

Uma Plataforma Aberta e Instrumentada para *Streaming* de Vídeo: Medição de *Trade-offs* entre Desempenho, Custo e Qualidade

Michel Almeida da Silva¹, Leonardo Bidese de Pinho¹

¹Universidade Federal do Pampa (UNIPAMPA) – Campus Bagé – RS – Brasil

michelsilva.aluno@unipampa.edu.br, leonardopinho@unipampa.edu.br

Abstract. *Video consumption dominates global Internet traffic, challenging streaming efficiency amid a scarcity of open analysis tools. This work developed and evaluated a parameterizable streaming architecture for both research and market use, investigating performance, cost, and quality of service and experience (QoS/QoE). A reduced-scale measurement and projection strategy was adopted, projecting large-scale behavior from real implementation costs. Two experiments evaluated distribution capacity and QoE, and codec efficiency via rate-distortion (BD-rate). Results confirm distinct trade-offs in each pipeline stage: i) in distribution, egress dominates cost while control-plane concurrency limits scale; and ii) in encoding, efficiency is balanced against speed.*

Resumo. *O consumo de vídeo domina o tráfego global da Internet, desafiando a eficiência de streaming. Diante da carência de ferramentas abertas, este trabalho desenvolveu e avaliou uma arquitetura de streaming parametrizável, voltada à pesquisa e ao uso em mercado, investigando desempenho, custo, qualidade de serviço (QoS) e qualidade de experiência (QoE). Adotou-se medição em escala reduzida e projeção. Dois experimentos avaliaram capacidade e QoE da distribuição, e eficiência de codecs por rate-distortion (BD-rate). Os resultados confirmam trade-offs distintos em cada estágio do pipeline: i) na distribuição, o tráfego de saída domina o custo, enquanto a concorrência do plano de controle limita a escala; e ii) na codificação, o equilíbrio entre eficiência e velocidade.*

1. Introdução

O consumo de vídeo constitui, atualmente, a parcela majoritária do tráfego global de Internet, com projeções que indicam um crescimento contínuo e acentuado para os próximos anos [ERICSSON 2024]. Esse crescimento pressiona a eficiência das arquiteturas de *streaming*, especialmente na relação entre qualidade de transmissão, consumo de recursos e latência. Conforme destacado em [Laghari et al. 2023], a área enfrenta obstáculos complexos relacionados à largura de banda, custos de infraestrutura e heterogeneidade de dispositivos. Embora se observem avanços notáveis em *codecs* e protocolos, persiste uma lacuna técnica na integração desses componentes: falta uma plataforma aberta, instrumentada e parametrizável, que cubra todo o *pipeline* de vídeo sob condições controladas e mensuráveis. A alternativa usual é contratar integralmente um serviço gerenciado de terceiros: soluções proprietárias do tipo “caixa-preta”, cujos mecanismos internos não são observáveis nem ajustáveis pelo responsável por instanciar e operar o serviço.

Este trabalho ocupa essa lacuna com uma plataforma de código aberto, concebida como instrumento de pesquisa reproduzível e, por ser aberta e modular, também como base viável para operação real em mercado, alternativa às soluções comerciais de caixa-preta. A contribuição pretendida reúne uma metodologia de medição em escala reduzida e projeção, detalhada na Seção 4.1, e conhecimento explicativo sobre o comportamento do *pipeline* de *streaming* sob carga.

O problema central desta investigação reside na complexidade de orquestração de uma arquitetura de *streaming* de Vídeo sob Demanda (VoD) que atenda a requisitos de disponibilidade e escalabilidade em um ambiente controlado. Especificamente, questiona-se como medir, em escala reduzida e instrumentada, os *trade-offs* entre desempenho, custo e qualidade nas duas etapas críticas do *pipeline* de VoD: a distribuição dos segmentos e a codificação do vídeo, de modo a projetar o comportamento da arquitetura em escala de produção. Essa questão é decomposta em duas perguntas operacionais, cada uma investigada por um estudo específico: a primeira, na distribuição, é qual a capacidade de entrega em espectadores concorrentes que a arquitetura sustenta, com que QoE medida do lado do *player*, como *proxy*, e quanto custa operá-la em escala de nuvem; a segunda, na codificação, é qual *codec* preserva a mesma qualidade perceptual com menor taxa de bits, e a que custo de velocidade de codificação.

