ELUV.IO

The Content Fabric
Next Gen Content Distribution

LIVE
PVOD
FAST
ARCHIVES

Find out what the
Content Fabric
can do for you.

Learn More
@Eluv.io

Eluvio

Eluvio invented the Content Fabric, the most advanced secure content distribution network for owner-controlled storage, distribution, and monetization of digital content at scale. The hyper-efficiency of the Content Fabric protocol is unparalleled by expensive public cloud vendors and transcode platforms, eliminating the need for:

- Individual Cloud Transcoding Services
- Aggregation
- Content Management Systems
- Content Distribution Networks

Eluvio has powered content initiatives for Bedrock, EPCR, FOX, Globo, MGM/Amazon Studios, Microsoft, SONY Pictures, Telstra, UEFA, Warner Bros., WWE, and entertainers such as Dolly Parton, Black Eyed Peas, and many others.

Based in Berkeley, California, USA, Eluvio has received numerous industry awards including the prestigious Engineering Excellence Award by the Hollywood Professional Association (HPA), National Broadcasters Association (NAB) Product of the Year 2024, the SEICON Grand Prize, and is recognized with 14 US patents. Eluvio is led by Emmy Award-winning technologists, Michelle Munson and Serban Simu, founders and inventors of Aspera, a pioneer in digital video transport technology, and a core team of innovators.

The Eluvio Content Fabric

The Eluvio Content Fabric provides live and file-based content publishing, transcoding, packaging, sequencing, and dynamic and static distribution for all ranges of content experiences.

Video content is dynamically served directly from source objects as live and on-demand streaming, with dynamic sequences, and smart contract-based rights control secures the access and versioning of content across a global network of nodes. The nodes in the network all run the same open software protocol, which is both a blockchain and a just-in-time componentized content storage and distribution protocol.

The Content Fabric software stack executes a decentralized data protocol on nodes implemented in a global network, and replaces and/or consolidates the conventional functions of:

- Live Ingest

- Cloud Origin
- Live Transcoding
- Content Management
- Encryption/DRM
- Program Sequencing
- Rights and Avails Control
- CDN streaming
- Static Content Distribution

Nodes are peered directly with all of the major public clouds (AWS, Azure, GCP) which allows for local ingest from cloud storage buckets, and have high capacity (multi-terabit/second) IP transit capacity for direct ingest of live content and distribution of streaming and static content to audience (see "Content Fabric Distribution" diagram).
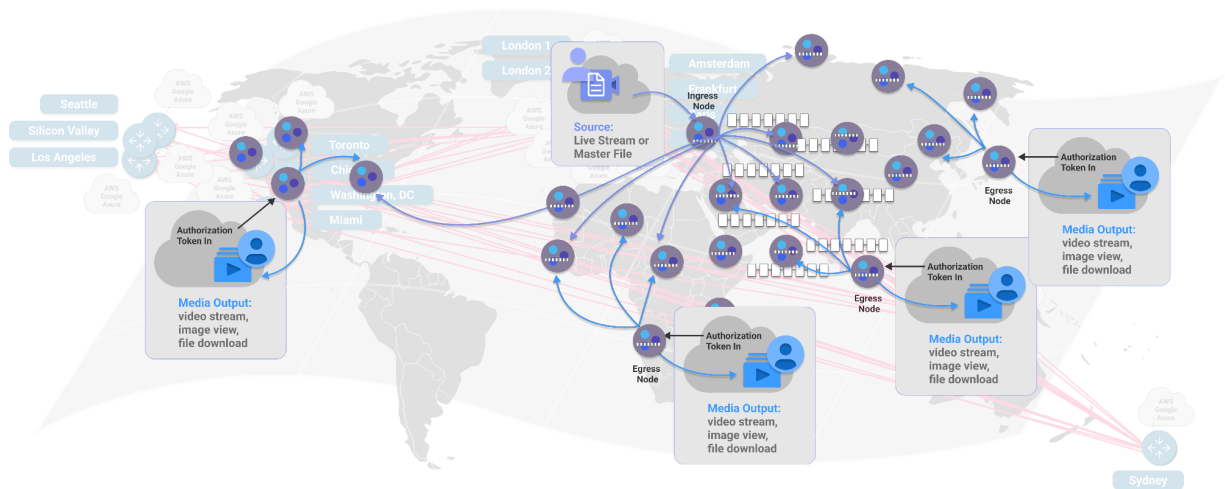

Diagram 1: Content Fabric Distribution

Content is efficiently represented, stored, routed, and composed as a core principle (see diagrams below). By virtue of being a content-centric service, there are no additional copies of content. This ensures efficient content composition and high security as there is no restful file state. Core bandwidth and storage are minimized because of the content awareness of the Fabric.

