclusion of video (claim 7)

# FARBER AND THE ’441 PATENT

- In general, the Challenged Claims add well-known and common features.
- Patent Owner does not contend he invented any of these features, which were all routinely implemented by POSITAs.
- Those features are either directly disclosed in Farber and/or plainly disclosed in the cited secondary references.

# PATENT OWNER'S ATTEMPTS TO DISTINGUISH FARBER ARE UNAVAILING

# ARGUMENT: FARBER'S EMBODIMENTS ARE LIMITED TO A CDROM OR A CD PLAYER?

- Patent Owner argues that Farber's disclosure of playing streams out "at a later time" is limited to a CDROM or using a CD Player.

> Farber does not need to determine space as CDs, DVDs, PCMIA slots, have a fixed space size. CD ROMs a few hundred Megabytes for example. The retrieving "at a later time" argument at Petition, 17 is questionable. A POSITA will know that "at a later time" means with a CDROM, using a CD Player connected to a display.

IPR351: PO's Response at 21.

# ARGUMENT: FARBER'S EMBODIMENTS ARE LIMITED TO A CDROM OR A CD PLAYER?

- Farber explicitly discloses use of rewritable storage technologies and is not limited to CDROMs, CD Players, or other Read Only Memory.

> Alternatively, in applications where playout is desired at a later time, the storage/playout device **150** may be implemented utilizing a recorder for creating or writing to any suitable storage medium such as DVD, CD ROM, hard disk, or any other suitable read only or rewritable storage medium.

Farber at 7:66-8:3.

IPR351: Pet. at 16-17; Reply at 7.

# ARGUMENT: FARBER DOES NOT USE THE MEDIA IDENTIFIER TO RETRIEVE MEDIA FILES?

- Patent Owner argues that the claimed "media identifier" must be used for the retrieval of the media file.

> Using Media Identifiers in Music Platform is extremely important, speciallywhen retrieving content cloud-service, there is more control of media files for royalties and music rights. Hence, although a data souce could be "remote," the Petition, 21-22 does not show that the cloud-service system uses the media identifier for the retrieval process, nor such a media file.

IPR351: PO's Response at 23.

# ARGUMENT: FARBER DOES NOT USE THE MEDIA IDENTIFIER TO RETRIEVE MEDIA FILES?

- The plain and ordinary claim language does not require the media identifier to be "used" in the retrieval process – only that it is retrieved.

1. A computer implemented method, comprising:
creating a plurality of multimedia assets at a caching unit,
for each one of the plurality of multimedia assets:
determining that there is sufficient storage space for the multimedia asset at the caching unit;
generating a channel identifier for the multimedia asset;
retrieving from a cloud service a plurality of media files to be included in the multimedia asset including a media identifier for each one of the media files;
for each one of the media files, creating a custom hypertext transfer markup language (HTML) user interface that includes video;

'441 Patent at Claim 1.

IPR351: Pet. at 20-21; Reply at 8.

# ARGUMENT: FARBER DOES NOT USE THE MEDIA IDENTIFIER TO RETRIEVE MEDIA FILES?

- Farber retrieves media identifiers from a data source, which may be co-located on the cloud.

channel. The template data may consist of a number of tags (e.g., XML tags or other tags). In the given example, the data includes a screen tag "<Scrn>" and a number of graphic object tags "<Grobj>". The screen tag may include a number

Farber at 6:36-39.

shown in FIG. **6**, the graphic object tag may include the following tags: songtitle, artist, recordlabel, CD value and uniqID.

Farber at 6:59-61.

arrangements of screens **1**, **2**, and **3**. As an alternative, the template information may be housed in one of the data sources **42**, **44**. The data carousel subsystem **46** is capable of

Farber at 5:48-50.

```
<Grobj
xPos="214" yPos="92" xSz="214" ySz="251" wid="214" hgt="251"
pos="1" >
 <SongTitle value="Are You Ready For Love (2003)"/>
 <Artist value="Elton John"/>
 <RecordLabel value="ULTRA"/>
 <CD value="Elton John&apos;s Greatest Hits 1970-2002"/>
 <UniqID value="3310722"/>
</Grobj>
<Grobj src="mcBT_BuyPrevious_1.jpg" desc="Buy Previous Button"
xPos="209" yPos="314" xSz="129" ySz="37" wid="129" hgt="37"
pos="2" />
<Grobj src="mcAA1730327.jpg" desc="Elton John&apos;s Greatest
Hits 1970-2002"
xPos="501" yPos="338" xSz="100" ySz="100" wid="100" hgt="100"
pos="3" />
<Grobj src="mcBA506412.jpg" Billable="N"
xPos="336" yPos="128" xSz="320" ySz="192" wid="320" hgt="192"
pos="4" />
```

Farber at Fig. 6.

IPR351: Pet. at 20-22; Reply at 7.

# ARGUMENT: FARBER DOES NOT DISCLOSE HOW AN HTTP SERVER WORKS?

- Patent Owner argues that Farber does not disclose custom HTML user interfaces because it "does not disclose how an HTTP server will work."

agree. Farber fails to show "creating a custom hypertext markup language (HTML) user interface that includes video," as Farber does not disclose how an HTTP server will work.

the '441 patent teaches that for a "Custom UI" or a "User Interface" is in use an HTTP Server needs to be present. Farber does not disclose one, hence

IPR351: PO's Response at 23.

# ARGUMENT: FARBER DOES NOT DISCLOSE HOW AN HTTP SERVER WORKS?

- The plain and ordinary claim language does not require an HTTP server.

1. A computer implemented method, comprising:
creating a plurality of multimedia assets at a caching unit,
for each one of the plurality of multimedia assets:
determining that there is sufficient storage space for the multimedia asset at the caching unit;
generating a channel identifier for the multimedia asset;
retrieving from a cloud service a plurality of media files to be included in the multimedia asset including a media identifier for each one of the media files;
for each one of the media files, creating a custom hypertext transfer markup language (HTML) user interface that includes video;

'441 Patent at Claim 1.

IPR351: Pet. at 22; Reply at 9-10.

# ARGUMENT: FARBER DOES NOT DISCLOSE HOW AN HTTP SERVER WORKS?

- The plain and ordinary claim language does not require an HTTP server.
- The '441 Patent specification does not even require the use of HTTP.

> The caching unit can communicate with Multicast-01 **215** and Multicast-02 **220**. The two Multicast units **215**, **220** can pull or obtain information from the caching unit via TCP/IP or UDP, using protocols such as HTTP (Hypertext Transport Protocol) or RTSP (Real-time Streaming Transport). The

'441 Patent at 5:6-10.

> The caching unit can add a "Custom UI," or a customizable User Interface that is designed using Hypertext Markup Language (e.g. HTML) or a web service. A web service that matches a REST interface (Representation State Transfer) and uses Javascript Object Notation (JSON) or XML (Extensible Markup Language) to respond to a custom UI request can be used. Moreover, this custom UI can

'441 Patent at 6:66-7:5.

IPR351: Pet. at 22; Reply at 9-10.

# ARGUMENT: FARBER DOES NOT DISCLOSE HOW AN HTTP SERVER WORKS?

- The plain and ordinary claim language does not require an HTTP server.
- The specification does not even require the use of HTTP.
- To the extent use of HTTP or an HTTP GET request would be advantageous (see Claim 2), this was commonly known and readily implemented by POSITAs.

IPR351: Pet. at 32-33; Reply at 14.

# ARGUMENT: FARBER DOES NOT DISCLOSE AN MPEG ENCODER?

- Patent Owner argues that Farber does not disclose an MPEG encoder and only works on MPEG transport streams.

> if accessing the CD ROM took place, Farber only works on MPEG Transport Streams, and is silent on encoding. Hence as the Image Encoder outputs MPEG Transport Streams (Ex-2039¶139)

IPR351: PO's Response at 24.

