Cloudflare Platform Report — EdgePointe Vantage

First edition · Saturday, July 18, 2026

For Bob, Niven, and Paul · ~5-minute read

Vantage runs entirely on Cloudflare's developer platform — Workers, D1, KV, Pages, and Workers AI, with R2 pending enablement. This report is a baseline scan of where that platform stands today: what we use, what we should adopt, what's coming, and what could break or bill us if we don't act. Every claim is linked to a primary source.

Executive summary

- **The platform is converging on one model.** Cloudflare is folding Pages into Workers, unifying front-end + back-end into a single deployable Worker with static assets. As of March 2026, Workers reached feature parity with Pages. This validates our stack but means new Vantage features should target Workers-with-assets, not Pages. (Migrate from Pages to Workers)
- **Three billing changes land in 2026** that touch us: Workflows per-step billing (no earlier than Aug 10, 2026), Durable Objects SQLite storage billing (started Jan 7, 2026), and R2 Data Catalog billing (30-day notice pending). None are catastrophic, but all deserve a look before the meter starts. See **Risks** below.
- **Two breaking changes are already in effect** in tooling: `@cloudflare/workers-types` v5 (simplified type entry points) and the removal of service environments from Wrangler. Both affect how we build and deploy; the service-environments change is the one to verify against our CI/CD config.
- **High-value, low-effort adoption targets** for Vantage: Workers Cache (free tiered cache in front of Workers — directly helps our KV-cached publishing), the Images binding (our R2 media pipeline), AI Search / AutoRAG with per-tenant instances (our AI Receptionist grounding), and the Agents SDK voice pipeline (our WebRTC voice work).
- **We are on the right platform.** For multi-tenant edge sites + edge AI + D1, Cloudflare remains the price and global-performance leader versus Vercel and Netlify. This is confirmation, not a migration prompt.

---

1. Capabilities & features — what we use, what we should

Mapped to the Vantage architecture: Genesis Pipeline, Renderer, media serving, AI Receptionist, CI/CD deploys, and per-tenant isolation.

Workers (compute — our core runtime)

- **Static assets = one Worker for site + API.** Workers now serve static assets with full Pages parity (static, SSR, custom domains) as of March 2026. For the **Renderer** this means we can collapse

any split front-end/back-end into a single deployment. *Impact: high · Effort: medium (migration).* (Static Assets, Pages vs Workers 2026)

- **Dynamic Workers (open beta, Mar 2026).** Auto-scaling Workers that hold state across invocations and run tasks up to 30 minutes, booting in single-digit milliseconds. Relevant to the **Genesis Pipeline** for long-running per-tenant build/generation jobs that exceed normal Worker limits. *Impact: medium · Effort: medium.* (Dynamic Workers, open beta, Sandboxing AI agents, 100x faster)
- **Cron Triggers limit removed.** The old 3-cron-per-Worker cap is gone (account limits still apply) — useful for per-tenant scheduled jobs (report generation, cache warming). *Impact: low · Effort: low.* (Workers Changelog)
- **Node.js compatibility improvements** — `performance.now()`, `nodeJsCompatModule`, common Node globals (`process`, `Buffer`) — reduce friction when porting npm libraries into the Genesis Pipeline. (Workers Changelog)

D1 (our per-tenant SQL)

- **GA + 10 GB per database** (up from 2 GB) on Workers Paid, up to 50,000 databases per account (millions by request). This underpins our **per-tenant isolation** model — headroom is no longer a concern for the tenant count we're targeting. *Impact: high · Effort: none (already benefiting).* (Making state easy: D1 GA, D1 Limits)
- **Global read replication (Sessions API).** Read replicas in six regions with sequential consistency, no extra charge — data transfer is already included. For read-heavy tenant sites this lowers latency for far-from-primary visitors. Requires adopting the **D1 Sessions API** in the Worker binding (not yet in REST). *Impact: medium · Effort: medium (code change).* (Global read replication, How D1 implements global read replication)
- **40–60% lower Worker API request latency** from eliminating redundant round trips — a free win already in effect. (Making state easy: D1 GA)

KV + Workers Cache (our cached publishing)

- **Workers Cache (2026) — the headline for us.** A regionally tiered cache that sits directly in front of Worker entrypoints: a lower tier in the data center nearest the user and an upper tier aggregating fills network-wide. On a hit, Cloudflare returns the response *without running the Worker and without charging CPU time*. This maps directly onto our **KV-cached publishing** model — it can cut compute cost and latency on published tenant pages, and tiering is free for any Worker with caching enabled. *Impact: high · Effort: low–medium.* (Your Worker can now have its own cache)
- **KV** remains our fast per-tenant read layer; pricing is per-key with bulk-read billed per key, included in both Free and Paid plans. No breaking change — but Workers Cache is the complementary layer to evaluate. (KV Pricing)

R2 + Images (media serving — pending enablement)