The Content Fabric creates and serves output such as adaptive bit rate streaming manifests and segments, and static content through a just-in-time process that executes within the nodes in the network. This process relies on an open decentralized data storage and distribution protocol and a componentized object representation of media essence, metadata and code called a "content object" (see "Anatomy of a Content Object" diagram). The content object is a data structure composed of references to the cryptographic hash signatures of the binary "parts" that comprise the object.
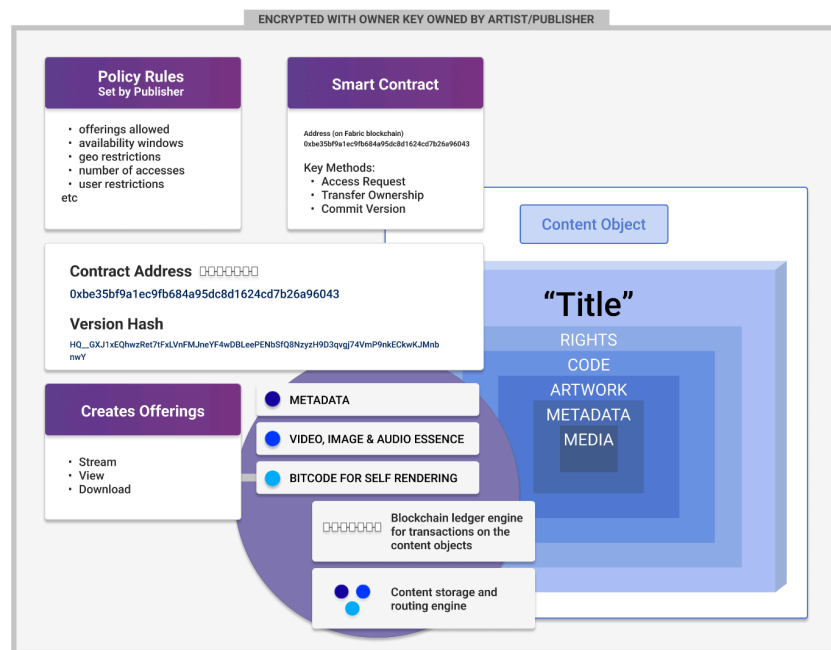
Diagram 2: Anatomy of a Content Object

Consumable media is built just-in-time, providing unrivaled efficiency and security personalization, with all associated metadata (see "Content Delivery Pipeline" diagram). Content transformation and transport are integrated within the live source lifecycle, which is pipelined and built for ultra low latency and performance of 2-3 seconds end-to-end from anywhere in the globe.
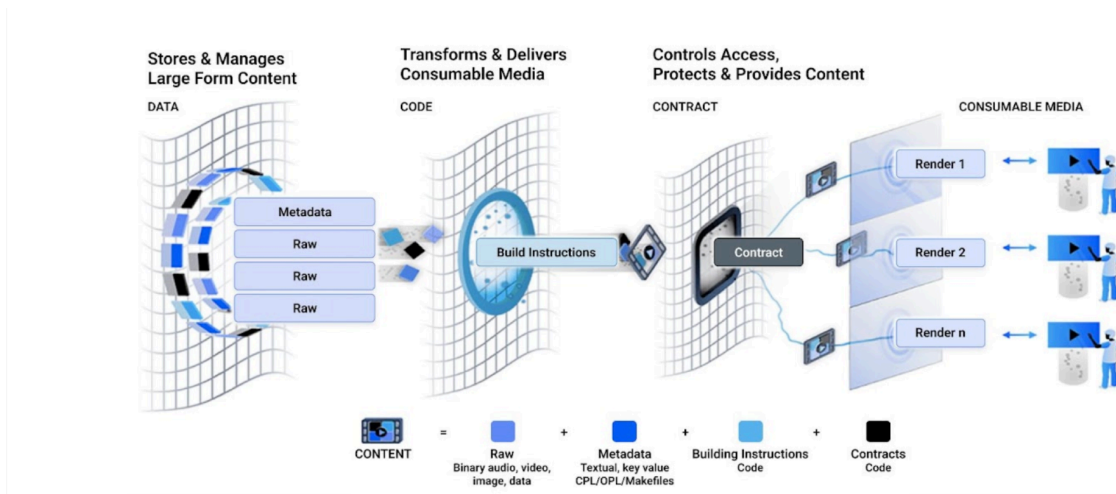

Diagram 3: Content Delivery Pipeline

Key Features for the Content Providers

Low Latency Live Streaming at Scale

- 2-3 second global delivery for HLS streams - 2-3 seconds end-to-end latency through playout for live, <500 milliseconds start-up latency for VoD/catch up
- Automatic ingest, recording, transcoding, packaging and distribution of live source streams (MPEG-TS, RTMP, SRT, etc.)
- Next Generation CDN providing hyper efficiency, speed, and extensibility
- Standard HLS streaming - *no custom player required*
    - Single bitrate HLS output for CDN origin & social streaming targets
    - Multi-variant adaptive bitrate (ABR) HTTP streaming direct to consumer
- Advanced features and applications
    - 4K, HDR, multi-channel audio, captions
    - Automatic and instant conversion of live-to-VoD and "catch up" viewing
    - Dynamic ad insertion and graphics
    - Multi-camera switchable stream views
    - Eluvio-provided embeddable player (all major browsers)
    - Eluvio-provided connected TV streaming app and SDK (Apple, Android, Amazon Fire, MS XBox, and others)

Next Gen CDN and Media Cloud

- Fabric is a next generation content native network eliminating the need for traditional CDNs
- Provides dramatically better speed, simplification and content re-use at scale
- Protocol provides inherently redundant storage and serving, reliability, linear scaling, and deterministic performance for global streaming and static serving
- No duplication of content
- Solution cost is dramatically lower than cloud + CDN
- No additional public cloud needed - full media stack built-in
- Dynamic, data driven content functions such as just-in-time transcoding, ad stitching and personalization are built-in and can be extended
- Content updates and all access is secure, authorized, and transparently logged

Scalability and Visibility

- Extensible to every event anywhere in the world
- Ingest event streams from any location directly to the Fabric
- Distribution capacity for greater than 500,000 concurrent streamers
- 99% of streaming traffic ingested, distributed, and output within 800 milliseconds
- 99.99% uptime for Fabric delivery services in year 2023
- Inherent redundancy with no partitions - any URL can be served by any node and all content parts are replicated 6-7 times
- No file size, stream duration or video/audio resolution restrictions
- End-to-End quality of experience monitoring, reporting and analytics per title, per stream, and in aggregate

