CITHORUM

Jam Technical Specification

Source-grounded architecture, operational model, security model, integration surface, and benchmark envelope.

Engineer · Cithorum · May 2026

Engineering specification distinguishing documented public/API behaviour from inferred internal pipeline structure. Inferred stages are labelled as such and should be reviewed by Cithorum engineering before customer publication.

1. Source Basis

The canonical source layer is the Cithorum Reworked-v2 archive, especially MASTER-REFERENCE.md and the v2.1 Jam technical documents. Older decks are used for product history, trial evidence, and roadmap context only.

2. Component Map

Component	Role	Documented interfaces / properties	Status
JSL	Jam Standard Library for crypto and RAMOPS primitives.	DeriveSessionKey, PutObject, GetObject, SignRelease, VerifyRelease, SelfTest.	Canonical v2.1.
RAMOPS	Networked storage layer with deterministic layout.	Content-addressed SHA-256 objects, append-only log, quorum writes, bounded-tail replication.	Canonical v2.1.
jamd	Host daemon / managed service process.	Initialised by jam init; systemd service; discovers peers over VLAN/VPN per the install guide.	Canonical install guide + legacy network detail.
jam CLI	Operator control and verification surface.	jam verify --full, jam verify --offline, jam seed init, jam upgrade.	Canonical install guide.
TechConnect	Authenticated M2M transport for OEM/embedded/unattended workloads.	99.9% managed uptime target; AEAD + ECDH; air-gap clause.	Canonical SLA.
SaaS Archiver	Managed immutable archive built on RAMOPS.	WORM retention, legal hold, signed retention policies, signed audit records.	Canonical product brief.
Adapters	Integration points into existing systems.	Local/block storage, S3/cloud, VSAN, VPN, Apache Spark Java/Scala target.	Mixed canonical/legacy; Spark is roadmap.

2b. SSD-Bound Architecture

The single most useful sentence about Jam's run-time profile is this: no GPU, no CPU bottleneck, no RAM bottleneck — only the SSD.

Jam's encoder and decoder are engineered to pin themselves against the underlying NVMe device, not against the CPU pipeline or RAM pressure. The April 2026 instrumented runs on the reference rig observed:

Path	Bound	Observed ceiling	Notes
Encoder	NVMe random-read	281 MB/s	RAM consumption < 400 MB during encode.
Decoder	NVMe sequential-write	1.13 GB/s	Stable across the 64 TB benchmark corpus.
CPU utilisation	Sub-saturating	Single-digit % per worker	Encoder/decoder do not become CPU-bound.
RAM footprint	Constant	Bounded < 400 MB	Decompression is constant-memory.

Operationally this means the engineering rule for sizing is simple: pick the NVMe device first; the rest of the rack falls out of that decision. If the drive can write 1 TB/s, Jam will write 1 TB/s. If the drive caps at 281 MB/s on random read, the encoder caps there too.

3. Data Model

- Objects: immutable content-addressed objects; canonical object ID is SHA-256 in the RAMOPS brief.
- Layout: append-only log and deterministic layout.
- Integrity: per-object SHA-256 and periodic scrub; corrupt objects are auto-repaired from quorum replicas.
- Replication: default quorum writes are 3-of-5; 2-of-3 supported for edge deployments.
- Consistency: consistent reads within quorum and eventual consistency across regions.
- Durability: managed Enterprise and Defence RAMOPS-64 target is 99.999999999% annual object durability.

4. Public Primitive Surface

Primitive	Purpose	Source
DeriveSessionKey(seed, transcript)	HKDF-SHA-256 over BIP39-derived entropy and handshake transcript.	JSL table.
PutObject(content)	Content-addressed write; returns SHA-256 object ID.	JSL table.
GetObject(object_id)	Read by SHA-256 object ID.	JSL table.
SignRelease(artefact)	Ed25519 signature over release artefact.	JSL table.
VerifyRelease(artefact, signature)	Ed25519 verification against release key.	JSL table.
SelfTest()	Invokes jam verify self-test stages 3–5.	JSL table.

5. Ingest and Retrieve Pipeline

The exact proprietary algorithm is not published in the source pack. The operational pipeline below is the safest engineering abstraction because every stage is anchored in source language.

Stage	Documented or inferred behaviour	Do not overclaim
Ingress adapter	Accepts file/folder, block-device, S3/cloud, RAMOPS, TechConnect, or future API input.	Do not imply every adapter is production-ready without a build artefact.
Chunk/object formation	Canonical source says content-addressed immutable objects and append-only layout.	Exact chunking strategy is proprietary/not documented.
Reduction engine	Lossless compression and deduplication; old decks describe AI learning/offloaded RAM.	Do not reuse 48,900× or 140× RAM claims; use canonical wording.
Index/cache path	JSL GetObject and indexed data-loading	Do not claim universal query

	claims support faster compatible AI/ML reads.	engine/database replacement.
Envelope	AEAD transport; deterministic key derivation; no key storage at rest.	Do not say data itself is literally the cryptographic key.
Replicate/store	Quorum writes, periodic scrub, auto-repair, optional multi-region/sovereign modes.	Managed SLOs do not automatically apply to air-gapped deployments.
Retrieve/reconstruct	Lossless reconstruction is core to all demos and product docs.	Benchmark ratios depend on corpus and hardware.