- **R2 core: $0.015/GB storage, zero egress.** This is why R2 is the right home for tenant media once enabled — no egress bill regardless of traffic. (R2 Pricing)
- **Images binding for Workers.** Optimize and manipulate raw image bytes directly in a Worker, sourcing from R2, Images, a `fetch()`, or a request body — enabling a cache-first media pipeline (check cache → check R2 → transform on the fly). First 5,000 unique transformations/month free, then $0.50 per 1,000; no Pro plan required. This is the concrete design for our **media serving** once R2 is on. *Impact: high · Effort: medium.* (Images binding, Optimizing image delivery with resizing + R2)

Workers AI + AI Gateway + AI Search (our AI Receptionist)

- **AI Search (formerly AutoRAG) with per-tenant instances.** Fully managed RAG — ingests, chunks, embeds into Vectorize, retrieves, and generates with Workers AI. Crucially, Cloudflare now supports **dynamic, per-tenant search instances provisioned at runtime** from a Worker. This is the clean grounding architecture for our **AI Receptionist** — one isolated knowledge index per tenant, spun up/down on demand. Instances created after Apr 16, 2026 include built-in storage, vector index, and web crawling. *Impact: high · Effort: medium.* (Introducing AutoRAG, AI Search / multi-tenancy)
- **AI Gateway.** A control plane in front of model calls: caching (identical requests served from cache, up to 90% latency reduction), analytics/logging, rate limiting, retries, and model fallback. A unified REST API (launched May 21, 2026) calls any model through one endpoint. Recommended layer between Vantage and any model provider for cost control and observability. *Impact: high · Effort: low.* (AI Gateway features, AI Gateway caching)
- **New Workers AI models (2026):** Deepgram Aura-2 TTS (`@cf/deepgram/aura-2-en` / `-es`), IBM Granite 4.0, FLUX.2 [klein] 9B image model, GLM-4.7-Flash (131K context), plus regional embedding/LLM models. Pricing is $0.011 per 1,000 Neurons with 10,000 Neurons/day free, now with per-model unit mapping. (Workers AI Changelog, Workers AI Pricing)
- **Agents SDK voice pipeline + realtime.** Real-time voice over WebSockets in ~30 lines server-side, using Workers AI bindings with built-in Deepgram STT (Flux, Nova 3) and Aura TTS — no external API keys to start. Workers AI now supports WebSocket connections (first model: PipeCat smart-turn-v2 turn detection). This is directly relevant to our **WebRTC voice** / **Ask Vantage** work and worth evaluating against our current pipeline. *Impact: medium–high · Effort: medium.* (Add voice to your agent, Cloudflare is the best place to build realtime voice agents)

Queues, Durable Objects, Workflows (underused primitives)

- **Queues now on the Free plan (Feb 2026)** — guaranteed delivery across up to 10,000 queues to Workers or HTTP pull consumers; 10,000 ops/day included; pull ops $0.40/million. Candidate for decoupling **Genesis Pipeline** stages (generate → render → publish) and buffering webhook/lead events. *Impact: medium · Effort: medium.* (Queues on Free plan, Queues Pricing)
- **Durable Objects: SQLite backend now GA and on the Free plan** (Free tier not charged for storage; 5 GB total on Free). Good fit for per-tenant coordination/session state and realtime

features. Note the storage-billing change under **Risks**. (DO Pricing, Zero-latency SQLite in every DO)

- **Workflows** — durable multi-step execution; a fit for reliable Genesis Pipeline orchestration. Adopt with eyes open on per-step billing (**Risks**). (Workflows Pricing)

Wrangler + CI/CD (our deploys)

- **Workers Builds** = native, integrated CI/CD: connect a GitHub/GitLab repo and Cloudflare auto-builds and deploys on push, with build status surfaced as PR comments/check runs. This is a managed alternative to our current GitHub Actions flow — worth comparing for simplicity, though our existing edit→commit→push→CI loop already works. *Impact: low–medium · Effort: low.* (Workers Builds, CI/CD docs)

---

2. Roadmap — what's coming / recently previewed

- **Pages absorbed into Workers.** Cloudflare is turning Pages-specific features into general Workers features; no forced-migration deadline, but every major feature now ships for Workers first. Direction of travel: build new on Workers-with-assets. (Migrate from Pages to Workers)
- **Agents Week 2026 (Apr 13–17).** A large agent-infrastructure push: Dynamic Workers (AI code 100x faster than containers), Sandboxes GA, Cloudflare Mesh (private network access for agents), and managed Agent Memory. Signals where Cloudflare is investing — agentic/AI workloads on the edge, which aligns with Ask Vantage / AI Receptionist ambitions. (Agents Week 2026 Updates, Agents Week infra recap)
- **D1 read replication maturing** toward broader availability (Sessions API today via Worker binding; REST not yet). Expect REST support and continued limit increases. (D1 read replication)
- **AI Gateway unified API (May 21, 2026)** — single endpoint for any model — is the emerging front door for model access; expect deeper caching/observability features. (AI Gateway)
- **Database integrations** (Neon, PlanetScale, Supabase) and direct TCP-to-Postgres from Workers — not our stack today, but expands options if a tenant ever needs relational scale beyond D1. (Workers Changelog)

---

3. RISKS — deprecations, breaking changes & pricing

> **Read this section first if you read nothing else.** These are dated items that can break a build or start a bill.