Security

- Dynamic and personalized text and/or image watermarking per stream, e.g. with the recipient's identity
- Forensic watermarking of the recipient's stream (live and VoD, Nagra)
- Deterministic and end-to-end content authorization via in object policies allowing for geo blocking, time restrictions and other rights management
- Session authorization includes a revocable authorization token tied to the object's authorization policy. Blocking of a specific session/authorization token can be applied instantly through a policy update on the source object..
- End-to-end content encryption, including targeted re-encryption per receiver with strong DRM, optional forensic watermarking (which prevents leaks), and content part proofs that cryptographically verifies content delivery.
- The security model provides a complete solution at the content distribution source for controlling all content access and decryption, and revoking access on a per-session basis to prevent piracy, leaks or other unauthorized use.

Just-In-Time Content Delivery

- Flexible, high quality, and just-in-time content delivery - content is assembled and encoded for the required output at the time of delivery.
- Resource efficient - 50x less carbon footprint
- Tamper-proof content
- Inherent chain of ownership
- "On-chain" hashes of rights data and media
  - Policy is a "part" of the content and its hash is stored on-chain (ie. "does this user address had the requisite authorization for this offering?")
  - Policy can include any criteria such as date window, client geography, or client possessing a requisite on chain asset (token, ticket, etc.)
- Pluggable code pipelines, such as dynamic views, NFT traits, transformations, ect.
  - Cheap, fast factories for media NFTs and tokens
  - Single page apps that "read" from the Content Fabric with signing authorization through the "media wallet"
  - Live and interactive content governed by on-chain ownership

Simplicity - Single Technology Stack

The Content Fabric software stack executes a decentralized data protocol on nodes implemented in a global network, and replaces and/or consolidates the conventional functions of:

- Live Ingest
- Cloud Origin
- Live Transcoding
- Automatic Live to VoD
- Content Management
- Encryption/DRM
- Program Sequencing
- Rights and Avails Control
- CDN streaming
- Static Content Distribution

The Solution

For any live event, the live production stream(s) will be published directly from the contribution source to the Eluvio Content Fabric, or in the case of a pre-recorded event, the file recording will be published as a VoD object. The live stream source will be globally distributed as a live playable stream, and allow for automatic and instant conversion of live-to-VoD and catch up viewing. The VoD object can be replayed by the Fabric "as live" or streamed on demand. The Fabric's global footprint allows for convenient (network-local) ingest from virtually any location in the world. Live ingest through the Fabric adds virtually zero latency to the contribution feed; VoD ingest durations are typically 2-3X video time for professional formats.

As part of the service, the Eluvio Content Fabric will generate a globally streamable HLS/DASH output suitable for all required destinations – for direct consumption by end user viewers and for third parties (social, other CDNs, etc.) as well as RTMP output as needed (e.g. Facebook). The Fabric allows the same stream object to concurrently provide a single bit rate offering for third party social and other CDN/origin server targets and a multi-bitrate ABR output offering for direct consumption (e.g. in Content Providers.tv or other Content Providers digital properties). The playable URL can be directly integrated in any existing Content Providers application for global CDN playout by any standard HLS/DASH streaming player, or as an embeddable URL using the Eluvio embeddable player.

The end-to-end streaming path will be encrypted and playout offerings can include strong DRM and forensic watermarking. Additionally all of the playout offerings - whether dedicated to 3rd party or used for direct consumer distribution - will be protected with an authorization policy applied on the stream object. The policies can encode any business logic needed such as geographic restrictions, time restrictions, or user/group based access control, and are evaluated to validate a session access token on each request. Access tokens have a configurable expiration but better yet, a specific session can be revoked instantly with a policy update on the content object. The authorization policies and bitrates/variants for each playout offering one one content object can be distinct, allowing the same stream source and Fabric distribution to support all transmission destinations, saving cost and time.

Ingest to the Fabric can be over any TCP/IP path - 4G, Internet, bonding etc as mentioned. Redundant streams are supported allowing for automatic switchover to an alternative source without disrupting playout. Fabric generated playout manifests support automatic failover URLs. In the following sections we will detail how the solution will meet the

requirements of [Transmission, Streaming Production, Publication, Distribution to Third Parties, and Video Player].

Transmission

Latency

- Eluvio's Content Fabric supports ingest of live ingest, recording and publication of contribution feeds with just-in-time transcoding and streaming playout of standard HLS with end-to-end latencies of 2-3 seconds globally through playout on the end user device. The Fabric serves 2-second fragmented mp4 HLS segments with CMAF packaging (HLS/DASH) and generates both single and adaptive bit rate outputs with deterministic latencies.
- The Fabric on its own (exempting the real-time video processing) ingests, publishes and serves streaming playout in under 800 milliseconds for 99 percent of clients at scale for a deterministic 2 seconds +800 milliseconds of end-to-end latency for fMP4 video playout.
- This number is *independent of the number of streaming clients/receiving stakeholders*.
- Eluvio's Content Fabric will accept MPEG/TS over UDP, RTMP, SRT live input sources and any professional master video format (ProRes, XDCam, mp4, etc.) for file based content.
- Input stream and source file bit rates and resolutions are unlimited (including 4K / UHD, 8K, etc.) and can include HDR (for VoD). (HDR can be added for live if needed.) See https://eluv.io/StreamingAndReporting-2023-11.pdf for details. Our costing proposal assumes a proportion of the ingested content and streaming playout will be 4K/UHD.

