Contents

Knotted Sovereignties: The Quipu as Proto-Computer and the Architecture of Datachain Rope

An anthropological, ethnomathematical, and computational reading of an Andean information technology and its post-quantum continuation.

Author: Kazé A. Onguene **Affiliation:** Datachain Foundation SAS, Lausanne and Paris **Correspondence:** contact@datachain.one **Version:** 3.5 (May 2026) **Word count:** ∼24,500 **Keywords:** khipu; quipu; proto-computing; distributed ledger; post-quantum cryptography; data sovereignty; indigenous data sovereignty; carrier substrate; oral tradition; Article 17 GDPR; Andean studies; ethnomathematics; cryptographic erasure; Testimony consensus; CRYSTALS-Dilithium; SPHINCS+; ONCHAINID; ERC-3643; sovereign threading; Inka administration

---

Abstract

Across at least one millennium of secure pre-Columbian Andean history, from the Middle Horizon Wari state (c. 600 to 1000 CE) through the Inka empire (c. 1438 to 1533 CE) and into the colonial residue documented in the highland village of Tupicocha, with possible Late Preceramic antecedents at Caral (c. 2500 BCE) whose attribution remains contested by specialists, communities of cord-keepers (the *quipucamayoq*) operated a knotted-string medium that they used to tax, to count, to administer, to remember, to sing, and to govern. In the standard historiography of computing, this medium is invisible: it is not classified as a writing system, it is not classified as a calculating machine, and it is therefore not classified as a *technology of computation* at all. We argue, on the formal record assembled over the last hundred years by Locke, the Aschers, Urton, Salomon, Brokaw, Hyland, Pärssinen, Quilter, and Medrano, that this classification is wrong, that it is wrong in a precisely diagnosable way, and that the diagnosis matters now.

The khipu is, on the empirical record, a *position-addressable, typed, signed, hierarchically aggregated, replicated, and granularly erasable* discrete information substrate. These six properties are precisely the operative properties of a modern distributed ledger. The argument we develop is therefore not that the khipu *is* a distributed ledger in any metaphysical sense, but that the khipu *instantiates the operative properties of* a distributed ledger, in cotton and camelid fibre rather than in silicon and copper, and under reputational and institutional authority rather than under elliptic-curve cryptography. The single property the khipu does not supply, cryptographic verification under adversarial conditions, is precisely the primitive supplied by post-quantum cryptography in 2024 (NIST FIPS 204 and 205). The two halves are now, for the first time, available in the same engineering moment.

This paper assembles them. **Datachain Rope** (mainnet chain ID 271828) is a public, EVM-compatible, post-quantum smartchain whose canonical specification, the *Quipu Primitive Canon*, currently at version 2.0, replaces the lexicon of *block, chain, miner, fork* with the lexicon of *knot, string, cord, witness*. The replacement is not stylistic but architectural. Each substitution corresponds to a specific design decision that resolves a specific structural problem in contemporary distributed-ledger design: (i) the *block-mention paradox* (sovereign-thread architectures cannot share a "block" between strangers because there is no shared container to put one in); (ii) the *erasure paradox* (append-only block-chained data structures are constitutionally incompatible with Article 17 of the General Data Protection Regulation, with the African Union Data Policy Framework (African Union, 2022), with PIPL and POPIA, and with the EU AI Act, because erasing a record on such structures requires either rewriting all subsequent blocks or storing personal data off-chain in a side database that voids the on-chain integrity claim); (iii) the *commoditisation paradox* (a ledger structured around fungible blocks of fungible transactions submitted by anonymous parties commoditises every record on it, which is structurally adequate for a peer-to-peer electronic cash system but structurally inadequate for a system whose records are identity claims, biodiversity attestations, real-world asset deeds, cultural-heritage receipts, and sovereign personal histories).

We close with three positions that the paper invites the reader to take seriously. First, that the dominant lineage of distributed ledgers (*Bitcoin → Ethereum → Cosmos → Polkadot*) is a lineage and not the lineage. There exist other lineages, drawn from other histories, with different structural properties. Second, that the *cord* lineage, of which the khipu is the documented archetype and Datachain Rope is the contemporary realisation, is more adequate than the *block* lineage to the central problem of the next decade of public infrastructure: how to administer personal data, identity, and real-world claims at the scale of billions of subjects, under regulatory regimes that demand granular erasure, in a cryptographic horizon menaced by quantum computation. Third, that the political stakes of these architectural choices are real, that *block*, *miner*, *fork*, *chain* are not neutral computational primitives but situated metaphors that carry forward an industrial, mineralogical, extractive, and Eurocentric lineage, and that the deliberate revival