2. Trabalhos Relacionados

Os trabalhos comparados neste artigo foram identificados na fase exploratória desta pesquisa, por uma Revisão Sistemática da Literatura (RSL) conduzida segundo o protocolo de [Neiva and da Silva 2016]. Três questões de pesquisa orientaram buscas em cinco bases acadêmicas, Google Scholar, IEEE Xplore, ACM Digital Library, SpringerLink e ScienceDirect, refinadas em seis iterações até reunir 150 referências, filtradas por triagem de título, resumo, introdução e conclusão, e avaliadas por um checklist de qualidade em escala Likert até a síntese final de 19 artigos de alta relevância.

Desse corpus, quatro trabalhos tratam aspectos isolados do *pipeline*: protocolos e *bitrate* [Bentaleb et al. 2025, Timmerer et al. 2025], codificação em nuvem [Moina-Rivera et al. 2024], e distribuição comercial [Suman et al. 2022]. Nenhum integra desempenho, custo e qualidade em uma única plataforma aberta e reproduzível. É essa lacuna que este trabalho aborda.

3. Arquitetura Proposta

A arquitetura proposta é composta por cinco módulos funcionais interconectados, materializados em microsserviços políglotas (Figura 1). O fluxo de processamento inicia no módulo de Ingestão, que recebe o conteúdo bruto de forma orientada a eventos. Em seguida, o módulo de Transcodificação processa o vídeo em múltiplos *codecs*, H.264, H.265 e AV1, gerando a *bitrate ladder*. O módulo de Empacotamento fragmenta essas representações no formato *Common Media Application Format* (CMAF), único e que serve simultaneamente aos protocolos *Dynamic Adaptive Streaming over HTTP* (DASH) e *HTTP Live Streaming* (HLS). A entrega é realizada pelo módulo de Distribuição, que atua como servidor de catálogo e manifesto, com os segmentos de vídeo entregues por uma *Content Delivery Network* (CDN). Por fim, o módulo de Reprodução executa a adaptação de *bitrate* no cliente e coleta as métricas de QoE.