Monitoring Solution

The Eluvio Content Fabric provides comprehensive real-time monitoring of the entire streaming pipeline from ingest of the contribution feed to playout by the end consuming client and a highly reliable and robust architecture. The uptime of the Content Fabric as a global full-stack media platform and CDN in year 2023 was 99.99% and the architecture's inherently redundant, agile, end-to-end and partition-free design allows it to provide robust guarantees on performance and availability.

The gathered statistics are reported via a continuous monitoring API and are displayed on multiple UX tools available per Content Fabric tenant. Operational staff will have direct

access to all monitoring tools to support operations of each event, and Content Providers staff may choose to have login access as well. We detail each part of the monitoring pipeline and relevant UX tools below.

- **Quality of Service Monitoring through the Content Fabric**
  - **Streaming QoS Data and Metrics**: The following statistics are gathered in real time and reported through the monitoring API for every live playable stream object
    - ingest - active status of the ingesting stream, active recording progress (stream time and bytes recorded and successfully published), ingest faults (source stream no longer sending)
    - transcoding and playout - segment serving time (average, max, and minimum, standard deviation, and 99 percentile metrics); active (successful) and failed playout sessions; filters by Fabric node and region; playout bitrate distribution
    - faults - number of failed playout requests and the error code/cause; filters by Fabric node and region
    - all metrics are logged directly within the Fabric nodes and aggregated periodically for persistent storage and query via REST/JSON APIs
  - **Visualization / UX Monitoring Tools**:
    - The **Live Stream Manager** displays the current status and health of all running streams and provides start/stop/reset control. It also provides a comprehensive live playout console for viewing and playing out any stream within a tenancy with filter options.
    - **Picture Video Quality of Monitoring & End User Quality of Service** - The per-stream drilldown in the Live Stream Manager allows for interactively viewing any stream as a client would, and continuously displays the time-to-first byte, segment arrival time, and player buffer statistics for the client. Advanced options allow for pinning the stream to a given Fabric node or to a specific playout profile for active troubleshooting during a live event.
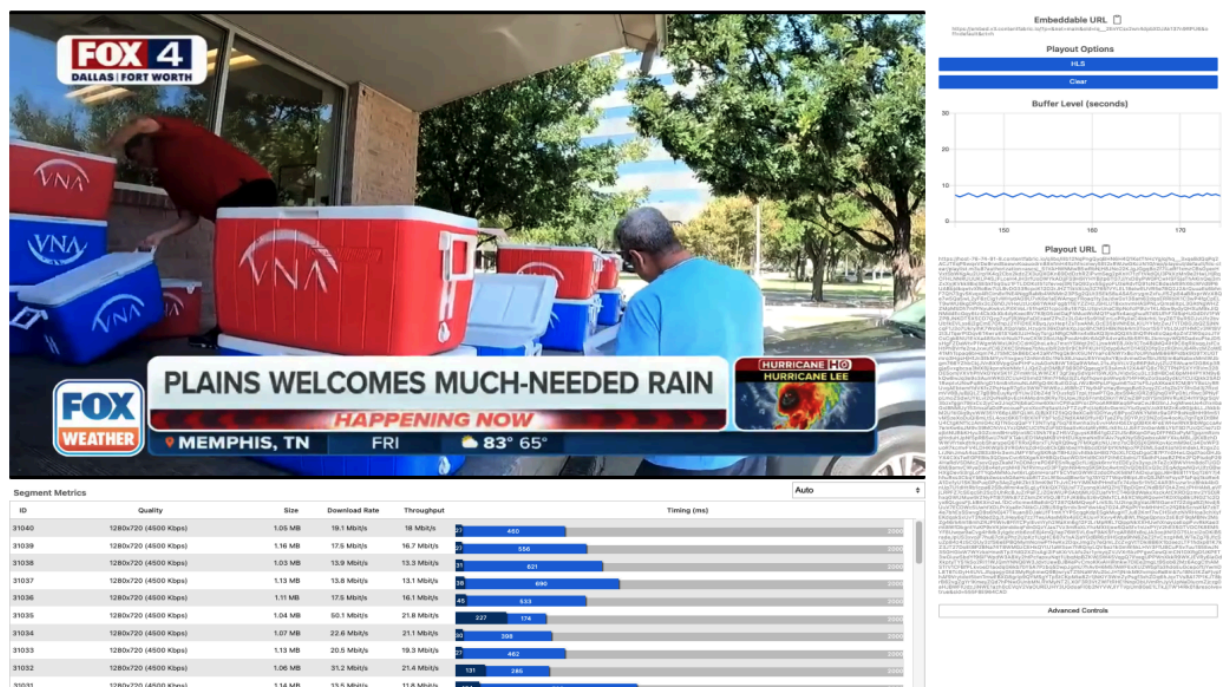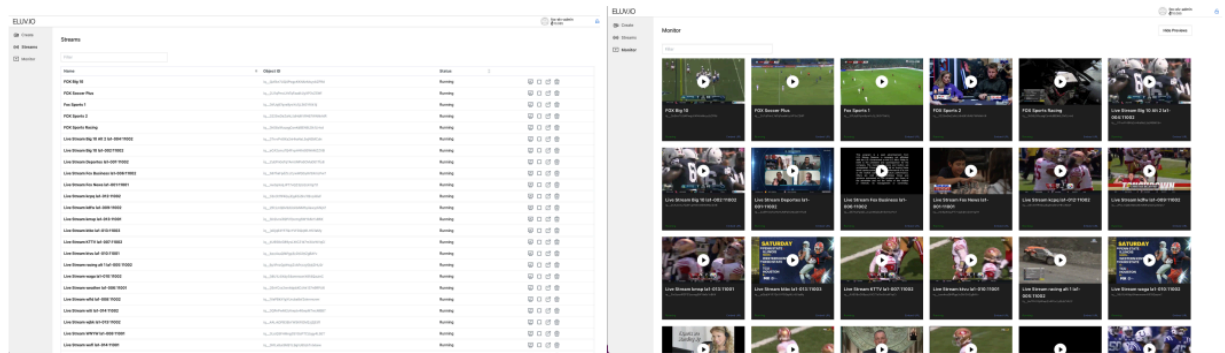
- The per tenant **Analytics** app provides continuous reporting and graphing for the following streaming data: (SEE SLIDE DECK FOR THE WALK THROUGH)
  - Real time (last 60 minutes), last 7, 24, and 30 days, by Title
    - Views, Unique Viewers, Total Playing Time, and Average Playing Time
    - Fabric Delivery Status: average, max, min, and 99 percentile segment delivery latency
    - Also filterable by content object ID, version hash, and fabric region