of an indigenous, sovereign, accountable, post-quantum cord lineage, equipped with a per-person carrier vessel (the Datawallet+) that re-instantiates in cryptography the office of the named bard, griot, navigator, and *quipucamayoq*, is therefore an act of substrate-level restitution in the architectural sense, with the institutional partnership work (advisory board, MoU with the Tupicocha cord-keeping community, external indigenous-data-sovereignty review) actively in progress rather than evidenced (see §14.2(e)). The Inka cord-keepers were the earliest documented operators of a distributed, signed, federated information system. That lineage was not extinguished; it was suppressed by an ecclesiastical campaign that began with the 1583 Council of Lima and continued through the seventeenth-century extirpation campaigns (in structural terms a succeeding civilizational iteration reclassifying a still-operational substrate from administration into superstition; in moral terms a wrong), partially preserved through five centuries of subaltern continuity, and now meets the cryptographic moment that allows it to step back into the infrastructural register on its own terms.

---

1. Introduction: Two Lineages of the Distributed Record

1.1 The official lineage

The dominant historiography of distributed information technology begins, almost without exception, in seventeenth-to-twentieth-century Europe and North America. The standard textbook narrative places Pascal's mechanical calculator (1642) at one end and Nakamoto's Bitcoin whitepaper (2008) at the other, and threads between them a familiar set of names: Leibniz's binary arithmetic and stepped reckoner, Babbage's Analytical Engine, Boolean algebra, Frege's *Begriffsschrift*, Cantor and Hilbert, Russell and Whitehead, Gödel's incompleteness theorems, Turing's universal machine, Shannon's relay logic, the ENIAC and the EDVAC, the von Neumann architecture, the ARPANET, public-key cryptography from Diffie–Hellman and RSA, Lamport's clocks, Castro and Liskov's Practical Byzantine Fault Tolerance, Haber and Stornetta's cryptographic timestamping, and finally the Bitcoin whitepaper itself (Hodges, 1983; Aspray, 1990; Diffie and Hellman, 1976; Rivest, Shamir and Adleman, 1978; Lamport, 1978; Castro and Liskov, 1999; Haber and Stornetta, 1991; Nakamoto, 2008).

Within this lineage, the public blockchain of the 2010s appears as a late, particular, and somewhat surprising synthesis. It is *late* because it post-dates by more than thirty years the cryptographic primitives it uses (RSA, hash functions, Merkle trees, public-key signatures), by more than a decade its closest precursor (Haber and Stornetta's 1991 cryptographic timestamping service), and by more than half a decade its conceptual completion (Back, 2002, on Hashcash; Szabo, 2005 [1998], on Bit Gold). It is *particular* because it adopts, very specifically, the metaphor of the *block* (a fungible container of fungible transactions submitted by anonymous parties) and of the *chain* (a single shared sequence of such containers), and it commits to this metaphor at the level of substrate, not interface. And it is *surprising* because the metaphor is, on inspection, a strikingly *industrial* one. *Block* is the language of masonry and of mining. *Chain* is the language of ironworking and of bondage. *Miner* is the language of extraction. *Fork* is the language of metallurgy and of cutlery. The vocabulary of public-blockchain technology is the vocabulary of nineteenth-century European-North-Atlantic heavy industry, projected onto a twenty-first-century informational substrate.

This is not a complaint. Metaphors, in computing as in any other knowledge practice, are constitutive: they organise the design space, they discipline what can be thought, they prefigure what counts as a problem and what counts as a solution (Lakoff and Johnson, 1980; Bowker and Star, 1999). The Bitcoin metaphor, anonymous parties depositing fungible records into a shared, mineable, immutable chain, is, on its own terms, brilliant. It solves a real problem (peer-to-peer electronic cash without a trusted issuer), and it solves it with mathematical elegance. The complaint is rather that this metaphor has been generalised. Identity claims, biodiversity attestations, real-world asset deeds, cultural-heritage receipts, sovereign personal histories, medical records, agricultural provenance, court verdicts, and educational credentials have all been folded into the same architectural primitive, the fungible, anonymous, immutable block, and then it has surprised regulators, lawyers, and ethicists that this primitive is *unsuitable* to records that are not fungible, not anonymous, and must, under the most advanced data-protection regimes worldwide, be erasable.