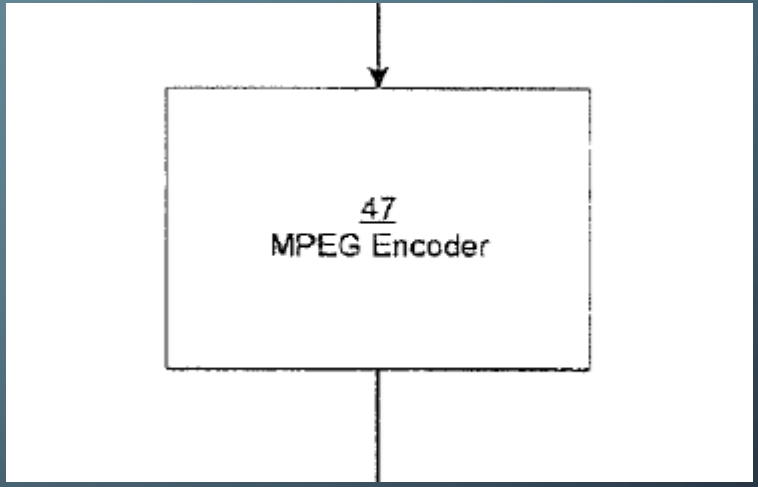
# ARGUMENT: FARBER DOES NOT DISCLOSE AN MPEG ENCODER?

- Farber explicitly discloses "MPEG Encoder 47" that encodes "MPEG video frames."

carousel subsystem **46** generates a trigger to an encoder **47**, which may be an MPEG encoder **47**. In response to the trigger, the encoder **47** pulls template information, images, and text from the data carousel subsystem **46** and creates, based on the template information, a screen having embedded assets such as those shown in screens **1**, **2** or **3** which are associated with the currently queued song. The screen may be a video frame, such as an MPEG video frame.

It should be understood that this process may be executed for a plurality of channels simultaneously. The MPEG encoder **47** then creates an MPEG transport stream for all channels with the embedded MPEG video frames. The MPEG encoder **47** may be implemented utilizing a commercially available encoder or a general-purpose computer.

Farber at 6:1-14.




Farber at Fig. 4.

IPR351: Pet. at 24-25; Reply at 10.

# ARGUMENT: FARBER USES "TRIGGERS"?

- Patent Owner argues that Farber is a trigger-based machine and that "receiving a request" will disrupt the process.

Farber is a stateful, state machine, that triggers from one state to the next. E.g., Audio Encoder sends a trigger to MPEG encoder EX1003, ¶¶115-116. Assuming Farber is working on Elton, John's song, it must wait for it to end, to trigger starting on a new song. As such, if either when Elton John is being "multiplexed" or while is transitioning, a "request at the caching unit: arrives, the process will be disrupted. This is derived from incompatible models, stateless using HTTP REST APIs, and stateful as Farber's state machine.

IPR351: PO's Response at 25.

# ARGUMENT: FARBER USES "TRIGGERS"?

- The claim language is silent on "stateless" versus "stateful" implementations.
- It is also silent on whether the current song/process needs to finish – *i.e.*, is not disrupted.
- Claim language only requires providing the multimedia asset to the broadcasting unit in response to receiving the request.

receiving a request at the caching unit for one of the multimedia assets from a broadcasting unit; and

responsive to receiving the request, the caching unit providing the multimedia asset to the broadcasting unit in the MPEG transport stream format.

'441 Patent at Claim 1.

IPR351: Reply at 11-12.

# ARGUMENT: FARBER USES "TRIGGERS"?

- Indeed, the Challenged Patents state that the use of a "stateless" REST API is optional.

> The caching unit can add a "Custom UI," or a customizable User Interface that is designed using Hypertext Markup Language (e.g. HTML) or a web service. A web service that matches a REST interface (Representation State Transfer) and uses Javascript Object Notation (JSON) or XML (Extensible Markup Language) to respond to a custom UI request can be used. Moreover, this custom UI can

'441 Patent at 6:66-7:5.

- To the extent use of a REST API would be advantageous (see Claim 2), this was commonly known and readily implemented by POSITAs.

IPR351: Pet. at 32-33; Reply at 14.

# WANNAMAKER

U.S. Patent Pub. No. 2004/0031052

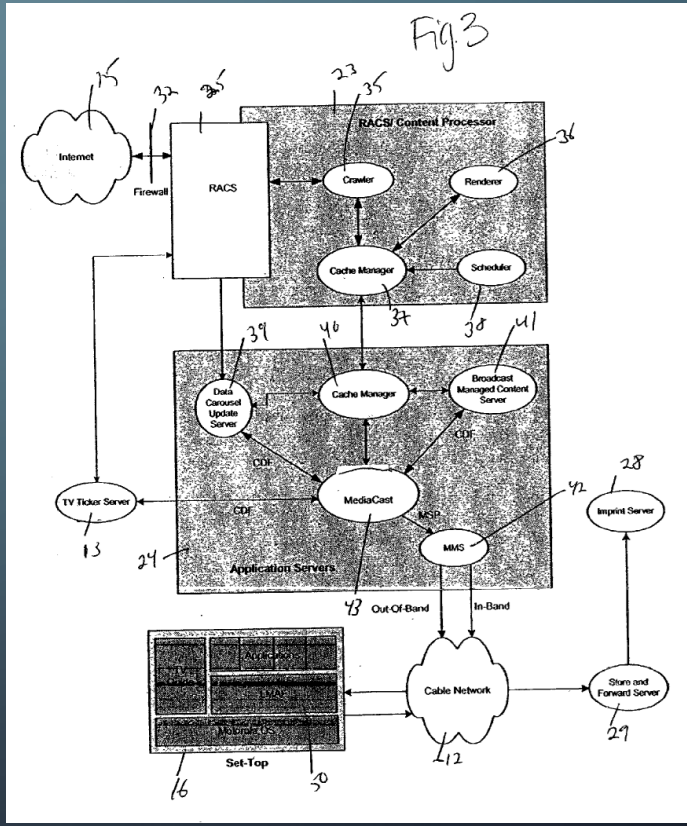
## THE ’002 PATENT: CLAIM 7

| | |
|---|---|
| **[7 pre]** | A system, comprising: |
| **[7a]** | a multicast server configured to generate parameters relating to at least one of video content, image content and audio content, and configured to output the parameters via MPEG outputs; |

- Wannamaker discloses a system that delivers web content to cable television subscribers.

> television applications. The information platform mediates the communication between an abundance of Web-based content and the television set. Content and applications are

Wannamaker at [0098].




Wannamaker at Fig. 3.

| | |
|---|---|
| **[7 pre]** | A system, comprising: |
| **[7a]** | a multicast server configured to generate parameters relating to at least one of video content, image content and audio content, and configured to output the parameters via MPEG outputs; |

- The Application Server 24 has a Multimedia Streamer (MMS) for multicasting content.

sent out-of-band. Generally, the MMS_OM1000 is used to send OOB directories and files, subjecting all data to rate-throttling to prevent buffer overflow in the set-top boxes.

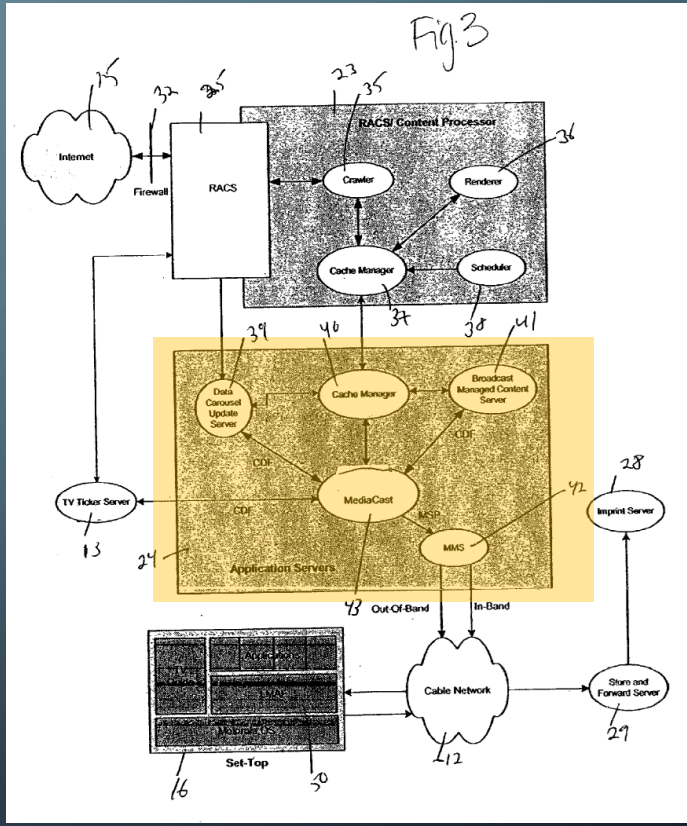
**[0229]** The data may be:

**[0230]** Broadcast to all set-top boxes

**[0231]** Multicast to a group of set-top boxes

**[0232]** Single-cast or "addressed" to a single set-top box

Wannamaker at [0231].




