

Video.js v10 evaluation checklist

Bundle math, decision tree, migration questions, yes/no recommendation grid.

Bundle math (gzip kB, 10 Mar 2026 snapshot)

Video.js v8 (core, no ABR): 75.2 kB.

Video.js v10 [HTML, no ABR]: 25.1 kB — 88% smaller than v8.

Video.js v10 [React, no ABR]: 18.0 kB.

Video.js v10 Background [React]: 3.5 kB — landing-page hero.

Video.js v8 + VHS (with ABR): 202.7 kB.

Video.js v10 + SPF [HTML]: 38.7 kB — 81% smaller with ABR.

Video.js v10 + hls.js (advanced): 164.1 kB — 20% smaller than v8+VHS.

SPF-composed engine only: 12.1 kB vs hls.js 155.9 kB.

Five questions in your planning meeting

1. Is this a greenfield project or a migration of an existing v8 deployment?
2. Does the workload need DRM, server-side ads, or low-latency live? (If yes → not SPF beta.)
3. Is the team React-first or framework-agnostic? (React → v10's strongest fit.)
4. Does the team have bandwidth to rewrite the player skin and re-test the integration?
5. Can you ship a beta API or do you need a stable contract today?

Yes/no recommendation grid

Marketing landing page, single video, no ABR: Ship v10 default HTML (25 kB). Yes today.

Short-form CMAF social feed, no DRM: Ship v10 + SPF (38 kB total). Yes today.

Course platform, HLS VOD, captions, no DRM: Ship v10 + hls.js (164 kB) today; revisit SPF post-GA.

Subscription OTT with multi-DRM, ads, live: Ship v10 + Shaka or hls.js today; SPF post-GA roadmap.

Existing large v8 deployment in production: Stay through GA. Plan migration post-GA + 6 months.

Smart-TV app (Tizen, webOS, Roku): Wait for GA + platform validation. v8 today.

Migration-readiness questions (existing v8 → v10)

Do you have a custom skin? (Eject in v10; rewrite primitives in React/HTML.)

Do you use third-party plugins? (Most will not work; audit the list.)

Do you use prototype overrides on Player? (Pattern removed; reimplement via state slices.)

Do you use cast / fullscreen / PiP? (APIs renamed in v10; update calls.)

Is your test corpus automated? (Run against v10 beta before scheduling cutover.)

Do you ship to TVs? (Wait for vendor validation before TV migration.)

Red flags — DO NOT ship v10 + SPF beta if...

... you need DRM today (Widevine / PlayReady / FairPlay). Roadmap, not beta.

... you need server-side ad insertion or VAST/VPAID. Roadmap.

... you need low-latency live (LL-HLS / LL-DASH). Roadmap.

... the project ships to Tizen / webOS / Roku and needs vendor sign-off.

... you need a stable API contract for a 12+ month support window.

AI-agent affordances (v10 only)

Top-level `llms.txt` with curated low-context doc map.

Framework-specific `llms.txt` per React docs section.

Markdown variant of every doc page via `Accept: text/markdown`.

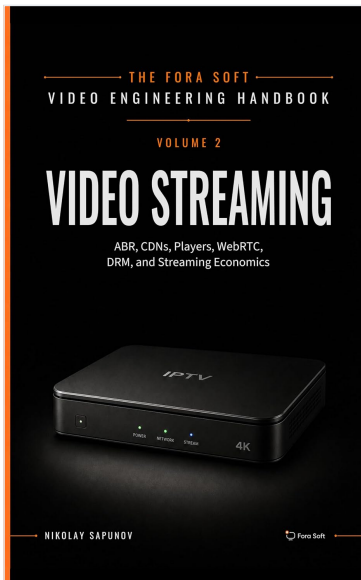
`.claude/` and `.zed/` skill directories in the repo.

Unstyled primitives = less abstraction for agents to navigate.

Engine bundle comparison (with ABR for simple HLS, gzip kB)

Engine	gzip kB	Notes
dash.js	294.2	DASH only. DASH-IF reference.
Shaka Player	239.1	HLS + DASH + Smooth. Multi-format generalist.
hls.js (full)	155.9	HLS only. The web's default HLS engine.
hls.js-light	103.4	No DRM, subs, alt-audio, CMCD, interstitials.
SPF-composed engine	12.1	v10 SPF, simple ABR over CMAF only. 8% the size of full hls.js.

Source: Video.js v10 Beta announcement post, 10 March 2026. Numbers are gzipped, ABR-included for the simple-HLS workload.



GO DEEPER · VOLUME 3 OF 7

Video Streaming: A Complete Guide to Video Delivery

From ABR and CDNs to Players, WebRTC, DRM, and the Economics of Streaming

By Nikolay Sapunov, CEO at Fora Soft · 1,182 pages · Kindle & paperback on Amazon

[Read it on Amazon →](#)
amazon.com/dp/B0HH76ZYWC

This checklist weighs Video.js v10 and SPF against v8, hls.js and Shaka on bundle size, DRM, ads and live. Volume 3 covers the player layer in depth: ABR algorithms, stall recovery, trick play, subtitles, DRM and QoE monitoring.

WHAT YOU WILL FIND INSIDE

- Player architecture and ABR algorithms
- DRM: Widevine, PlayReady, FairPlay
- Stall recovery and trick play
- QoE monitoring in the player

THE FULL HANDBOOK · 7 VOLUMES · 9,000+ PAGES · ALL ON AMAZON

Vol. 1 · Video Encoding — pixels and color to H.264, HEVC, AV1, quality metrics and AI

Vol. 2 · Video Streaming I — network transport, contribution and ingest, HLS, MPEG-DASH, CMAF

Vol. 3 · Video Streaming II — ABR, CDNs, players, WebRTC, DRM and the economics of delivery

Vol. 4 · Audio for Video I — PCM, codecs, loudness, accessibility, real-time voice processing

Vol. 5 · Audio for Video II — jitter buffers, lip sync, Atmos, quality metrics, neural codecs

Vol. 6 · AI for Video I — models and inference, computer vision, speech, search, generative video

Vol. 7 · AI for Video II — real-time pipelines, agents, serving, cost, compliance, verticals

Deciding whether to move your player to Video.js v10?

We will audit your plugins, skin and DRM needs, run v10 against your streams, and plan a migration that does not break playback. 250+ video products shipped since 2005.

[Book a 30-min call →](#)
[WhatsApp →](#)
[Email us →](#)