  - ○ Player stats as reported by MUX APIs including playback success, start up time, smoothness and video quality
- Please note - The Analytics app is a real-time UX for reporting data currently provided in batch, offline. Fabric tenancies currently provide downloadable reports of all viewer playout sessions by Title/Stream in the Reporting area. Data includes Number of accesses/views and duration of playout (totals and by bitrate) for all playable content objects (titles, streams, interactive experiences). For downloadable types (e.g. files, images), bytes served per object. Individual user sessions including unique authorization address or wallet address, content object URL accessed, and user agent details including device platform/browser/client IP and client IP to geo resolution. These core reports can be accessed as exportable CSV or via the tenant reporting

Transmission Endpoint and Demarcation Point

As described in the introduction the HLS streaming output created by the Content Fabric can be a source for Content Providers's present CDN, or the Fabric can be used as the CDN. Our cost proposal assumes the Fabric will be used as the streaming CDN for maximum benefit. Additionally playable offerings can also be provided for direct ingest by the social networks. The Fabric can provide single bit rate and adaptive bit rate HLS outputs with optional encryption and DRM suitable for the specified CDN and social platforms.

A live event source stream object can have multiple HLS offerings with different settings for encryption, watermarking and bit rate ladders and different authorization policies, allowing the Fabric to provide a highly cost and resource efficient transmission and flexible demarcation for all streaming transmission needs.

Streaming Production

Encoding and Enrichment of the Live Stream

The Content Fabric's ingest and publishing of a live contribution stream will automatically encode the stream for playout where the bit rate ladder and playout resolutions are configurable metadata on the stream itself. The Fabric pipeline supports all of the following for live and VoD ingest and streaming playout:

- Live, Delayed Live, VOD Playout
- Low Latency Live - 2 seconds "live edge" pullback globally
- Any resolution - 100 Mbps Visually Lossless, 8K, 4K with dynamic ABR
- HDR10
- Subtitles/captions
- Multiple audio tracks
- Multi-channel surround audio with AAC or Dolby AC3 & E-AC3 source encoding and AAC playout (Dolby Atmos in roadmap).
- Live Multiview Switch for personalized views
- Forensic watermarking using NAGRA A/B
- DRM (Apple Fairplay, Google Widevine), and Clear Key
- Content Encryption (CBCS and CENC)
- Discontinuities for inserted content / play across content boundaries, e.g. ad insertion
- Dynamic Content Selector for personalized insertion of content
- SCTE cues read, stored, and transmitted - can be added as needed to the output manifest
- Automatic Content Verification through Fabric Hashes

For **Graphics Enrichment** and Slates before, during and after the event, Eluvio has recently developed an in-Fabric HTML-5 graphics generator for dynamic graphics insertion in the Content Fabric audio/video pipeline.

Publication

3rd Party Social - Youtube and Facebook and CDN

As described in previous sections all Content Fabric streaming objects will automatically have a single bitrate HLS output URL for publishing to Youtube and major CDNs. The bitrate is configurable and different offering URLs can be configured on the same stream source object as needed to meet the requirements of these networks and Content Providers's preferred delivery bitrate. Additionally, we will provide an RTMP offering, e.g. for Facebook Live. Any dedicated HLS or RTMP direct distribution to a 3rd party can be automated using the Fabric's orchestration software.

Automatic (Instant) VoD Creation after the Live Event

The live stream objects will be configured for automatic creation of a VoD asset from the object using the Fabric's built in capability. The VoD creation is nearly instant - takes about 20 seconds for 2 hours of video - and requires no additional storage / file copies as the VoD object is created from the live recorded content object "parts". The resolution and bitrate of the VoD object will be the same as that of the live stream's top bit rate / recorded bit rate and downloadable versions can be provided at any bit rate.

Content Providers can use this feature for the downloadable link in H264 - 1080p @ 10 Mbps. Additionally, Content Providers could use the top bit rate as a downloadable version for storage in a MAM – OR - create a new feature in the MAM that simply refers to the top bit rate playable VoD stored in the Fabric, avoiding the file copy (and keeping the encyrption/security in place).