Wannamaker at Fig. 3.

| | |
|---|---|
| **[7 pre]** | A system, comprising: |
| **[7a]** | a multicast server configured to generate parameters relating to at least one of video content, image content and audio content, and configured to output the parameters via MPEG outputs; |

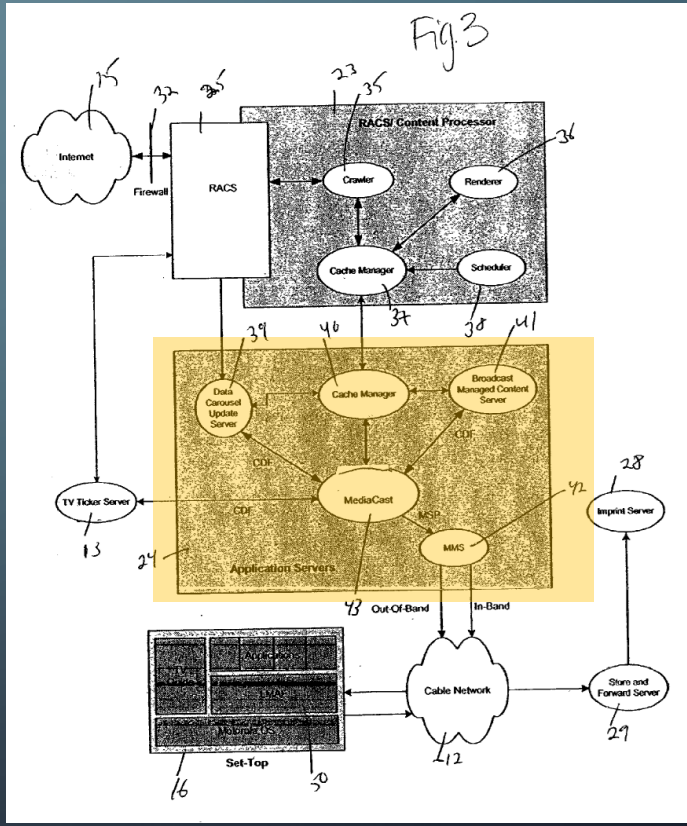
- The server's cache manager generates parameters relating to the requested content.
- Message includes parameters to resolve the URL request and obtain the content.

**[0237]** The cache manager checks the local rendered cache for the content and notifies the application if the content is present. If the content is not local, the cache manager must resolve the URL. To resolve the URL, the cache manager sends a message to all cache managers on content processor hosts that have been configured to resolve URL requests.

Wannamaker at [0237].

**[0509]** There are many ways to encode information in a URL. Because the content feed server is the only piece of code interpreting the URL, it makes sense that the content feed provider define their own URL syntax.

Wannamaker at [0509].



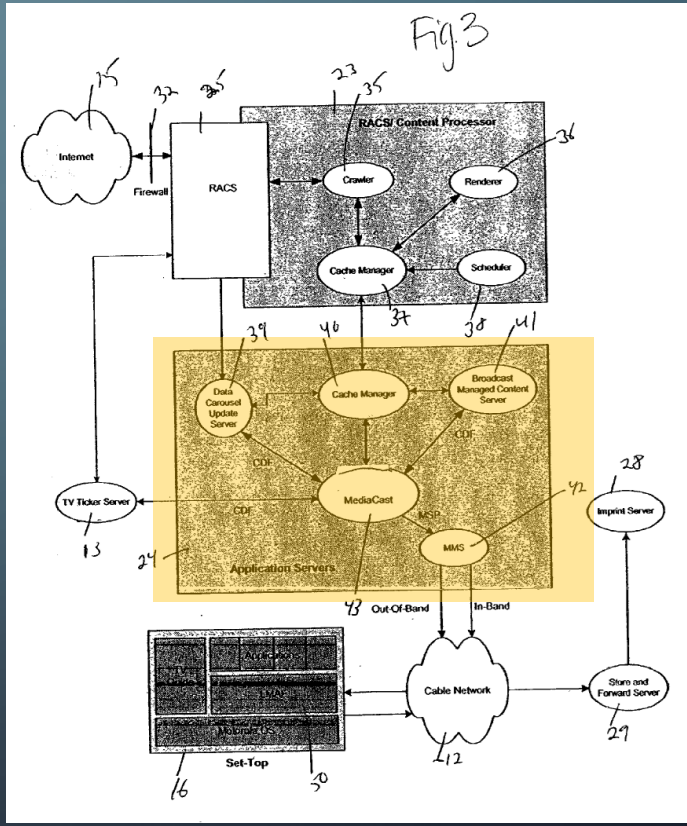

Wannamaker at Fig. 3.

| | |
|---|---|
| **[7 pre]** | A system, comprising: |
| **[7a]** | a multicast server configured to generate parameters relating to at least one of video content, image content and audio content, and configured to output the parameters via MPEG outputs; |

- Can also include parameters specifying the content's modification date.

**[0254]** The crawler is intelligent about fetching data from the Internet. When instructed to fetch an HTML document, the crawler only fetches the document if it is newer than the one on the application server, based on the modification dates provided by the Web server. When the crawler checks the modification dates of the content referred to by the HTML document, it loads only those files that are absent or whose modification dates are newer than the versions on application server. If the assets have changed, they are

Wannamaker at [0254].



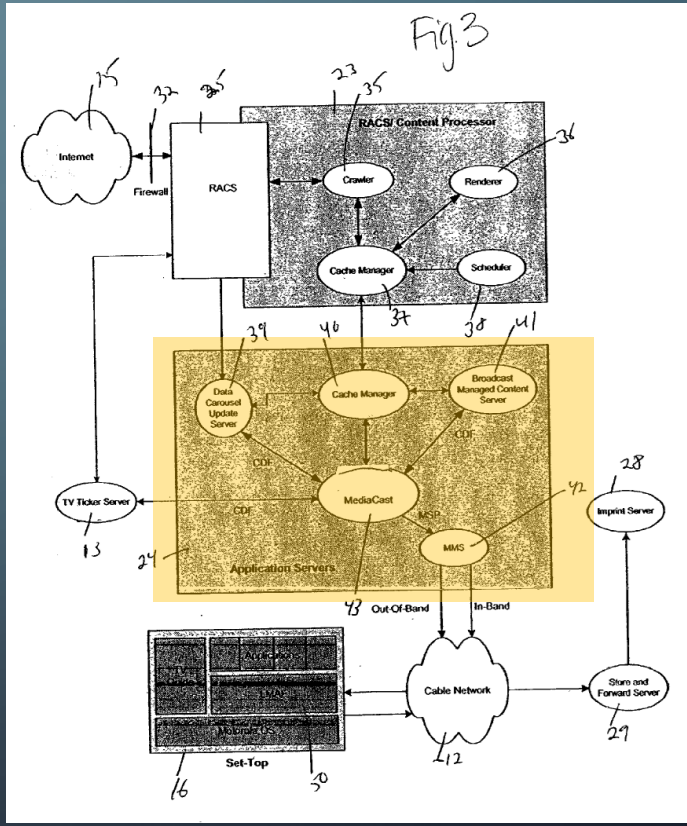

Wannamaker at Fig. 3.

| | |
|---|---|
| **[7 pre]** | A system, comprising: |
| **[7a]** | a multicast server configured to generate parameters relating to at least one of video content, image content and audio content, and configured to output the parameters via MPEG outputs; |

- Can also include parameters relating to the content's layout size.

uses the batch mode only. The renderer also supports a layout preview mode that emulates a system running the microbrowser. All the pages for this mode are limited to a single screen of information. All pages longer that a single screen are clipped at the bottom. In this mode, there is additional information displayed below the screen capture. It includes the size (in bytes) of each asset, produced by this page, that are broadcast by the Mediacast server and whether the background I-frame is modified or not.