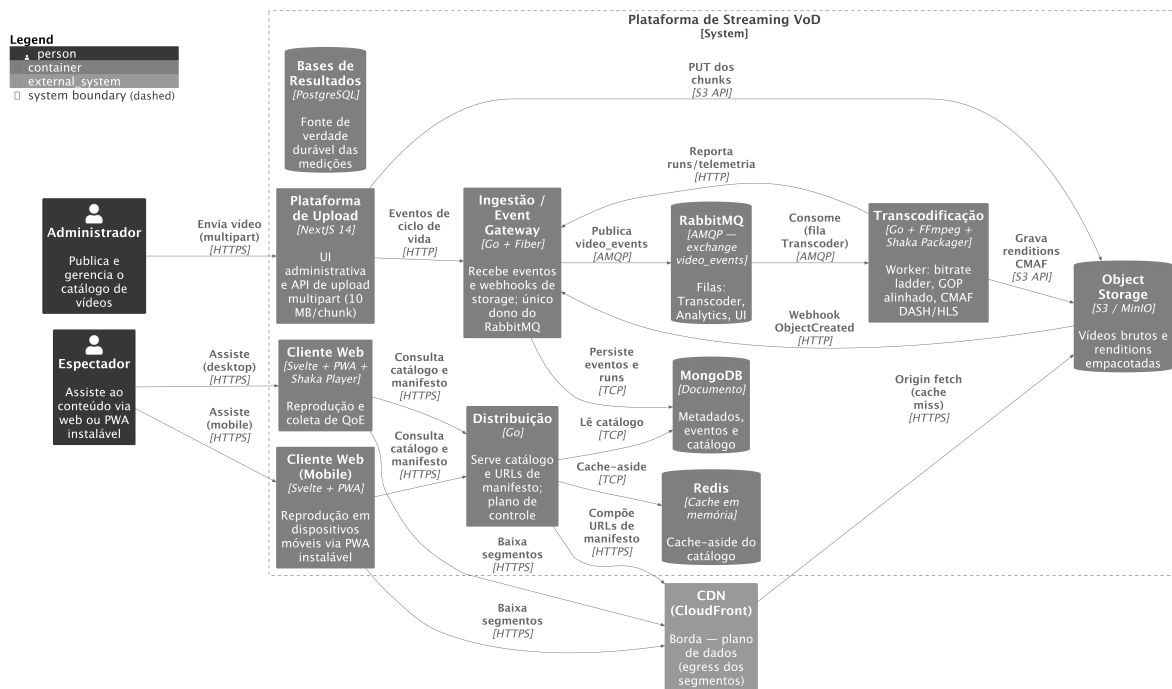


Figura 1. Diagrama de contêineres (C4) da plataforma de streaming VoD.

Três decisões são transversais à arquitetura. A primeira é a separação entre plano de controle e plano de dados: o catálogo e o manifesto, de baixo volume mas sensíveis à latência de leitura, são servidos pelos microsserviços com apoio de *cache* em Redis; já os segmentos de vídeo, volumosos e responsáveis pela maior parte do *egress*, são delegados a uma CDN comercial. Essa separação torna o plano de dados um módulo substituível, sem alterar a lógica dos demais serviços. A segunda é a aplicação do padrão *adapter* por meio de interfaces de abstração, que desacoplam a lógica de domínio dos provedores concretos de armazenamento e mensageria, permitindo substituí-los sem alterar o código de negócio. A terceira é a alocação heterogênea do modelo de computação, alinhada ao perfil de cada carga: o plano de requisição, ingestão e distribuição, apresenta operações curtas e tráfego elástico, perfil atendido pelo modelo *Function-as-a-Service* (FaaS) da AWS Lambda; já o plano de processamento, a transcodificação, executa *jobs* longos e acelerados por GPU, perfil que demanda o modelo de instâncias do AWS Batch sobre EC2, evitando tanto o custo ocioso do *serverless* mal aplicado quanto o limite de tempo que ele impõe a *jobs* longos.

4. Análise Experimental

4.1. Metodologia

Adotou-se a estratégia de medição em escala reduzida e projeção: mediu-se o custo unitário de operação, o *footprint* por espectador, por requisição e por codificação, sobre a implementação real e instrumentada, e essas constantes empíricas alimentaram a projeção do comportamento agregado em patamares de produção, revelando o custo financeiro, a capacidade necessária e o primeiro ponto de saturação de cada cenário.

O cenário experimental é uma plataforma de VoD no contexto de uma rede de farmácias, cujas características de carga parametrizam o modelo de custo e capacidade:

cerca de 150 mil acessos por dia, concorrência de pico de ~ 7.500 espectadores em uma janela de ~ 1 h/dia, vazão de entrega de pico de ~ 1.875 req/s, e latência de *streaming* alvo abaixo de 400 ms.

O desenho experimental adota quatro controles de validade para garantir que cada medição reflita o sistema sob teste, e não fatores externos: a definição precisa do serviço e da rota sob medição; um ambiente de rede controlado entre *containers*; replicação com reporte de mediana; e geração de carga por *players* reais, simulando um consumo sustentado e realista. As métricas coletadas cobrem QoS no servidor, como vazão, latência por percentis, *egress* e custo, e QoE no cliente, como *startup time*, *rebuffering* e *bitrate* médio, além de eficiência e velocidade de codificação: *Video Multimethod Assessment Fusion* (VMAF), métrica de qualidade perceptual, e *Bjøntegaard Delta Rate* (BD-rate), a diferença percentual de taxa de bits entre *codecs* na mesma qualidade.