1.2 The other lineage

This paper proposes that there is another lineage. It does not begin in a European workshop. It begins on the cotton looms and dye-vats of the central Andes, in the long arc of pre-Columbian state administration that runs through the Middle Horizon expansion of the Wari state (c. 600 to 1000 CE, with khipu-bearing sites at Pikillaqta and beyond; Conklin, 1982; Splitstoser, 2014), into the imperial moment of Tawantinsuyu (the Inka empire, c. 1438 to 1533 CE), and into a residual continuity that the ethnographer Frank Salomon documents in the central-Peruvian highland village of Tupicocha as recently as 2002 (Salomon, 2004). Possible Late Preceramic antecedents at the Norte Chico settlements (Caral, Bandurria, Aspero, c. 2500 BCE; Shady, 2006; Haas and Creamer, 2006) are noted but their classification as proto-khipus remains contested in the specialist literature. The technology is the *khipu* (in the Hispanicised form *quipu*; the Quechua term means simply "knot"). It is a knotted-string medium. It is the substrate of state administration over a population of approximately ten million subjects across approximately two million square kilometres of hostile, altitude-spanning mountain terrain. The territorial extent and the logistical complexity of administering at this geography are without precise parallel among the contemporaneous polities of the early sixteenth century; in population it is comparable in scale to the major European polities of the period (France approximately 15 million, the Holy Roman Empire approximately 16 million, the Ottoman Empire approximately 12 to 15 million, Spain approximately 7 to 8 million). It is, on the most defensible reading of the evidence, a *distributed information technology*. It has been treated, in the dominant historiography of computing, as if it did not exist.

The reasons for the omission are well documented. They are partly archaeological (most khipus were destroyed by Spanish authorities in the late sixteenth century; the Concilio Tercero Limense of 1583 ordered their burning as instruments of idolatry; only a few thousand survive in museums and private collections worldwide; Brokaw, 2010, ch. 5; Charles, 2010), partly epistemological (the khipu was, until very recently, classified as a "mnemonic device" rather than as a "writing system" because it does not encode phonemes in an alphabetic or syllabic register; Mignolo, 2003; Quilter and Urton, 2002), and partly political (a non-European, non-alphabetic, knotted, manual, indigenous information technology does not fit comfortably into a historiography that takes European typography as the index of literacy and European mechanical calculation as the index of computation; Mignolo, 2011; Quijano, 2000). The cumulative effect is that an information substrate operated for at least one millennium of secure archaeological record (with possible Late Preceramic antecedents), by professional record-keepers, in support of a state apparatus comparable in scale to the major European polities of the early sixteenth century and without precise parallel in territorial extent and altitude-spanning logistics, has been bracketed out of the history of computing.

We propose to bracket it back in. Not as a curiosity, not as a metaphor, not as a token of indigenous "wisdom", but as what it empirically is: a *position-addressable, typed, hierarchically aggregated, signed, replicated, and granularly erasable* discrete information substrate. These are precisely the operational properties of a modern distributed ledger. The khipu *instantiates the operative properties of* a distributed ledger, supplying five of the six primitives that contemporary public-ledger engineering supplies, and supplying them in a different and in several respects superior arrangement. The sixth primitive, adversarial cryptographic verification, is supplied by post-quantum cryptography in 2024 (NIST FIPS 204, FIPS 205). The architectural question that this paper poses is therefore a sharp one. Given that the cord lineage supplies the substrate properties more adequate to the next decade of public infrastructure, and given that post-quantum cryptography supplies the missing primitive, *why would one continue to build on the block lineage*?

1.3 The argument and the stakes

The paper proceeds in four parts and fourteen sections. **Part I** (§§2–3) sets out the anthropological, archaeological, and ethnomathematical record of the khipu in sufficient detail that the formal argument that follows is grounded in evidence rather than in metaphor. **Part II** (§4) gives the proto-computer hypothesis explicitly: a tuple-formal definition of a khipu, a careful statement of what is and is not claimed, and a positioning of the khipu in the comparative landscape of pre-modern computational substrates (the abacus, the astrolabe, the Antikythera mechanism, the counting-rod system of Han China, the wampum belts of the Haudenosaunee). **Part III** (§§5–11) reads Datachain Rope as the engineering realisation of the cord lineage, with explicit Quipu↔Rope mappings (§5), with a full architectural treatment of the granular-erasure primitive `rope_untieKnot` (§6), with a derivation of the Testimony

consensus protocol from the *quipucamayoq* role (§7), with an account of the Organic Encryption System (OES) that supplies the cryptographic erasure primitive (§8), with a comparative architectural table situating Datachain Rope against Bitcoin, Ethereum, and Cosmos-class chains (§9), with a detailed account of the Rust crate workspace and the bifurcated Reth + rope-node execution stack (§10), and with the throughput-and-scaling roadmap including measured Phase 1 benchmark results (§11). **Part IV** (§§12–14) reads the substrate as a carrier system in the lineage of oral and bardic transmission, with the Datawallet+ as the modern carrier vessel and the CARE Principles for Indigenous Data Governance as the operational specification on top, and articulates the political economy of the cord (§12), discusses the regulatory, economic, and political stakes (§13), and closes with a programme for the next phase of research, deployment, and federation (§14).