Wannamaker at [0261].




Wannamaker at Fig. 3.

| | |
|---|---|
| **[7 pre]** | A system, comprising: |
| **[7a]** | a multicast server configured to generate parameters relating to at least one of video content, image content and audio content, and configured to output the parameters via MPEG outputs; |

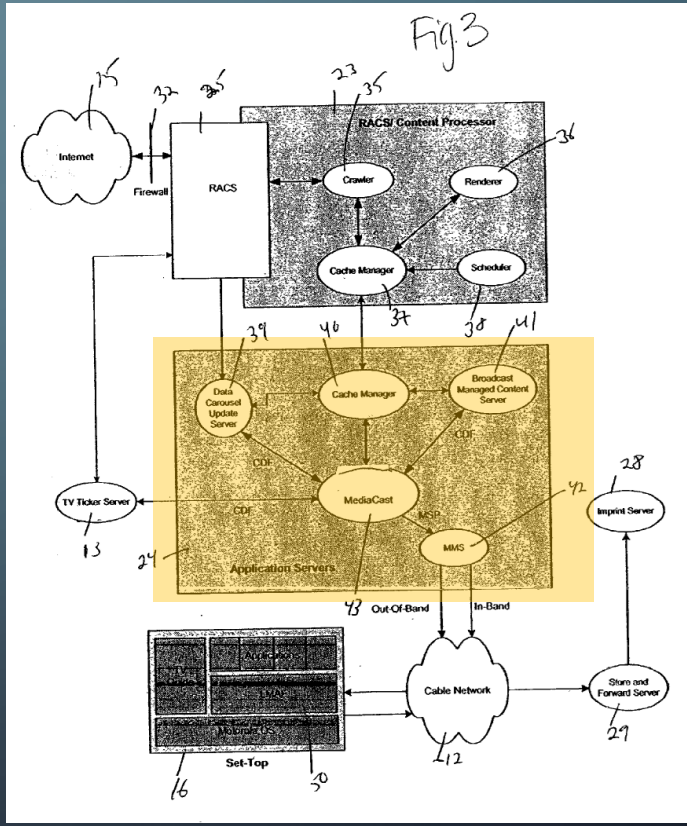
- Can also include parameters relating to the content's layout size.

**[0264]** After layout, the information platform renderer quickly and effectively compresses the output buffer to MPEG syntax in the FMB file format. All dynamic objects,

Wannamaker at [0264].

pixel block boundaries where necessary. The FMB file is guaranteed to have the same width as the TV screen and the height is guaranteed to be at least the height of the screen.

Wannamaker at [0264].




Wannamaker at Fig. 3.

| | |
|---|---|
| **[7 pre]** | A system, comprising: |
| **[7a]** | a multicast server configured to generate parameters relating to at least one of video content, image content and audio content, and configured to output the parameters via MPEG outputs; |

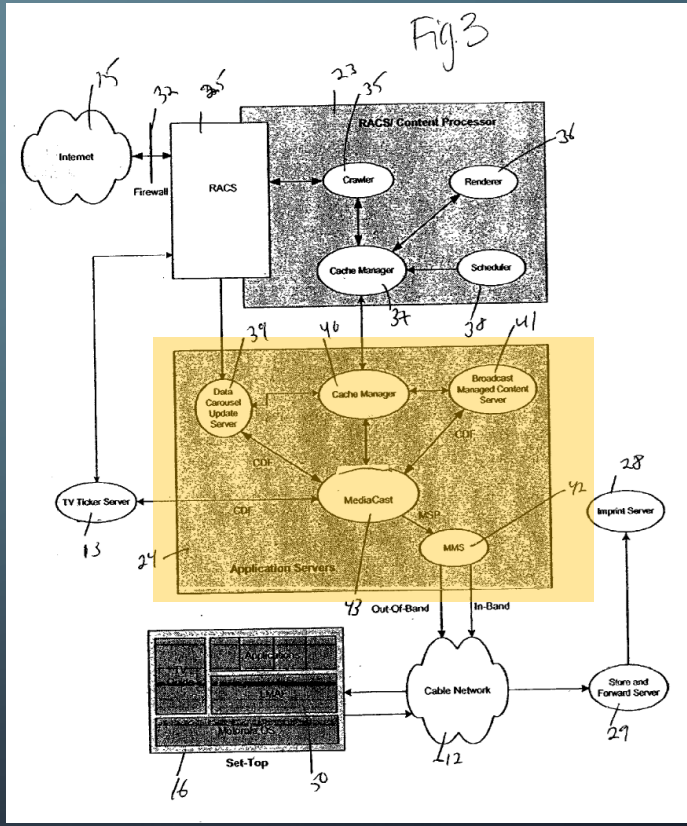
- The Application Server 24 is configured to output the parameters via MPEG outputs.

**[0193]** A multimedia streamer (MMS) consists of the audio/video/terminal server (AVTS) that delivers the units of audiovisual information sent by the system to the user's set-top box. It also contains the out-of-band servers (the

Wannamaker at [0193].

**[0195]** The AVTS is responsible for delivering MPEG video and Dolby AC-3 audio streams to the set-top box. A special MPEG multiplexer (MUX) card, controlled by the AVTS, streams data onto the cable network.

Wannamaker at [0195].




Wannamaker at Fig. 3.

| | |
|---|---|
| [7b] | a caching server communicatively coupled to the multicast server, |
| [7c] | and configured to receive and store the parameters, |

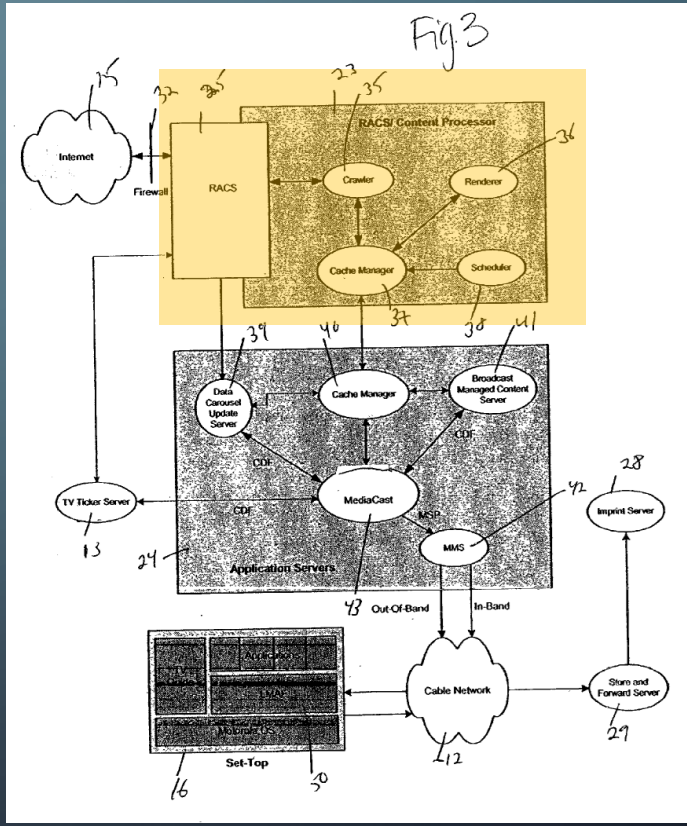
- The Content Processor 23 is coupled to the Application Server 24.
- Produces and caches content requested by the Application Server.