4.2. Resultados e Discussões

O primeiro estudo mediu a capacidade de entrega do módulo de Distribuição em nuvem, usando o gerador leve *vclient* para escala e o Shaka Player instrumentado como âncora de fidelidade de QoE. O *vclient* sustentou 10.000 sessões com 100% de entrega, sem *rebuffer* e *footprint* constante ($\sim 4,9$ MB); o Shaka, em menor escala, validou o padrão de requisições e ancorou a QoE. Assim, o resultado em 10 mil estabelece a capacidade de distribuição do sistema, enquanto a QoE se limita ao piso de infraestrutura ancorado pelo Shaka, sem representar a experiência subjetiva dos usuários em condições reais de uso. A Tabela 1 relaciona, para cada patamar de espectadores concorrentes, o egress diário projetado, o custo mensal no teto de transmissão contínua, e o componente que primeiro satura.

Tabela 1. Capacidade, custo-teto e saturação da distribuição (medido até 10 mil espectadores, projetado até 25 mil)

Espectadores	Egress/dia	Custo/mês (US\$ milhões)	Satura
10.000	486 TB	$\sim 1,27$	—
15.000	729 TB	$\sim 1,90$	—
20.000	972 TB	$\sim 2,54$	—
25.000	1.215 TB	$\sim 3,18$	Lambda

Na coluna de espectadores concorrentes da Tabela 1, apenas o valor de 10 mil foi medido; 15, 20 e 25 mil são extrapolações lineares. As colunas de egress e custo, em todas as linhas, correspondem a um limite superior teórico: o custo de operação sob transmissão contínua, 24 horas por dia, durante um mês, não o valor efetivamente incorrido em uma execução real. O teste durou poucos minutos e produziu um volume de dados da ordem de dezenas de gigabytes, quatro ordens de grandeza abaixo das centenas de terabytes projetadas. Essa transposição, de um intervalo de medição reduzido para uma projeção de escala de produção, constitui a estratégia adotada neste trabalho. A coluna Satura indica o componente que primeiro atinge seu limite de capacidade em cada patamar: a função *serverless* do plano de controle, cuja cota de concorrência limita a arquitetura no patamar projetado de 25 mil espectadores.

A análise de custo mostra que o *egress* da CDN domina a fatura, da ordem de 85% a 95% do total [Amazon Web Services 2024], o que aponta como principais alavancas

de redução a adoção de *codecs* mais eficientes, a maximização da taxa de *cache-hit* e políticas de ciclo de vida no armazenamento. A saturação da função *serverless* é explicada pela Lei de Little [Little 1961]: o número de execuções concorrentes L de um recurso é o produto entre a taxa de chegada de requisições λ e o tempo de serviço $\bar{W}$. No plano de controle, λ cresce linearmente com o número de espectadores, $\lambda = \bar{r} \cdot N$, em que $\bar{r}$ é a taxa de consultas ao plano de controle por espectador. No ponto de saturação, a concorrência consumida iguala a cota disponível C , o que resulta em

$$N_{\max} = \frac{C}{\bar{r} \cdot \bar{W}}. \quad (1)$$

A Equação (1) expõe três alavancas independentes para escalar acima desse ponto: aumentar a cota de concorrência C junto ao provedor, com efeito linear, adotar concorrência provisionada para reduzir $\bar{W}$ nos picos de início de sessão, ou servir o manifesto pela própria CDN, reduzindo $\bar{r}$ a quase zero e deslocando o teto para o *egress*, que escala horizontalmente.

O segundo estudo mediu o BD-rate [Bjøntegaard 2001] de três *codecs*, H.264, H.265 e AV1, tanto em *software* quanto em aceleração por GPU via NVENC, relativo à âncora `libx264`. A Tabela 2 cruza a eficiência de compressão com o ganho de velocidade de codificação em 1080p.

