

# p4-cache — Executive Briefing

## The opportunity

Every Perforce shop with a depot above a few terabytes pays for storage capacity at the rate of its worst-case access pattern. In typical environments, 5–15% of depot data is read in any 30-day window; the other 85–95% sits on premium storage — NetApp, Pure Storage, Isilon, or premium cloud SSD — paying enterprise prices for content that is rarely accessed.

p4-cache eliminates this inefficiency. The product is a userspace daemon plus a per-call +X archive trigger binary that transparently tiers cold depot files to commodity object storage — Azure Blob, AWS S3, GCS, or NFS — while keeping the hot working set on commodity NVMe. Perforce itself is unchanged: no patches, no protocol changes, no client software, no developer impact. The result is a storage architecture that pays for capacity at rates aligned to actual access patterns.

## Financial impact

*Recurring annual savings for a multi-petabyte Perforce deployment running on enterprise storage:*

| Depot size | Current storage TCO/yr | With p4-cache/yr | Annual savings     |
|------------|------------------------|------------------|--------------------|
| 200 TB     | \$900,000              | \$182,000        | <b>\$718,000</b>   |
| 500 TB     | \$2,250,000            | \$447,500        | <b>\$1,802,500</b> |
| 1 PB       | \$4,500,000            | \$835,000        | <b>\$3,665,000</b> |
| 2 PB       | \$9,000,000            | \$1,570,000      | <b>\$7,430,000</b> |

**Plus avoided refresh CapEx.** Enterprise storage arrays carry a \$2–3M+ refresh pulse every 4–5 years. p4-cache lets the customer skip that pulse entirely. For a multi-petabyte deployment, the avoided CapEx is on the same order as the recurring annual savings.

## Five-year picture: 1PB Perforce deployment

| Path forward                            | 5-year TCO     | Operational impact                       |
|---|----------------|--|
| Refresh the array                       | <b>\$7–15M</b> | <i>Status quo. Next refresh in 2030.</i> |
| FabricPool to S3 (stay on NetApp)       | <b>\$5–9M</b>  | <i>Tiering helps. Lock-in persists.</i>  |
| <b>p4-cache + commodity NVMe + blob</b> | <b>\$4–5M</b>  | <i>New software license. No refresh.</i> |

## Why this works now

- Modern enterprise NVMe is fast enough to replace SAN for Perforce workloads. A 2U commodity server holds 60–120 TB of NVMe at \$40–80/TB acquisition — an order of magnitude cheaper than enterprise SAN over a 5-year horizon.
- Object storage is durable, well-understood, and 15–100× cheaper per TB than enterprise arrays. Azure Blob Cool at \$120/TB/year vs. enterprise NetApp at \$1,800/TB/year fully loaded.
- The remaining piece — a Perforce-aware tiering layer that operates transparently to `p4d` — is what p4-cache provides. The architecture works because of these three facts together, not any one alone.

## Risk profile

p4-cache is purpose-built for environments where Perforce uptime is non-negotiable and audit trails matter:

- **Reversible by design.** Stop the daemon, remove the `+X` modifier from the typemap, and `p4d` returns to a stock deployment serving whatever is locally resident.
- **Perforce-native high availability.** p4-cache fits inside commit-edge replication; it does not invent a new HA model or modify the existing one. Failover is the same operation as today.
- **Content-hash verified.** Every restore is streaming-MD5 verified against `db.storage.digest` — the same hash `p4d` already stores. A corrupted or tampered file fails the restore rather than reaching `p4d`.
- **Forensic-watermarked.** Every cold-tier write carries a deployment-identifying watermark in object metadata. A leaked or exfiltrated blob is traceable to the licensed deployment that produced it.
- **Audit-logged.** Every depot file read is recorded to Elasticsearch or PostgreSQL with TLS, custom CA, deduplication, batching, and disk-spool resilience.
- **Production-grade.** ~47K-line Rust workspace across six crates (daemon, trigger, shared, license, license-tool, integrity-stamp). Three independent code audits, all engineering findings closed. Zero open HIGH or CRITICAL findings. Workspace-wide compile-time defenses against entire classes of bugs.

### *Engineering maturity, with citations*

*This is the pillar most year-1 commercial products cannot claim. We can. Three independent code audits — quality, security, and verify-binary verification — are documented in the source repo and available under NDA. All engineering findings closed with pinning tests. The workspace enforces, at compile time: every unsafe operation requires an explicit unsafe block; holding mutex guards across await boundaries is a compile error; silent integer casts are denied. Rust edition 2024, MSRV 1.88. Automated gates on cargo audit and cargo deny. Fuzz harnesses on the trigger request handler and the Perforce checkpoint parser.*

## Commercial structure

- Annual subscription, priced per terabyte of managed capacity. Published list ranges from \$7K/yr (5 TB) to \$519K/yr (2 PB).
- Multi-year enterprise license agreements available for deployments above 500 TB. Typical structure: 3-year commitment, paid migration services, reference rights, committed growth bands.

- Free 60-day evaluation — feature-complete, capacity-limited (5 TiB combined managed capacity) — designed for running a real proof-of-concept against a representative subset of your depot.
- Perpetual license option available for procurement environments that prefer CapEx accounting. Priced at 3× annual list plus 18% annual maintenance.
- The runtime carries an Ed25519-signed license envelope. Soft and hard managed-capacity ceilings surface as Prometheus metrics so operator alerting can fire before a hard breach. A hard managed-capacity breach refuses new admissions only — cold-tier uploads keep running (draining the cache is the recovery path), and reads and restores of already-tracked files continue.

## **Next steps**

1. Request the architecture deep-dive, security & compliance brief, and engineering maturity brief for your technical and security teams.
2. Schedule a 30-minute call to discuss your storage baseline, refresh timeline, and proof-of-concept scope.

**info@p4-cache.com** *p4-cache.com*