[0244] A content processor (CP) cluster produces digital cable system compatible (rendered) content from standard Web content retrieved from the Internet. Multiple content

Wannamaker at [0244].

[0236] Cache managers **37** (see **FIG. 3**) are responsible for accepting requests for content and handling the steps required to deliver the content to the necessary location. Once content is pulled from the Internet, the crawler **35** deposits the unrendered content (HTML, JPEG) into source cache. The rendered **36** cache is a storage area for content

Wannamaker at [0236].



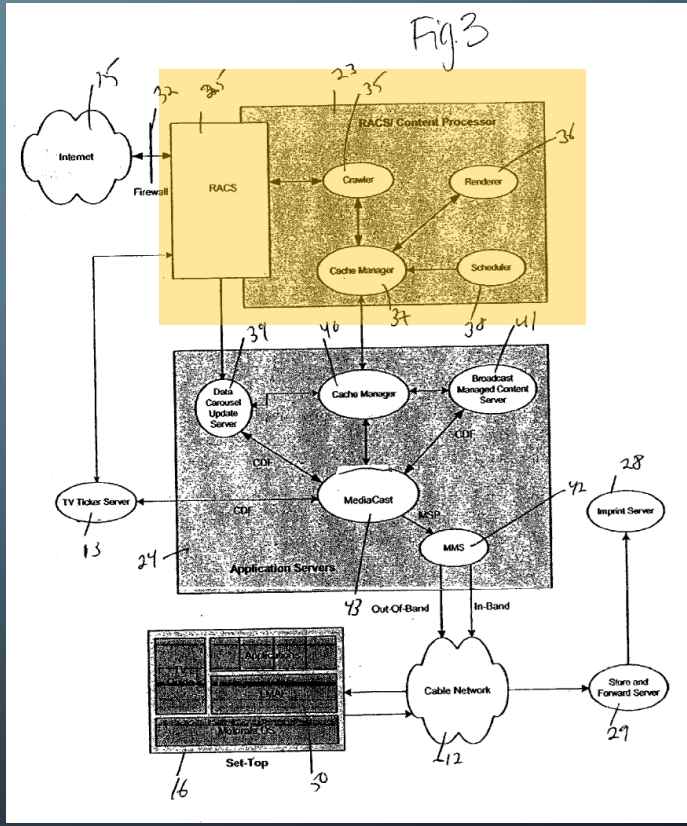

Wannamaker at Fig. 3.

| | |
|---|---|
| [7b] | a caching server communicatively coupled to the multicast server, |
| [7c] | and configured to receive and store the parameters, |

- The Content Processor 23 is coupled to the Application Server 24.
- Produces and caches content requested by the Application Server.

[0237] The cache manager checks the local rendered cache for the content and notifies the application if the content is present. If the content is not local, the cache manager must resolve the URL. To resolve the URL, the cache manager sends a message to all cache managers on content processor hosts that have been configured to resolve URL requests.

Wannamaker at [0237].




Wannamaker at Fig. 3.

| | |
|---|---|
| **[7d]** | create a temporal sequence of screen captures of a rendered web page, where each said screen capture defines content of the rendered web page at a given time, and at least two adjacent screen captures in the temporal sequence illustrate a dynamic change of at least a portion of the video content, image content or audio content over time, |

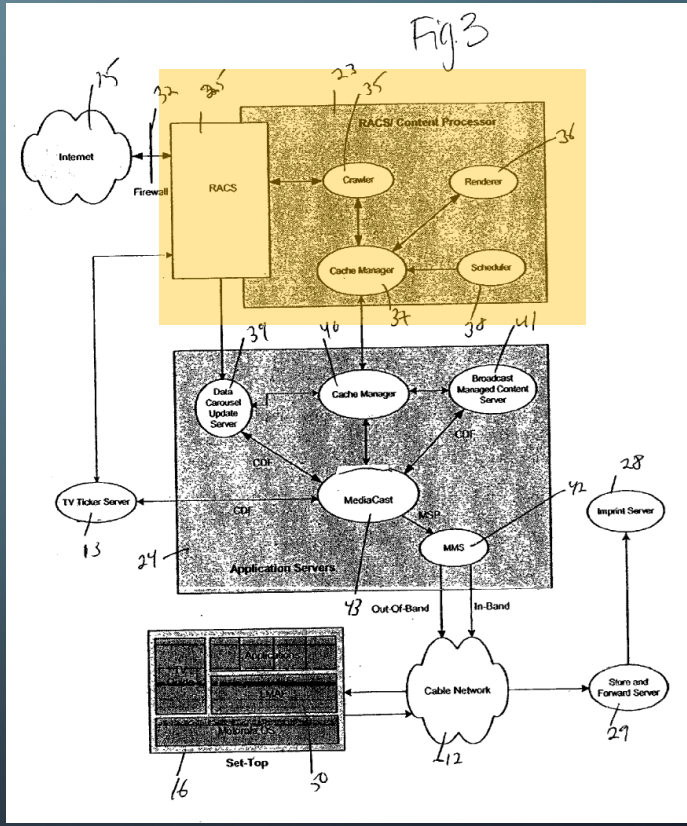
- The Content Processor's renderer 36 produces rendered web content.

**[0244]** A content processor (CP) cluster produces digital cable system compatible (rendered) content from standard Web content retrieved from the Internet. Multiple content

Wannamaker at [0244].

**[0259]** Most content requires rendering before it can be displayed on the set-top box. The renderer converts the content into a format compatible with the user's set-top box hardware. It does this by accepting URLs of HTML files that are stored locally, and then parsing and rendering the HTML to produce output image files (FMB) and output CPML or MML.

Wannamaker at [0259].




Wannamaker at Fig. 3.

| | |
|---|---|
| **[7d]** | create a temporal sequence of screen captures of a rendered web page, where each said screen capture defines content of the rendered web page at a given time, and at least two adjacent screen captures in the temporal sequence illustrate a dynamic change of at least a portion of the video content, image content or audio content over time, |

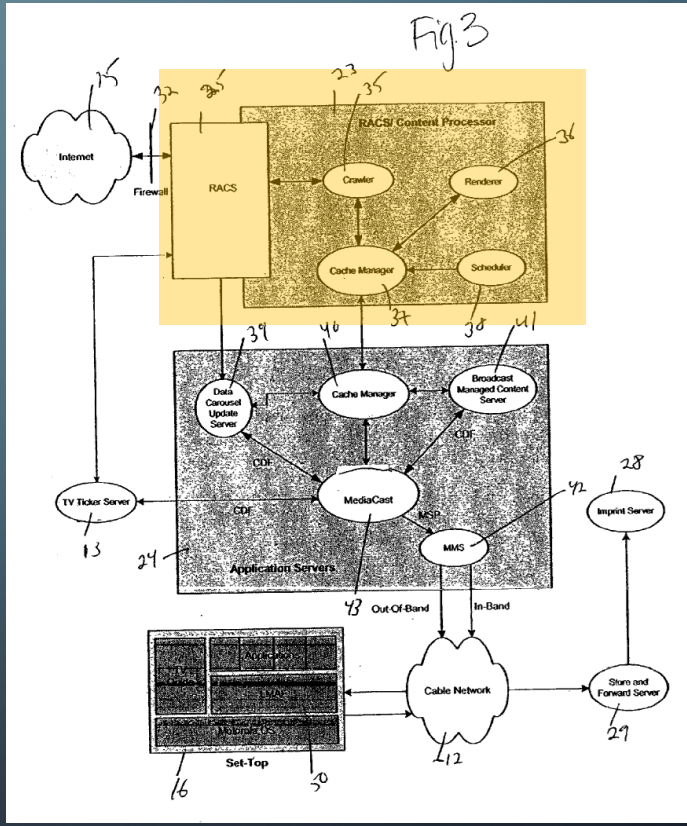
- Rendered content includes animated images and animated GIFs.