Content Fabric as CDN

We are proposing to use the Eluvio Content Fabric as the CDN for the live event streaming distribution, as it meets all of the requirements for scalability, redundancy, andreplay, and live-to-VoD feature set, and offers a new level of capability and efficiency. The Content Fabric as CDN can reduce the latency, improve the picture quality and user experience, simplify the streaming delivery, and reduce the cost of distribution, while opening new revenue streams for direct monetization of Content Providers content in the future.

Scalability: We have verified through live scale testing that the Fabric will serve the required concurrent viewers while maintaining the latency of 2-3 seconds of end-to-end latency through playout. The latency consists of 2 second segments + an average of 300 milliseconds of Fabric end-to-end latency for ingest, publishing and real-time transcoding and serving to players or 800 milliseconds for the 99th percentile of segments and clients.

We have built a live performance testbed "elv-perf" using the nodes in the Fabric as clients to generate scalable client streaming load against regions of the Fabric. The testbed activates the client population to concurrently playout live streaming URLs from the Fabric. In tests of 100,000 and 150,000 clients streaming from the Content Provider's live streams currently running in the Fabric, the segment delivery latency for 99% of segments and clients was less than 1 second, and the average delivery latency was under 300 milliseconds.

elv-perf-test	End-to-end segment latency (ingest/publish/transcode/serve)			
clients	99 percentile	Average	Maximum	Minimum
100,000	792 ms	250 ms	2077 ms	91 ms
150,000	972 ms	299 ms	4887 ms	90 ms

==This means streaming clients were served 2 second video segments where the "live streaming edge" advertised by the manifest was pulled back 2 seconds to match the real time video duration with an average end-to-end latency of 2.3 seconds and under 3 seconds for 99 percent of clients and segments.==

```
elvperf  test completed
started at: 2024-01-13T18:48:41.749Z
startup  : 30s
duration : 10m
spec     : /home/srv-prod-305/conf/elvperf-fr5-004.yaml

Segment Statistics

.---------.--------.--------.--------.--------.----------.----------.--------.---------.
|  STAT   | 2.5%   |  50%   | 97.5%  |  99%   |   AVG    |  STDEV   |  MIN   |  MAX    |
.---------.--------.--------.--------.--------.----------.----------.--------.---------.
| Latency | 96 ms  | 229 ms | 592 ms | 792 ms | 250.48 ms| 142.22 ms| 91 ms  | 2077 ms |
.---------.--------.--------.--------.--------.----------.----------.--------.---------.


.---------.--------.--------.--------.--------.----------.----------.--------.--------.
|  STAT   |  1%    | 2.5%   |  50%   | 97.5%  |   AVG    |  STDEV   |  MIN   |  MAX   |
.---------.--------.--------.--------.--------.----------.----------.--------.--------.
| Req/2s  | 4798   | 4863   | 5025   | 5216   | 5041.66  | 183.47   | 4744   | 6966   |
| Bytes/2s| 2.5GB  | 2.5GB  | 2.8GB  | 3.0GB  | 2.8GB    | 139.0MB  | 2.3GB  | 3.7GB  |
.---------.--------.--------.--------.--------.----------.----------.--------.--------.


.-------------------.---------.
|      TOTALS       |    #    |
.-------------------.---------.
| Test duration     |     10m |
| Clients           |  100.0k |
| Requests          |         |
|    2xx response   |    1.5M |
|    Other          |      98 |
|    Errors         |       0 |
|    Total          |    1.5M |
|    Delayed        |       0 |
| Bytes transferred | 33.58TB |
.-------------------.---------.


elvperf  test completed
started at: 2024-01-14T00:30:35.330Z
startup  : 30s
duration : 2m
spec     : /home/srv-prod-305/conf/elvperf-fr5-004.yaml

Segment Statistics

.---------.--------.--------.--------.--------.----------.----------.--------.---------.
|  STAT   | 2.5%   |  50%   | 97.5%  |  99%   |   AVG    |  STDEV   |  MIN   |  MAX    |
.---------.--------.--------.--------.--------.----------.----------.--------.---------.
| Latency | 94 ms  | 291 ms | 767 ms | 972 ms | 298.57 ms| 181.59 ms| 90 ms  | 4887 ms |
.---------.--------.--------.--------.--------.----------.----------.--------.---------.


.---------.--------.--------.--------.--------.----------.----------.--------.--------.
|  STAT   |  1%    | 2.5%   |  50%   | 97.5%  |   AVG    |  STDEV   |  MIN   |  MAX   |
.---------.--------.--------.--------.--------.----------.----------.--------.--------.
| Req/2s  | 5688   | 6988   | 7555   | 10726  | 7735.27  | 790.20   | 5688   | 11293  |
| Bytes/2s| 3.2GB  | 3.7GB  | 4.2GB  | 5.7GB  | 4.3GB    | 396.1MB  | 3.2GB  | 5.9GB  |
.---------.--------.--------.--------.--------.----------.----------.--------.--------.


.-------------------.---------.
|      TOTALS       |    #    |
```

```
 --------------------.---------.
| Test duration     |      2m |
| Clients           |  150.0k |
| Requests          |         |
|    2xx response   |  463.6k |
|    Other          |    445  |
|    Errors         |      0  |
|    Total          |  464.1k |
|    Delayed        |      0  |
| Bytes transferred | 50.37TB |
 --------------------.---------.
```

The performance held consistent independent of number of clients. Additionally, tests showed that capacity scales linearly with number of nodes.

In a test of concurrent streams per node using the same testbed, the Fabric nodes maintained this performance serving 4,500 concurrent clients each (<1 second latency for 99% of segments and clients served). Clients were 50-80 milliseconds from the Fabric serving host (meaning that clients need not be highly local for this benchmark.