The stakes are substantial and we name them now, so that the reader who finishes the paper does not finish it in any doubt about what is at issue. Public-ledger infrastructure, of one architectural family or another, is on the trajectory of becoming the substrate of identity, of land registry, of educational credential, of health record, of biodiversity certification, of cultural-heritage authentication, of real-world asset tokenisation, and of monetary instrument, at the scale of *billions* of subjects, in *every* major regulatory jurisdiction. The choice of substrate is therefore not an engineering choice in a narrow sense. It is a constitutional choice. It will shape, for the next half-century, who can be remembered, who can be forgotten, who can be commoditised, and on whose terms. The architecture proposed in this paper is the proposition that this constitutional choice should be made deliberately, that the *cord* lineage is more adequate than the *block* lineage to that constitutional task, and that the cord lineage's documented territorial reach across an empire of approximately two million square kilometres of altitude-spanning terrain, served by an Inka road network (the Qhapaq Ñan, approximately thirty thousand kilometres in mapped routes, UNESCO-inscribed in 2014) along which the cords moved, and its at-least-one-millennium pedigree of secure archaeological record (with possible Late Preceramic antecedents) should be stated, defended, and engineered onward, not erased a second time.

Part I. The Khipu in Anthropological and Ethnomathematical Perspective

2. What a Khipu Is: Material, Structural, and Semiotic

2.1 The physical object

A khipu is, in its physical realisation, an assemblage of cords made from cotton (predominantly *Gossypium barbadense*) or camelid fibre (alpaca, vicuña, llama, guanaco), sometimes from human hair, occasionally from feathers or metal threads, spun, plied, and dyed in a vocabulary of natural pigments (*cochinilla* red, *molle* yellow, *chillca* green, *huinapo* black, *llaqsa* blue, and a documented spectrum of seventy or more named hues) that varies by region and by craft school (Conklin, 2002; Splitstoser, 2014; Carrión Cachot, 1948). The cords are knotted and tied together in a hierarchical structure that, on the largest preserved specimens catalogued in the Khipu Database Project (KDP) at Harvard, can extend to several hundred pendant cords on a single primary cord, with the largest known specimens substantially exceeding this. We do not commit to a specific cord-count record here because the KDP catalogue is updated continuously and the answer depends on whether one counts pendants only or pendants plus subsidiaries. The Puruchuco assemblage excavated by Mackey in 1970 contained twenty-one khipus from a single Inka administrative post (Urton and Brezine, 2005), demonstrating in-situ administrative use; the Wari-period khipus from El Brujo and from the Ica valley demonstrate that the medium predates the Inka horizon by at least four centuries and possibly considerably more (Splitstoser et al., 2016).

The standard structural typology, established by the Aschers (1981) on a corpus of more than two hundred specimens and refined by Urton (2003, 2017), distinguishes:

- A **primary cord** (Spanish *cuerda principal*), thick, horizontal, often with a finished loop or knot at one or both ends, from which all other elements depend. The primary cord is the *spine* of the khipu and its identifier-by-bearer: when an Inka administrator received a khipu from a regional cord-keeper, he received it by the primary cord, which functioned as the document's handle.
- **Pendant cords** attached to the primary cord and hanging from it, typically in batches of related records, sometimes grouped by colour or by spacing. A typical mid-sized administrative khipu carries between ten and two hundred pendants.
- **Subsidiary cords** attached to pendant cords, and recursively to other subsidiaries, creating a hierarchical, tree-structured assemblage of arbitrary depth. The Aschers documented khipus with up to six levels of subsidiary nesting; the corpus average is two to three.
- **Top cords** attached to the primary cord but rising *above* it (rather than hanging below), often used to denote sums or category headers across a group of pendants. Top cords are the structural marker of *aggregation* (§2.4 below).