Item	What changes	Date	Our exposure
Workflows per-step billing	New billing on *steps* + *storage* (each sleep/wait/step counts). CPU + requests already billed.	**No earlier than Aug 10, 2026**	Only if we adopt Workflows. Granular-step designs get penalized — count steps before enabling. (Changelog)

Item	What changes	Date	Our exposure
Durable Objects SQLite storage billing	Storage now metered on Paid plans; Free plan not charged (5 GB cap).	**Started Jan 7, 2026**	Exposure only if we use DO SQLite storage on Paid. Free tier safe. (DO Pricing)
R2 Data Catalog / R2 SQL billing	Catalog ops $9/M; compaction $0.005/GB + $2/M objects; R2 SQL $2.50/TB. Free tiers included.	**Billing not yet on; 30-day notice promised**	Only if we use Data Catalog/SQL (analytics). Core R2 media storage unaffected. (R2 Data Catalog pricing)
`@cloudflare/workers-types` v5	Dated entry points (e.g. `/2023-03-01`) removed; only main + `/experimental` remain.	In effect (Jul 3, 2026)	**Verify our imports.** If we pin a dated types path, the upgrade breaks it. Recommended: generate types via `wrangler types`. (Changelog)
Service environments removed from Wrangler	`--legacy-env` flag and `legacy_env` config field removed; each env deploys as its own `<name>-<env>` Worker.	In effect (2026)	**Verify CI/CD config.** If we set `legacy_env` anywhere, config now errors. Since `legacy_env = true` was the default, removing the field is a no-op — but any explicit `legacy_env = false` changes deploy topology. (Deprecations)
Wrangler v4	Drops Node.js 16 support (follows Node LTS lifecycle); esbuild v0.24 may affect dynamic wildcard imports.	In effect	Ensure CI Node version ≥ supported; check any dynamic imports in the Genesis Pipeline. (v3→v4 migration)
Workers AI model alias/price bump	`@cf/moonshotai/kimi-k2.5` auto-aliased to `k2.6` at a **higher price**.	**May 30, 2026**	Only if we call that model. Audit model IDs in the Receptionist. (Workers AI Changelog)

Net risk posture: Low-to-moderate. The two tooling breaks (types v5, service environments) are the only items that can fail a deploy, and both are quick config checks. The billing items are opt-in — they only bite if we adopt Workflows / DO SQLite storage / R2 analytics. No action forced on Pages, D1, KV, or Workers AI today.

---

4. Competitive slice — are we on the right platform?

We are committed to Cloudflare; this is awareness, not a migration proposal.

For Vantage's specific shape — **multi-tenant edge sites + edge AI + edge SQL (D1)** — the 2026 comparisons consistently land the same way:

- **Performance:** Workers start in under ~1ms (V8 isolates) with sub-50ms latency across 300+ PoPs — ahead of Vercel and Netlify edge functions on cold start and global consistency. (Cloudflare vs Vercel vs Netlify 2026)
- **Primitives:** Cloudflare offers the widest set — Workers, KV, Durable Objects, R2, D1, Queues, Images, Browser Rendering, Workers AI — under one roof. Vercel and Netlify lean on third-party databases for the equivalent of D1. (techplained comparison)

- **Cost:** Free egress (R2) + low compute + generous Workers plan ($5/mo incl. 10M requests) produce a fundamentally cheaper curve at scale — the decisive factor for a many-tenant hosting business. (CoderFile 2026)
- **Where rivals win:** Vercel leads on Next.js developer experience; Netlify on flexibility / lowest lock-in. Neither advantage outweighs price + global performance + D1 for our multi-tenant edge use.

Bottom line: Cloudflare is the default choice for exactly our profile, and the consensus recommendation for multi-tenant edge + edge AI + D1. Staying is the right call.

Recommended next steps (for discussion, not commitments)

1. **Two quick config audits** — confirm no dated `workers-types` import and no explicit `legacy_env = false` in Vantage's Wrangler/CI config (both are deploy-breakers).
2. **Prototype Workers Cache** in front of published tenant pages — likely our highest impact/effort ratio (cheaper compute, faster serves, KV-publishing complement).
3. **Design the R2 + Images media pipeline** now so it's ready the moment R2 is enabled (cache → R2 → transform on the fly).
4. **Pilot per-tenant AI Search instances** for the AI Receptionist to formalize grounding + isolation, fronted by AI Gateway for caching/observability.
5. **Watch the Aug 10 Workflows billing date** if we intend to use Workflows for Genesis Pipeline orchestration; otherwise Queues is the lower-risk decoupling primitive.

Compiled from Cloudflare's official docs, changelog, and blog plus 2026 third-party analyses. This is a Research Report — a first baseline edition for EdgePointe Vantage.