Tabela 2. BD-rate e *speedup* de codificação em 1080p

<i>Encoder</i>	<i>Backend</i>	BD-rate	<i>Speedup</i>
h264 (libx264)	<i>software</i>	0% (âncora)	1×
h265 (libx265)	<i>software</i>	−33,3%	~1×
av1 (libsvtav1)	<i>software</i>	−55,4%	~1×
h265_nvenc	GPU	−30,6%	~8–12×
av1_nvenc	GPU	−49,6%	~6×

Os resultados confirmam a ordem de eficiência esperada da literatura: em *software*, o AV1 supera o H.265, que supera o H.264. O achado central é que a penalidade do NVENC em relação ao *software* é pequena: o `h265_nvenc` entrega de 8 a 12 vezes mais velocidade de codificação que o `libx265` pagando apenas ~3 pontos de BD-rate, e o `av1_nvenc` cerca de 6 vezes a velocidade por ~6 pontos. Para um *pipeline* com recodificação frequente ou requisitos de baixa latência, o NVENC é vantajoso mesmo pagando o preço mais alto por hora de uma instância com GPU, pois o ganho de velocidade mais que compensa a diferença de custo por codificação. Registra-se, como ameaça à validade, que os *encoders* de *software* rodaram no *preset veryfast*, calibrado para velocidade e não para qualidade, o que subestima sua eficiência e torna a magnitude da penalidade do NVENC um limite superior.

5. Considerações Finais

Este trabalho propôs e avaliou experimentalmente uma plataforma aberta de streaming de vídeo como instrumento de medição dos trade-offs entre desempenho, custo e qualidade. As contribuições se organizam em três planos: metodológico, com o método de medição

em escala reduzida e projeção; técnico, com uma arquitetura de referência aberta, modular e reproduzível, com adaptadores que permitem substituir provedores concretos; e científico, com evidência de que o egress domina o custo na distribuição e o NVENC troca pouca eficiência por muita velocidade na codificação. A plataforma aberta, modular e validada sob carga atende à pesquisa reproduzível e à operação de mercado, como alternativa às ofertas comerciais de caixa-preta. O escopo limita-se a VoD, com segmentos entregues por CDN comercial, sem avaliar uma CDN *self-hosted*; as métricas de QoE representam o piso de infraestrutura, pois a experiência em campo depende da última milha e do dispositivo.

Referências

- Amazon Web Services (2024). Video on demand on AWS: Cost. Acesso em: 14 jun. 2026.
- Bentaleb, A., Lim, M., Akcay, M. N., Begen, A. C., Hammoudi, S., and Zimmermann, R. (2025). Toward one-second latency: Evolution of live media streaming. *IEEE Communications Surveys & Tutorials*. Early access.
- Bjøntegaard, G. (2001). Calculation of average PSNR differences between RD-curves. Technical Report VCEG-M33, ITU-T SG16 Q.6 Video Coding Experts Group (VCEG), Austin, Texas, USA.
- ERICSSON (2024). Exploring mobile traffic trends 2020-2024. In: *Ericsson Mobility Report*. Disponível em: <https://www.ericsson.com/en/reports-and-papers/mobility-report/articles/exploring-mobile-traffic-trends-2020-2024>. Acesso em: 27 ago. 2025.
- Laghari, A. A., Shahid, S., Yadav, R., Karim, S., Khan, A., Li, H., and Shoulin, Y. (2023). The state of art and review on video streaming. *Journal of High Speed Networks*, 29(3):211–236.
- Little, J. D. C. (1961). A proof for the queuing formula: $L = \lambda W$. *Operations Research*, 9(3):383–387.
- Moina-Rivera, W., Garcia-Pineda, M., Gutiérrez-Aguado, J., and Alcaraz-Calero, J. M. (2024). Cloud media video encoding: review and challenges. *Multimedia Tools and Applications*, 83(34):81231–81278.
- Neiva, F. W. and da Silva, R. L. d. S. (2016). Revisão sistemática da literatura em ciência da computação: Um guia prático. Relatório técnico, Universidade Federal de Juiz de Fora, Juiz de Fora.
- Suman, P., Moon, Y.-S., and Choi, M.-J. (2022). Netflix, amazon prime, and youtube: comparative study of streaming infrastructure and strategy. *Journal of Information Processing Systems*, 18(6):729–740.
- Timmerer, C., Amirpour, H., Tashtarian, F., Afzal, S., Rizk, A., Zink, M., and Hellwagner, H. (2025). Http adaptive streaming: A review on current advances and future challenges. *ACM Trans. Multimedia Comput. Commun. Appl.*, 21(7).