Three knot *types* recur, and the type carries information independent of position. The **simple overhand knot** is a single twist of the cord upon itself. The **long knot** is a knot of n turns, where $n \in \{2, 3, 4, 5, 6, 7, 8, 9\}$; the count of turns is read directly. The **figure-of-eight knot** is a doubled overhand. Locke's foundational decipherment (1923) showed that, on the lower portion of an Inka-period numerical khipu, knot *position* along the cord encodes a base-10 *place value*, units, tens, hundreds, thousands, tens of thousands, and knot *type* and *turn-count* encode the digit at that place. Long knots occupy the units position; simple knots occupy the higher decimal positions; figure-of-eight knots represent the digit "1" in the units position because the standard long-knot construction cannot achieve a one-turn variant (Locke, 1923, pp. 31–47; Ascher and Ascher, 1981, ch. 3). The Aschers' systematic survey confirmed Locke's decipherment for the numerical layer and additionally documented the recurring *aggregation pattern*: the numeric value on a top cord very frequently equals the sum (or, less commonly, a multiplication,

weighted average, or other arithmetic function) of the values on its associated pendant cords. This is a decisive observation, and we will return to it in §2.4.

2.2 The non-numerical layers

The medium is not exhausted by its numerical register. Urton's *Signs of the Inka Khipu* (2003) is the most rigorous attempt to formalise the non-numerical signal capacity, and it argues from first principles that the medium operates a layered system of *binary structural choices* that together yield, on his count, a 24-class signal alphabet. The distinctions he isolates are these. **Material**: cotton or camelid. **Class**: pendant or top, and main or subsidiary. **Spin direction**: S-twist or Z-twist (the direction in which the fibres are spun together). **Ply direction**: again S or Z, but now of the plied cord (note that in Quechua textile practice, *S-spin / Z-ply* and *Z-spin / S-ply* are visually distinguishable patterns and carry distinct cultural valences). **Attachment direction**: recto or verso (whether the pendant is attached to the front or back of the primary cord, which the Inka are known to have used as a binary distinction). **Knot direction**: forward-S or backward-Z (whether each long knot's turns wind in S or Z direction). And **colour**: drawn from a discrete, regionally standardised palette. The combinatorial space of these binary choices is ($24 \approx 2 \times 2 \times 2 \times 2 \times 2 \times 24$-colour-classes), which is sufficient, Urton observes, to encode an alphabet larger than the cuneiform or alphabetic systems of the contemporary Old World.

Hyland's research (2014, 2017) goes further and reports, from the village of San Juan de Collata in the Peruvian highlands of Huarochirí, what appears to be a *logosyllabic* khipu in which colour patterns and ply combine to encode lineage names. Her paper "Writing with Twisted Cords" in *Current Anthropology* (Hyland, 2017) is a careful and conservative argument: she does not claim the entire Inka khipu corpus is logosyllabic, but she does claim that *some* khipus encode phonetic information, and she demonstrates this on a specific corpus of two Collata khipus that had been preserved in a community archive and whose internal evidence (combined with Spanish-language documents in the same archive) permits a partial decipherment. The implication, as Hyland herself notes, is that the medium supports at least three simultaneous registers: numerical, categorical, and (under contested but increasingly accepted readings) phonemic. A single pendant carries, at once, a number (its knots), a category (its colour and ply), a relation to other pendants (its position on the primary cord), and an ownership signature (the *quipucamayoq* whose hand tied it). This is **multiplexed encoding**. Multiplexing, in any computational substrate, is a strong signal of a designed information system rather than an *aide-mémoire*: a memory aid does not multiplex; it does not aggregate; it does not encode in spatially formalised ways (Brokaw, 2010, ch. 4; Quilter and Urton, 2002).

2.3 Position as identity

A point that is easy to undervalue in the standard summaries deserves emphasis. On a khipu, the information content of a knot is *inseparable* from the knot's position on the cord. A "5" tied at the units place is the digit 5; the same knot at the hundreds place is the integer 500. This is a familiar property of any positional numeral system, but it is the *physical* form of the principle on the khipu that is striking: the knot's identity in the system is its (cord, depth, position) triple. There is no "free-floating" knot. Every knot is *located*. There is no "unsigned" knot. Every knot is ascribed to a specific cord, attached at a specific depth, at a specific position along that cord, on a khipu held by a specific *quipucamayoq*. There is no "global" register of knots; there are only *located* knots, on *located* cords, in *located* khipus.