6. Cryptographic Specification

Function	Primitive	Notes
Key agreement	ECDH over secp256k1	secp256k1 is an elliptic curve, not a hash function.
Signature	Ed25519; ECDSA/secp256k1 for interop	Release signing and long-term identity; interop where needed.
Authenticated encryption	AES-256-GCM or ChaCha20-Poly1305	AEAD with per-session IVs derived from session state.
Hashing	SHA-256 / SHA-3 / BLAKE3	SHA-256 for object IDs and integrity; BLAKE3 optional fast content addressing per JSL.
KDF	HKDF-SHA-256	Session-key derivation from BIP39-compatible seed material and transcript.
PQC migration	ML-KEM-768, ML-DSA-65	NIST FIPS 203/204 path; hybrid mode planned.

7. Session Establishment and MITM Hardness

- Threat model: Dolev–Yao adversary can read, drop, delay, replay, and inject messages; endpoint compromise and side-channels are non-goals.
- Session start: endpoints perform ECDH over secp256k1.
- Key derivation: shared secret is mixed with a deterministic per-session nonce and full handshake transcript through HKDF-SHA-256.
- Integrity: AEAD rejects modified or injected ciphertext.
- Replay defence: per-session nonces plus monotonic sequence numbers.
- Forward secrecy: ECDH ephemerals discarded after session close; rekey interval configurable, default documented as 15 minutes.
- BIP39 seed compromise: rotation invalidates derived session state on both sides; procedure referenced from the installation guide.

8. Installation, Verification, and Hardening

Topic	Specification
Supported OS	Ubuntu 22.04/24.04 LTS x86_64/ARM64; RHEL/Rocky 9 x86_64/ARM64; Debian 12 x86_64/ARM64; Windows Server 2022 via WSL2 only; hardened Yocto/DM&V ARM64 by request.
Artefacts	Container image, .deb/.rpm, or air-gap tarball, accompanied by SHA-256 checksum and Ed25519 signature.
Connected install	Install package, run jam init with /etc/cithorum/jam.toml, run jam verify --full, enable jamd.
Air-gapped install	Transfer signed tarball/checksum/signature via approved

	removable media; verify on intermediate workstation and target host; run install.sh --offline; run jam verify --full --offline before go-live.
Seed ceremony	Use customer-controlled BIP39 seed material; disable demo seed.
Hardening	Restrict management API to admin VLAN/VPN; set least-privilege service account; enable audit log shipping; disable outbound telemetry unless customer approves; set retention and rotation policies.

9. Self-Test Specification

Stage	Check	Evidence
Stage 1	Binary integrity: SHA-256 of installed binary vs release manifest.	Hash + manifest match.
Stage 2	Release signature: Ed25519 verification of manifest against release key.	Signature valid/invalid.
Stage 3	Crypto self-test: AEAD, SHA-256, Ed25519.	Known-answer vectors matched.
Stage 4	Determinism: RAMOPS deterministic key derivation from test seed.	Session-key fingerprint stable.
Stage 5	Platform entropy: RNG health check where available.	Pass/fail.

10. Benchmark Methodology and Reference Results

Aspect	Canonical value
Corpus	Five workload classes: sparse telemetry, mixed log, document, binary-heavy, pre-compressed.
Corpus size	64 TB aggregate, each class balanced to 12.8 TB.
Reference hardware	2× AMD EPYC 7543, 512 GB RAM, 4× 7.68 TB NVMe, 100 GbE.
OS	Ubuntu 22.04 LTS, kernel 5.15, io_uring enabled.
Jam version	Jam 2.1.x stable.
Metrics	Compression ratio, write throughput, p50/p95/p99 read latency.
Controls	Baseline raw-mode writes with no Jam processing; Jam default config for compression.
Statistics	Median of five runs reported; IQR and raw runs retained in benchmark pack.

Reference results (per workload class)

Workload class	Compression ratio	Write throughput	Read p95 latency
Sparse telemetry	Up to 100×	6.8 GB/s	1.8 ms
Mixed log	3.1× (69%)	5.2 GB/s	2.2 ms
Document corpus	2.3× (57%)	4.4 GB/s	2.8 ms
Binary-heavy	1.5× (35%)	3.9 GB/s	3.1 ms
Pre-compressed	1.04× (4%)	3.7 GB/s	3.2 ms

10b. April 2026 Backup-Tier Benchmark vs rsync

Cithorum ran a live April 2026 test against rsync as the comparator on a backup-tier corpus to give engineering a concrete number for the workloads operators actually present.