**[0136]** The information platform supports HTML 3.2, including text, tables, forms, images, animated images, image maps, frames, and background images. HTML has been extended to support Interactive TV applications.

Wannamaker at [0136].

**[0098]** The information platform converts Internet content and protocols into forms suitable for TV viewing. The content is fetched from Web sites, converted or rendered by the system, and then cached on application server The information platform supports content from standard Web sites, e.g. HTML source, GIF, animated GIF, JPEG, and PNG files. The majority of HTML 3.2 tags are supported,

Wannamaker at [0098].



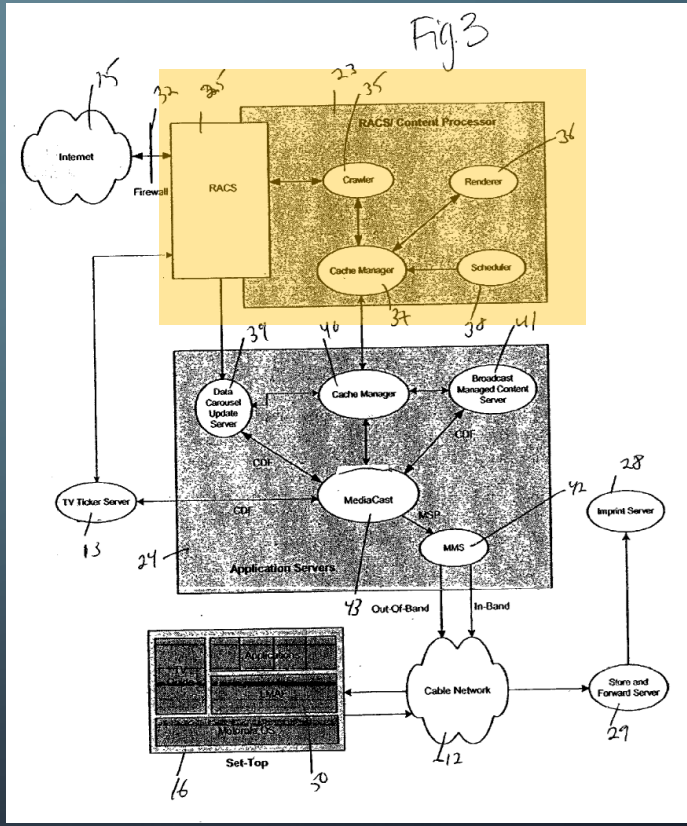

Wannamaker at Fig. 3.

| | |
|---|---|
| **[7d]** | create a temporal sequence of screen captures of a rendered web page, where each said screen capture defines content of the rendered web page at a given time, and at least two adjacent screen captures in the temporal sequence illustrate a dynamic change of at least a portion of the video content, image content or audio content over time, |

- Animated content is captured as separate frames – *i.e.*, a sequence of captures.

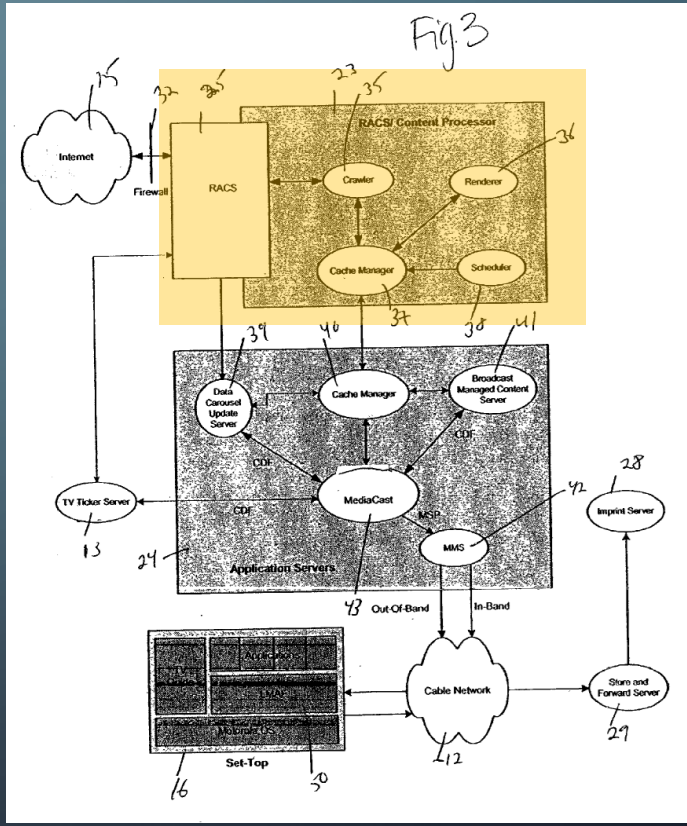
> given CPML file. Taking this example a step farther, if there is an animated GIF on the same page, with a URL of http://www.liberate.com/images/anim.gif, it is rendered as separate frames into the /prefix/www.liberate.com/images/anim.gif/directory. Each frame is called fmb.TIMESTAMP.seqence_number. The same scheme holds for all special assets of the presentation (animated GIFs, forms, frames).

Wannamaker at [0275].



| | |
|---|---|
| **[7d]** | create a temporal sequence of screen captures of a rendered web page, where each said screen capture defines content of the rendered web page at a given time, and at least two adjacent screen captures in the temporal sequence illustrate a dynamic change of at least a portion of the video content, image content or audio content over time, |

- Animated content is captured as separate frames – *i.e.*, a sequence of captures.
- The sequence of captures illustrates the dynamic change of the animation over time.



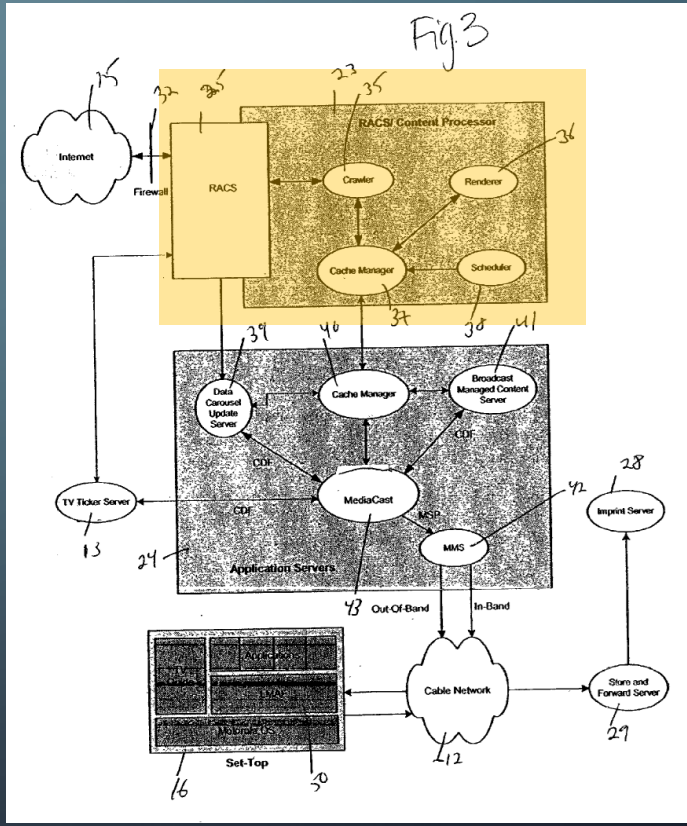

Wannamaker at Fig. 3.

| | |
|---|---|
| [7e] | assemble the temporal sequence of screen captures, and |

- The renderer assembles the sequence of screen captures into MPEG frames.

> **[0264]** After layout, the information platform renderer quickly and effectively compresses the output buffer to MPEG syntax in the FMB file format. All dynamic objects,

Wannamaker at [0264].