The current Fabric footprint currently has more than 100 nodes available within relevant serving areas to guarantee low latency serving for more than 400,000 concurrent clients at peak.

Finally, because the Fabric's capacity scales highly efficiently and linearly with number of nodes, as Content Providers grows its use of the Fabric for even larger use cases, we add more nodes.

Redundancy in playout streaming

The Fabric's streaming playout guarantees redundant paths for a client by design as technically *any node* can serve the streaming client on demand (even mid event) and the client will be redirected to another node local to it on retry, and according to the base Content Fabric parts distribution protocol, all base "content parts" (from which segments are dynamically generated) are replicated seven times throughout the network – meaning any egress node has 7 redundant copies of a part to retrieve. If a node fails, it can be marked "private" (no longer serving clients or other node requests) with no impact to the live streaming clients, and when nodes are replaced they can download the parts they are responsible for, be marked public and start operation seamlessly for active clients.

Redundancy in source ingest

The manifest files generated by the Content Fabric have redundant playout URLs for any streaming client.

There is no limit to the number of redundant input streams that can be provided for the same live event. For all events the solution will provide for redundant source streams. If an input streams fails, the offering being served to clients can be updated in real time to refer to a new content version hash (for the failover stream). The next request made to the playout URL will resolve to the new content version hash.

DVR feature to allow instant replay and post-event live to VOD switch

All Content Fabric live streams are automatically recorded and support "instant" creation of a VoD object for the entire stream or a designated duration, without file copies. The originally recorded live stream "parts" are reused by reference in the VoD object for speed and extreme efficiency.   This feature also allows for automatic catch-up/instant replay from a given timestamp, and the revert to live.

End to End Monitoring of the Playout to Wide Range of Devices

The Eluvio Content Fabric monitoring and the metrics reported through the **Analytics** UI provide reporting of the Streaming Quality of Service metrics for *all* clients independent of device and player type. Additionally, if the Eluvio Embed Player for browser or connected TV is used for streaming playout, it records comprehensive Viewer Quality of Experience metrics as described on page 11 as reported by MUX APIs including playback success, start up time, smoothness and video quality. These metrics are recorded from all devices using the embed player.

Additionally, Eluvio has created native streaming apps for non browser based connected TV platforms (TVOS, Android TV, FireTV) which are also in the process of embedding the same MUX reporting statistics.

(Optional) Embeddable Video Player

Eluvio provides an embeddable player consisting of an open source player based on HLS.js and various capabilities offered by elv-client-js to interface with the Content Fabric for automatic authorization and streaming from embeddable URLs and embeddable iframe code. The player works on all major browsers and all web based connected TV platforms. Additionally, Eluvio has developed a native streaming client and wallet (also open source) for TVOS, Android TV, Kindle Fire and XBox that can be treated in a similar way as an embeddable streaming player and component for use in Content Providers applications.

Features of the embeddable player cover the requirements in the RFP:

- Start/pause Live, slider for video browsing, delay
- Back-to-Live edge, forward and rewind features
- The current player *does not* have built in catch up viewing features for live back to the start of the event but we can build this without difficulty using the Fabric's automatic live-to-VoD features.
- Can be readily integrated into Content Providers.com/Media Channel with an IFrame and within iOS and Android apps using "react native" technologies. The elv-player-js open source package provides for use of the native iOS player on Apple devices. The player is intentionally written with simple javascript for universality.
- Supports switch between portrait and landscape mode automatically
- Is fully resizable to any standard screen/image size (square, vertical) for all common mobile, tablet and laptop/desktop devices
- Is compatible with all major browsers on desktop and mobile - Chrome, Safari, Edge, Mozilla/Firefox
  - Specific OS Support
    - Chrome 39+ for Android
    - Chrome 39+ for Desktop
    - Firefox 41+ for Android
    - Firefox 42+ for Desktop
    - Edge for Windows 10+
    - Safari 9+ for macOS 10.11+
    - Safari for iPadOS 13+
    - Safari for iOS 17.1+
- Supports automatic management including pre-roll and auto-play for live events
- Supports VoD just as Live in the same player
- Multi-language dubs/audio selection within the payer for VoD and Live
- Each URL includes a secure authorization token that is exchanged for a Fabric access token or ticket code. Can be used to integrate any 3rd party sign on - OAuth/Custom/Wallet or secure individual ticket codes.
- The token authorization period/revocation period is fully configurable (default is 24 hours).
- All content URLs are secured and served via the Fabric end-to-end strong and trustless encryption model (see CDSA slides for overview)

Posting video to the Content Fabric from the Player

- The Eluvio Embed Player can be readily extended to allow for uploading content (or specifying "cuts" of played content) to the Content Fabric (VoD or even Live ingest, e.g. recorded on the user's mobile device) and automatically generate Fabric playout URLs for user generated content applications. We would like to discuss the requirements with Content Providers - this is an exciting opportunity and fits the environment well.