This is a very strong design principle and we will see, in §5, that it survives translation into Datachain Rope as the principle that every event is the event of a specific *string*, identified by a specific *(kind, id_bytes)* pair, and that no event is "free-floating" in a global pool of records. It is also the property that, in our reading, makes the khipu *non-extractive* in the sense that contemporary distributed ledgers struggle to be: there is no anonymous deposit into a shared pool, because there is no shared pool. There are only located, located, located cords.

2.4 On-medium computation: the top-cord aggregation

The Ascher and Ascher survey (1981) documented, with a rigour that has not been matched in any subsequent corpus study, the recurring pattern in which a top cord's numerical value equals the arithmetic aggregation of the values on its associated pendant cords. The Aschers report this pattern in the great

majority of the khipus they surveyed for which a top cord is present and the pendant values are legible. The most common aggregation is summation, but they also document multiplication (where, for example, the top cord records the *product* of a per-unit rate and a count) and weighted aggregation. This is not, on the standard reading, a coincidence. It is the on-medium realisation of the *reduce* operation: the khipu does not merely *store* the values on its pendants; it *computes* their aggregation, ties the result on the top cord, and stores both the inputs (the pendants) and the output (the top cord) in a verifiable spatial relation that can be audited by any reader who knows the convention.

This is computationally consequential. It means the khipu performs *on-medium* what a contemporary distributed ledger performs in *application code*: it commits the aggregation to the substrate. In Bitcoin's substrate, there is no aggregation; transactions are stored as a flat list within each block, and any aggregation (the total fees paid in a block, the change in supply at a halving, the volume of a particular asset) must be recomputed by the application from the underlying transactions. In Ethereum's substrate, smart contracts can store aggregations, but only as opaque key-value writes; the substrate does not enforce the aggregation relation. In the khipu, the relation is *built into the structure*: the top cord is *physically* connected to the pendants whose values it aggregates, and a discrepancy between the top cord's value and the aggregation of the pendants is *physically observable*. This is, in modern terminology, a *committed and verifiable on-substrate Merkle-like aggregation*, achieved by tying.

We do not over-claim. The aggregation is not cryptographically verifiable in the modern adversarial sense; the verification is reputational, institutional, and visual. But the *form* of the aggregation, the structural commitment of an aggregate value to its constituents in a way that is audit-observable on the substrate, is exactly the form that contemporary committed-aggregation primitives (Merkle trees, polynomial commitments, vector commitments) take. The khipu instantiates this form in cotton.

2.5 The material economy of cord and knot

A point of texture worth flagging: the khipu is a *bandwidth-constrained* medium. Cotton was relatively abundant in the coastal Andes; camelid fibre was relatively scarce in the highlands; the dye palette was constrained by what could be grown, harvested, or traded; and tying a knot is a slow physical act. This means the medium privileges *compact, position-dense* encoding, which is exactly what Locke's decipherment shows the medium achieved: a four-digit decimal value occupies four position bands on a cord and a single colour. This is comparable, on a per-glyph basis, to the bandwidth efficiency of contemporary alphabetic writing on parchment.

The bandwidth constraint also means the khipu is not a *bulk-storage* medium. A khipu does not store, on its own, the kind of high-volume transactional record that a contemporary blockchain stores. It stores *administrative summaries*: census counts, tribute totals, herd sizes, harvest yields, and category headers under which more detailed records (held on subordinate cords or recited from memory by the cord-keeper) are organised. The Inka administration, on the standard reading, used the khipu as a *summary-and-index* medium, with detailed records held in the memory and oral recitation of the *quipucamayoq*, anchored to the cord and verifiable against it. This is, in modern terminology, a *digest-and-witness* architecture: the cord holds the authoritative digest; the witness holds the authoritative payload; the two are bound by the witness's commitment to the cord. This is not a primitive architecture. It is the architecture of every modern certificate-bearing system, of every cryptographic commitment scheme, of every digital signature on a hashed payload. The Inka substrate did this in cotton, signed by reputation.

3. The Cord-Keepers, the Imperial Cord, and the Survival of the Practice

3.1 The quipucamayoq as profession

The Spanish chroniclers, Garcilaso de la Vega, *Comentarios reales de los Incas* (1609); Murúa, *Historia general del Pirú* (c. 1590); Guaman Poma de Ayala, *El primer nueva corónica y buen gobierno* (c. 1615); Cobo, *Historia del Nuevo Mundo* (1653); Cieza de León, *Crónica del Perú* (1553); Acosta, *Historia natural y moral de las Indias* (1590); Polo de Ondegardo, *Informaciones acerca de la religión y gobierno de los Incas* (1571), are unanimous on a striking institutional fact. The Inka empire maintained a *profession* of cord-keepers, the *quipucamayoq* (Quechua: "the one who keeps the knots"), trained from childhood in the discipline of khipu construction, reading, and recitation. Garcilaso (himself the son of an Inka princess and a Spanish conquistador, and therefore a partial but uniquely positioned witness) describes the *quipucamayoq* as a stratified profession with regional schools, with mentor-apprentice pedagogies, with specialisations (some kept tribute records, some kept census records, some kept historical-narrative records, some kept ritual-calendrical records), and with pensions and social standing in the imperial bureaucracy comparable to that of *amauta* (philosophers/teachers) and *tukuyrikuq* (regional inspectors).