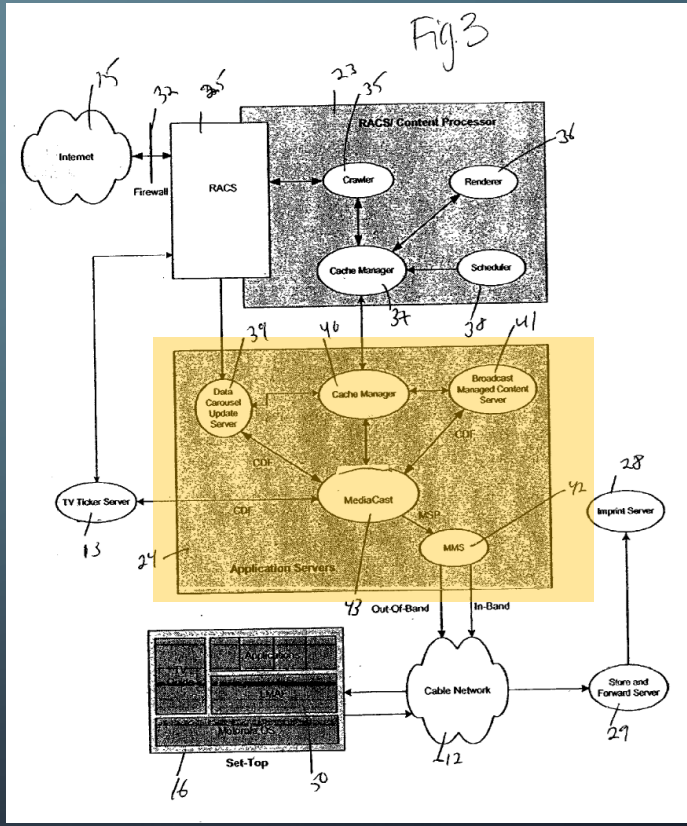

Wannamaker at Fig. 3.

| | |
|---|---|
| **[7f]** | provide at least one of the video content, image content and audio content to the multicast server for generating a multicast stream with MPEG encoded video and audio; and |

- The MPEG frames are provided to the Application Server, which generates MPEG streams for multicasting.

> **[0117]** The existing digital TV cable network infrastructure connects the set-top box to the application server. The downstream communication, from server to set-top box delivers an MPEG transport stream containing MPEG video, MPEG audio, and data via a specialized hardware multiplexer card. The information platform can be configured to

Wannamaker at [0117].




Wannamaker at Fig. 3.

| | |
|---|---|
| **[7f]** | provide at least one of the video content, image content and audio content to the multicast server for generating a multicast stream with MPEG encoded video and audio; and |

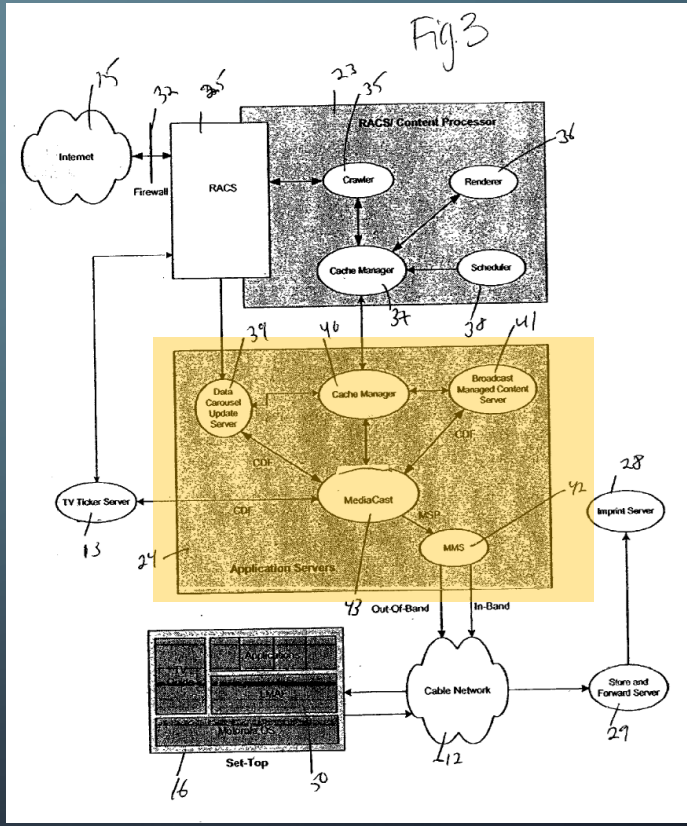
- The MPEG frames are provided to the Application Server, which generates MPEG streams for multicasting.

**[0193]** A multimedia streamer (MMS) consists of the audio/video/terminal server (AVTS) that delivers the units of audiovisual information sent by the system to the user's set-top box. It also contains the out-of-band servers (the

Wannamaker at [0193].

**[0195]** The AVTS is responsible for delivering MPEG video and Dolby AC-3 audio streams to the set-top box. A special MPEG multiplexer (MUX) card, controlled by the AVTS, streams data onto the cable network.

Wannamaker at [0195].




Wannamaker at Fig. 3.

| | |
|---|---|
| [7g] | a monitoring system configured to maintain generation of video and audio files, even in failure cases. |

- Wannamaker also discloses a monitoring system that provides fault tolerance.

problem, or restart the system through the OMI. The system is also capable of detecting fatal faults and restarting itself without human intervention. Much of the fault tolerance is configurable by the system operator.

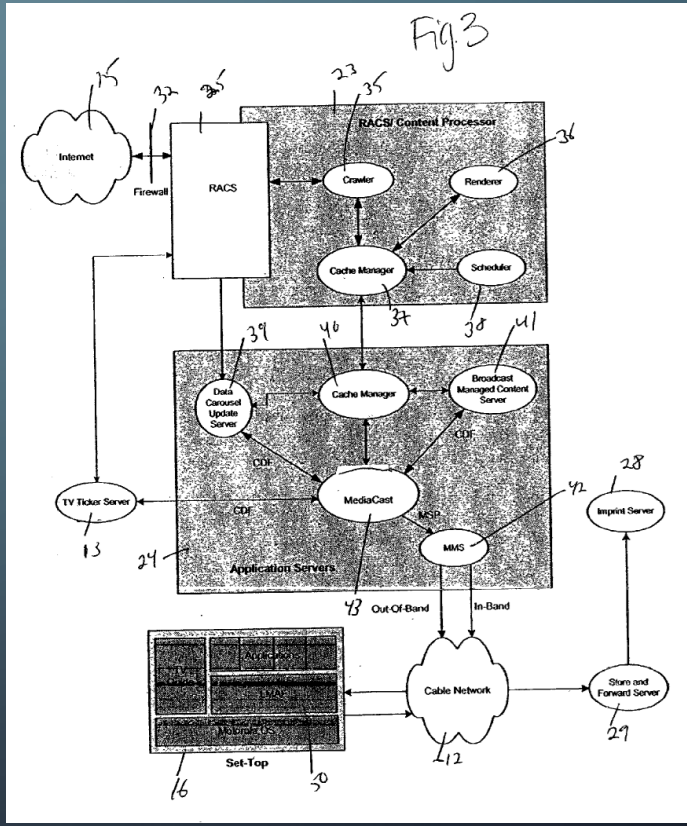
Wannamaker at [0320].

[0579] Fault tolerance—three levels of fault tolerance are provided, namely critical hardware, subsystem redundancy, and software fault tolerance.

Wannamaker at [0579].

server group. If a server fails, transactions are routed to another server in the server group. As many servers as necessary can be used to meet the load-balancing requirements of the service.

Wannamaker at [0101].




Wannamaker at Fig. 3.

# WANNAMAKER

U.S. Patent Pub. No. 2004/0031052

THE ’002 PATENT: CLAIMS 8-12

# WANNAMAKER AND THE ’002 PATENT

- In general, the Challenged Claims add well-known and common features.
  - The use of multiple servers for load balancing (claim 8)
  - The communication of a multicast or unicast stream (claims 9, 10)
  - The streaming of encoded video (claim 10)
  - The use of common standards, such as MPEG (claims 11, 12)
  - The addition of metadata (claims 11, 12)

# WANNAMAKER AND THE ’002 PATENT

- In general, the Challenged Claims add well-known and common features.
- Patent Owner does not contend he invented any of these features, which were all routinely implemented by POSITAs.
- Those features are either directly disclosed in Wannamaker and/or plainly disclosed in the cited secondary references.