Analytics for the Player

- The **Analytics** UI app and API includes reporting over the past (hour, 24 hours, 7 days, 30 days) by streaming object/Title the Views, Unique Viewers, Total Playout Time and Average Playout Time.
- Viewership is recorded by timeseries for analysis over the duration of the event.
- Additionally the app includes the ability to display these viewership metrics by
  - User Geography (country, city)
  - User Device, OS, and Browser
  - Viewer Quality of Experience via MUX stats including playback success, start up time, smoothness and video quality

Automatic Geo Blocking and Authorization/Rights Management Policies

The Content Fabric's per offering authorization policies allow for fine grained geo fencing/blocking of client sessions and policies per streaming offering that enforce any business rules governing a user or group of user's access. The policies are part of the hashed, verified content object data and can be updated in real time, e.g. to block identified pirate sessions or to extend new access rights to existing content. This is a very flexible and powerful capability that can be used in many ways.
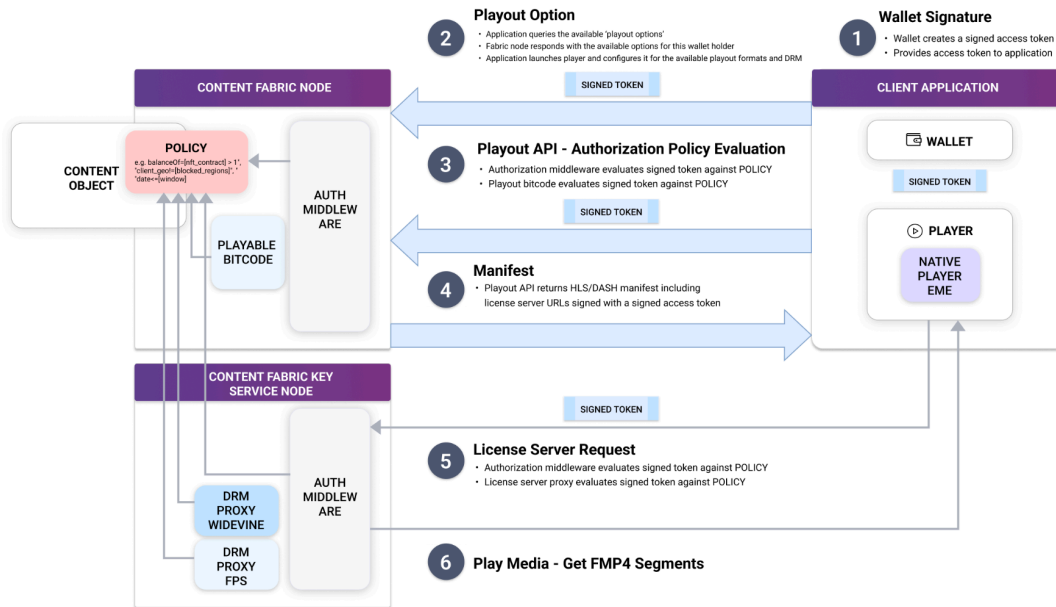
Diagram 6: Content Fabric Playout Authorization Flow

User Flow for Streaming Access

1. User or user group are booked access in the Content Providers booking system.
2. Account/entity is authorized for the respective stream object offering (including any policy restrictions on access)
3. The stream begins or is made available at the designated time
4. End user interfaces display the upcoming or live event offerings authorized to the user
5. Stream access ends at designated time (Fabric policy enforces time windows, geo windows, offerings)
6. NB Unauthorized access to the stream are not possible by virtue of the Fabric's unique encryption and authorisation
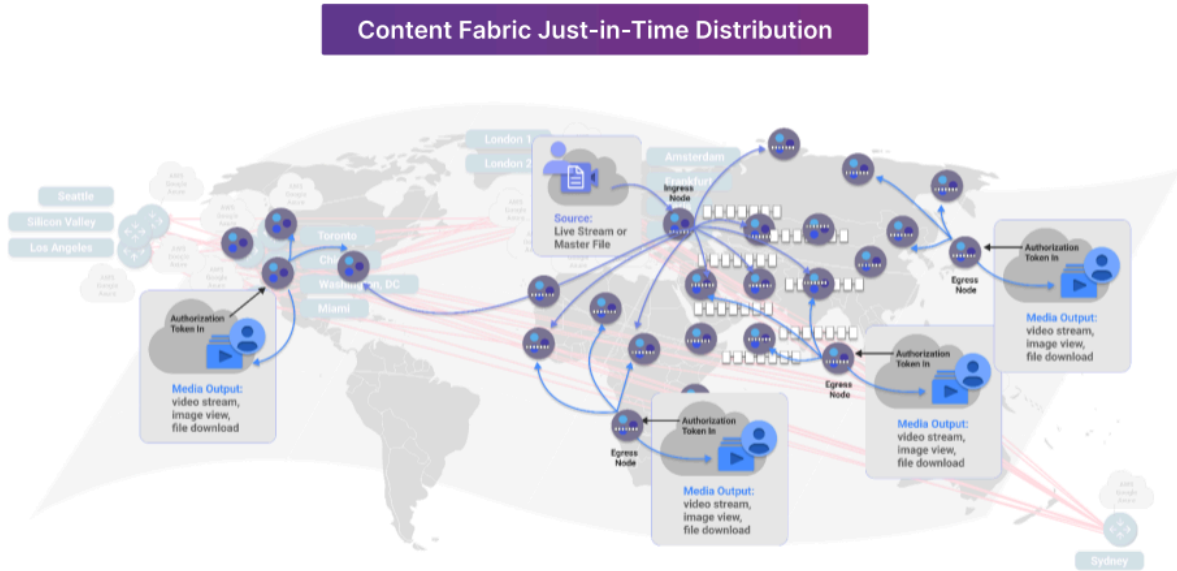
Appendix

Diagram 7: Content Fabric Just-in-Time Distribution

The low latency performance and consistency is made possible by the Content Fabric's novel Just-in-Time transcoding and streaming pipeline as described on pages 8-9 in https://eluv.io/EluvioContentFabricProtocolWhitepaper.pdf

Figure 5: Just-in-Time Transcoding and Streaming Pipeline