Aspect	Value
Corpus	123 GB backup workload (VM snapshot deltas + system

	images).
Comparator	rsync (default delta-transfer mode).
Result	Jam path delivered ≈100× compression on the same corpus.
Final size on the wire	~1.18 GB.
Encoder profile	281 MB/s NVMe random-read bound; <400 MB RAM; sub-saturating CPU.
Decoder profile	1.13 GB/s NVMe sequential-write bound.
Recording	videopress.com/v/w4Z0jvUC
Reproducibility note	Backup-tier corpus only. Do not generalise to mixed enterprise workloads (use the 35–75% typical envelope for those).

11. Operational SLOs

Service line	Target / behaviour
Managed TechConnect availability	99.9% monthly uptime target for hosted control plane / API surface.
Managed RAMOPS RPO	≤ 15 minutes per incident.
Managed RAMOPS RTO	≤ 60 minutes per incident.
Archive durability	99.999999999% annual object durability.
Archive availability	99.9% monthly control plane + retrieve API.
First-byte retrieve latency	≤ 500 ms p95 on standard tier.
Cold-tier retrieve latency	≤ 4 hours p95.
Air-gapped deployments	Managed uptime does not apply; RTO/RPO defined per deployment.

12. Compliance Posture

Cithorum's external compliance posture is graded as either compliant, aligned (control mapping available; certification not yet held), or planned (roadmap with clear path). Engineering should never overstate the grade.

Standard	Posture	Evidence / note
SOC 2	Compliant	Cithorum SOC 2 attestation document available under NDA in the dataroom.
ISO 27001	Aligned	Control mapping available under NDA; certification not yet held.
ISO 27701	Aligned	Control mapping available; ties into DPDP/GDPR posture.
HIPAA	Aligned	Technical/administrative safeguards mapped; BAA available where required.
PCI-DSS	Aligned	Cardholder data envelope mapped to RAMOPS isolation; not a direct PCI processor.
RBI / India fin-services	Aligned	Sovereign-storage and audit-log posture mapped to typical RBI cyber-security circular.
CERT-In	Aligned	Six-hour reporting playbook documented; logging retention configurable to 180 days.
MeitY Empanelment	Ready	Technical posture matches MeitY cloud-empanelment criteria; empanelment process not yet initiated.
DPDP (India)	Aligned	Data principal/processor distinctions, retention, and consent flows mapped.
FIPS 140/2	Compliant build	Source pack confirms FIPS 140/2 compliance.

FIPS 140/3	Planned	CMVP submission planned Q3 2026 per the source pack.

13. Engineering Roadmap and Open Work

- Publish release-key fingerprint in release notes and security page before broad customer release.
- Complete the Scale-AI reproducible benchmark pack, including raw run files and methodology appendix.
- Build and validate Java/Scala API for Apache Spark cluster integration, especially for Atreides-style DNS/big-query workloads.
- Formalise genomics benchmark reproducibility: FASTQ input corpus, compression ratio, alignment pipeline, baseline aligner, hardware, and raw logs.
- Clarify S3, Azure Blob, GCP, VSAN, and block-device adapter readiness levels per release.
- Complete FIPS 140/3 CMVP package and update docs when certificate status changes.
- Re-run the rsync 100× backup-tier benchmark on a published corpus (with raw logs) so the April 2026 number can be cited externally.

Source Boundary

These documents draw on the canonical Jam source archive maintained by Cithorum. They do not invent new performance numbers. Where a claim comes from an older deck rather than the April 2026 canonical source pack, it is labelled as historical, trial-derived, or requiring formal benchmark attachment before external publication.

- Cithorum Reworked-v2 MASTER-REFERENCE.md (April 17, 2026)
- Jam-Networked-Storage.docx, v2.1, April 2026
- Jam-Standard-Library.docx, v2.1, April 2026
- Jam-MITM-Hardness.docx, v2.1, April 2026
- Jam-Installation-Guide.docx, v2.1, April 2026
- Jam-SaaS-Archiver.docx, v2.1, April 2026
- Cithorum-Scale-AI-Scope-Requirements.docx, v2.1, April 2026
- Cithorum-Licensing-Proposal.docx, v2.1, April 2026
- Cithorum-M2M-TechConnect-SLA.docx, v2.1, April 2026
- Jam-Showcase.html, Reworked-v2 commercial showcase
- Cithorum Jam Company Deck.pdf, May 2025
- JAM - Legacy Deck.pdf, November 2024
- Direct user-supplied SwissVault/Ubuntu benchmark note, April 25, 2026
- M2M Partnership Proposal email thread PDF, April 28-29, 2026
- M2M Tech Connect formal $7K/month MRR confirmed by user, April 29, 2026; Lucas screenshot confirms first invoice sent and Jam for Linux delivered
- RTX / Supplier.io profile acceptance email PDF, April 29, 2026
- UNITE-Brave NATO portal email from NATO Communications and Information Agency PDF, April 29, 2026
- NVIDIA Inception reception invitation email PDF, April 29, 2026
- Cithorum + Jam backup-tier 100× rsync benchmark, April 2026 (recording: videopress.com/v/w4Z0jvUC)