# PATENT OWNER'S ATTEMPTS TO DISTINGUISH WANNAMAKER ARE UNAVAILING

# ARGUMENT: WANNAMAKER CAN ONLY RECEIVE A MULTICAST STREAM?

- Patent Owner argues that Wannamaker can only receive multicast and cannot generate multicast streams.

Motorala's "MPEG-2/Digital Out of Band Modulator." ***First,*** non of these constitutes of a "multicas server" ***Second,*** the OM1000 Motorola can only receive a multicast, and cannot "generate video content" in a multicast stream, only as part of modulation using QPSK (Quadrature Phase Shifting Keying) that work in the RF for the coaxial cable infrastructure. ***Thrird,*** the manual shows that a keypad command is needed, not a keyboard or a terminal, as a "server" requires.

IPR350: PO's Response at 42.

# ARGUMENT: WANNAMAKER CAN ONLY RECEIVE A MULTICAST STREAM?

- Wannamaker's Application Server explicitly includes Multimedia Streamer 42 (MMS) that can "multicast to a group of set-top boxes."

**[0193]** A multimedia streamer (MMS) consists of the audio/video/terminal server (AVTS) that delivers the units of audiovisual information sent by the system to the user's set-top box. It also contains the out-of-band servers (the

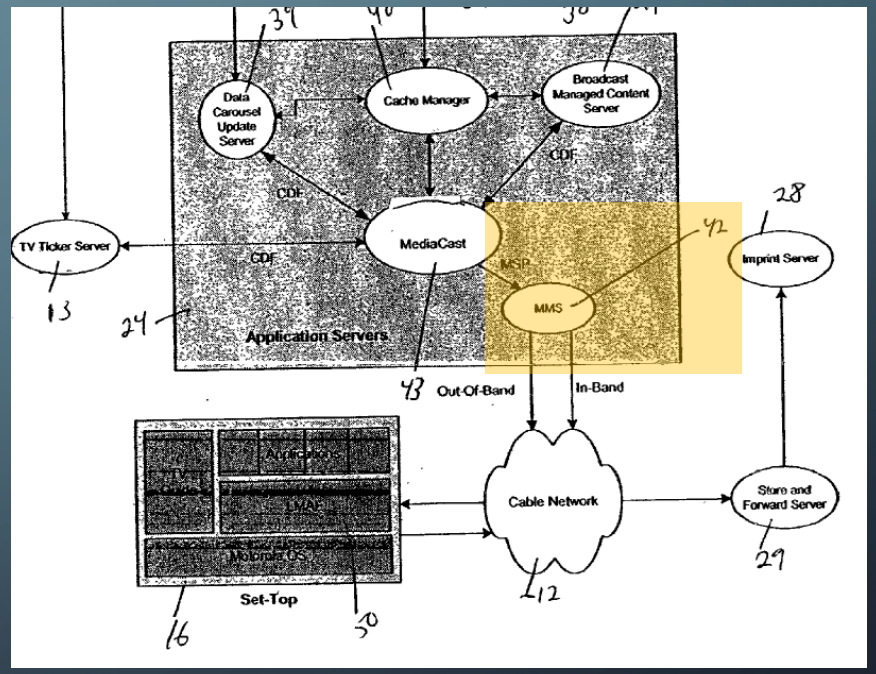
Wannamaker at [0193].

sent out-of-band. Generally, the MMS_OM1000 is used to send OOB directories and files, subjecting all data to rate-throttling to prevent buffer overflow in the set-top boxes.

**[0229]** The data may be:

- **[0230]** Broadcast to all set-top boxes
- **[0231]** Multicast to a group of set-top boxes
- **[0232]** Single-cast or "addressed" to a single set-top box

Wannamaker at [0231].




Wannamaker at Fig. 3.

# ARGUMENT: WANNAMAKER DOES NOT DISCLOSE ENCODING PARAMETERS?

- Patent Owner argues that Wannamaker does not disclose that the parameters sent from the Application Server to the Content Processor are encoding parameters.

> ***Thrid,*** In the context of encoding and processing video in the '002 Patent, as Claim 7 is, the parameters are, video and audio encoding parameters such as GoP (Group of Pictures), resolution, bit-rate in Kilobits per seconds, and encoders e.g. H.264 EX2039¶¶216-218

IPR350: PO's Response at 43.

# ARGUMENT: WANNAMAKER DOES NOT DISCLOSE ENCODING PARAMETERS?

- The claim language does not specify that the parameters are "encoding parameters" – just that they relate to video, image, and/or audio content.

7. A system, comprising:
a multicast server configured to generate parameters relating to at least one of video content, image content and audio content, and configured to output the parameters via MPEG outputs;

'002 Patent at Claim 7.

IPR350: Pet. at 50-53 and 57; Reply at 25-26.

# ARGUMENT: WANNAMAKER DOES NOT DISCLOSE ENCODING PARAMETERS?

- The ’002 specification includes example parameters that may be included in the set of parameters.

by the “Caching” unit. At step **505**, encoding flags or input parameters can be received which may include GOP (Group of Pictures) which is used by certain video encoders, resolution (e.g. 640×480) in pixels, bit rate in Kilobits per second, any encoding type of video (e.g. H.264), encoding type of audio (e.g. AC-3), duration of the video feed, etc. In

’002 Patent at 6:38-43.

pixel block boundaries where necessary. The FMB file is guaranteed to have the same width as the TV screen and the height is guaranteed to be at least the height of the screen.

Wannamaker at [0264].

IPR350: Pet. at 50-53 and 57; Reply at 25-26.

# ARGUMENT: WANNAMAKER DOES NOT CREATE SCREEN CAPTURES OF A RENDERED WEBPAGE?

- Patent Owner argues that "[n]o images are captured" in Wannamaker and that it instead operates using only markup language.

EX1006¶00259 shows that an HTML page is "Rendered" but into a CPML or MML page with FMB files. EX2039¶¶226-234 and EX1006 *Id.* [0660], CPML is an acryonim that means Information Platform Markup Language (CP possibly s the acronym used for Content Processor, hence Content Processor Markup Language), as well as MML, another markup language that is proprietary and referenced by EX1006 as well as "FMB" which is an image file format (Id. [0259]). Therefore, o images are captured as there is no need to create a "temporal sequene of screens" as simply the TV Navigator operares using CPML, MML, as Laptops use Safari and Chrome with HTML.

IPR350: PO's Response at 43-44.

# ARGUMENT: WANNAMAKER DOES NOT CREATE SCREEN CAPTURES OF A RENDERED WEBPAGE?

- Wannamaker expressly states that the web content is rendered as separate frames.

> given CPML file. Taking this example a step farther, if there is an animated GIF on the same page, with a URL of http://www.liberate.com/images/anim.gif, it is rendered as separate frames into the /prefix/www.liberate.com/images/anim.gif/directory. Each frame is called fmb.TIMESTAMP.seqence_number. The same scheme holds for all special assets of the presentation (animated GIFs, forms, frames).

Wannamaker at [0275].

IPR350: Pet. at 57-59; Reply at 26.

# ARGUMENT: WANNAMAKER DOES NOT CREATE SCREEN CAPTURES OF A RENDERED WEBPAGE?

- Wannamaker expressly states that the web content is rendered as separate frames.
- Each frame is captured as an image file – not just markup language.

> **[0259]** Most content requires rendering before it can be displayed on the set-top box. The renderer converts the content into a format compatible with the user's set-top box hardware. It does this by accepting URLs of HTML files that are stored locally, and then parsing and rendering the HTML to produce output image files (FMB) and output CPML or MML.

Wannamaker at [0259].

IPR350: Pet. at 57-59; Reply at 26.