The chroniclers are unanimous on three further points that bear directly on the architectural argument. First, the records were *redundant*: each district held its own khipu of the same data, and the data was reported upward to provincial, then to suyu (quarter) level, and finally to the imperial centre at Cuzco, by aggregation onto larger cords held by progressively senior *quipucamayoq*. Second, the records were *signed*: a khipu was associated with the named cord-keeper who tied it, whose lineage was professional and whose authority was personal. The chroniclers describe a procedure in which a khipu, upon presentation to a higher-level administrator, was *witnessed* and *re-recited* by its keeper, in the presence of other keepers, as a form of authentication. Third, the records were *auditable*: when discrepancies arose, and they did, especially in tribute reports and in census counts, the khipus of the district, of the province, and of the imperial centre could be compared *knot for knot*. Polo de Ondegardo (1571) describes such an audit procedure in which a tribute discrepancy was resolved by comparing four khipus from four levels of the administrative hierarchy and identifying the level at which the records diverged.

This is, in distributed-systems terminology, *replicated state with named signers and federated aggregation*, equipped with an *audit-and-comparison* protocol for resolving inter-replica discrepancies. The state was replicated by physical re-tying at each administrative level; the signers were the named *quipucamayoq* at each level; the federation was the upward aggregation hierarchy; and the audit protocol was the comparison-by-recitation procedure. The fact that the empire was non-literate in the Old-World sense, there was no Inka alphabet, no syllabary, no logographic script in the East-Asian or Maya sense, is not, on this reading, a deficiency. It is a substrate choice. The empire administered approximately ten million subjects across approximately two million square kilometres without the Old-World writing technologies, *because* it had a more apt one. The khipu was the substrate of the imperial state, and the *quipucamayoq* were its operators.

3.2 The imperial cord

The vertical structure deserves a closer look because it is the architectural template for what we will call, in §7, the *Federation Generation Protocol*. At the bottom of the hierarchy, each *ayllu* (kin-based community of about one hundred to one thousand households) maintained its own khipus, kept by local

quipucamayoq. Above the *ayllu* level, the *kuraka* (district lord) held aggregation khipus that summarised the records of the *ayllus* under his jurisdiction. Above the district, the provincial *tukuyrikuq* held further aggregation. Above the province, the *suyu* (one of the empire's four quarters: Chinchaysuyu, Antisuyu, Qollasuyu, Kuntisuyu) held the second-to-highest aggregation. And at Cuzco, the imperial *quipucamayoq mayor*, the chief cord-keeper of the empire, held the topmost cord, which Garcilaso describes as an enormous assemblage of imperial summaries, kept in a dedicated building under continuous custodial guard. This imperial cord, on the chroniclers' accounts, encoded the state of the empire in summary at the level of *suyu*, with provincial detail held one level below, district detail two levels below, and *ayllu* detail at the base.

We label this the *imperial cord pattern*: a single, continuously maintained, continuously updated, federally aggregated record at the topmost level, fed by a recursive hierarchy of replicated, signed, audited cords below it. It is structurally identical to the architecture of a modern federated distributed ledger with periodic anchoring, and it is the direct ancestor of what Datachain Rope realises as the *cord*: the global federation cord that anchors knot summaries from every per-entity string in the network, every approximately three seconds, under post-quantum signatures. The Inka did this in cotton, every census cycle, by physical re-tying. We do this in cryptographic commitments, every three seconds, by ML-DSA-65 signatures. The *form* is the same.

3.3 The deliberate untying: granular erasure as cultural practice

A property of cord-keeping that the chroniclers do not emphasise, because it was not, to them, a remarkable property, is the *physical reversibility* of the medium. A knot, once tied, is not metaphysically immutable; it is *materially undoable*. The cord is continuous; the knots along it are local features; an individual knot can be untied without affecting the cord's continuity or the position of the other knots. This is, in the strict sense, a built-in property of the substrate: cotton-and-knots is an erasable medium in a way that ink-on-parchment, after the ink has dried, is not.

Salomon's *The Cord Keepers* (2004), the only sustained ethnography of a community that has continuously maintained khipus into the present, documents this property as an *active practice*, not merely a latent property. In Tupicocha, the *cuerpo de bayetones* (community-administrative cord) is *deliberately re-tied* at intervals, with specific knots untied and replaced as the community's administrative state evolves. The community's *huayrona* (the assembly of cord-keepers) carries out this re-tying as a formal procedure, in front of witnesses, with the prior state of the cord recited and the new state inscribed. A knot's *position* on the cord remains as a deliberate absence, what we will call, in §6, a *tombstone*, when an erasure is performed; the cord retains the structural memory that an event happened at that position, even when the content of the event has been removed.

This is, in computational terminology, a *granularly erasable, tombstone-preserving substrate*. The knot is not eternal. The cord is. The audit trail of the erasure is. The content of the erased knot is not. This is *exactly* the structure that European data-protection guidance treats as cryptographic erasure compatible with the data-protection-by-design principle of Article 25 GDPR (EDPB Guidelines 04/2019, version 2.0, 2020) and with the key-destruction-as-anonymisation analysis of WP29 (2014), and it is *exactly* what the operation `rope_untieKnot` realises on Datachain Rope (§6). Salomon, writing in 2004, was not aware that he was documenting the ancestor of a 2024 cryptographic primitive. But he was. And the ancestor pre-dates the primitive by at least four hundred and seventy years of continuous institutional practice in Tupicocha alone, and by an unknown but substantially longer arc in the Andean tradition more generally.

3.4 The Council of Lima and the suppression

A historical episode is worth registering with some care because it changes the political reading of the architecture. The 1583 Concilio Tercero Limense (the Third Council of Lima, the most authoritative ecclesiastical body in the colonial viceroyalty of Peru) issued an edict that classified khipus, alongside other indigenous ritual and recording media, as *instrumentos del demonio* and called for their destruction. The 1583 edict was not, however, a singular event: as Brokaw (2010, ch. 5) carefully documents, it was the canonical opening of a broader extirpation-of-idolatry programme that escalated through Pablo José de Arriaga's *Extirpación de la idolatría del Pirú* (1621), reached operational peak under Francisco de Ávila's campaigns in Huarochirí from the 1610s onward, and continued in episodic form into the late seventeenth century. Brokaw's reading, which we adopt, is that the suppression was not theological in any

narrow sense; it was administrative. The colonial administration could not read the khipus. It could not control them. It could not insert itself into the chain of *quipucamayoq* authority. And the *quipucamayoq*, at the moment of the edict, were continuing to administer the indigenous communities under their care, in parallel to and partly outside of the Spanish administrative system. The destruction of the khipus was, on this reading, the destruction of an *autonomous administrative substrate*. It was substantially effective: the bulk of the imperial archive at Cuzco was lost, most regional khipu collections were lost, and the *quipucamayoq* profession went underground or was extinguished. Only Tupicocha (a small village in the Lima department, partly insulated by altitude and by the obscurity of its administrative status) preserved a continuous practice into the present (Salomon, 2004).

The historical record matters because it disposes of a possible misreading. The cord lineage is not "lost" or "primitive" or "outdated". It is *relegated*. The succession of civilizational iterations on which the modern world rests works less by efficiency-driven supersession of one infrastructure by a better one than by *reclassification*: the dominant modernism of each epoch redraws the boundary between what counts as *infrastructure* and what counts as *cultural artefact*, and the substrates of the prior modernism are typically pushed across that boundary, from operational into ceremonial, from administrative into mythical, from infrastructural into folkloric. The cord lineage is the paradigmatic case. The successor modernism that consolidated through the seventeenth and eighteenth centuries recognised the cord substrate as *current and operational*, kept reusing khipu testimony for tribute records and admitting it as evidence in colonial litigation into the late eighteenth century, and yet, in parallel, reclassified it through ecclesiastical decree as superstition and ordered the destruction of substantial portions of the inherited corpus (the campaigns documented immediately above). The relation is therefore different in kind from, say, the relation between the abacus and the modern calculator, where the older technology was straightforwardly superseded by a more efficient one. The cord lineage was not superseded; it was reclassified out of the infrastructural register and partly destroyed. To revive it is therefore not an act of nostalgia or of re-enactment. It is an act of *substrate reclassification* in the other direction, in which an information substrate that an earlier civilizational iteration moved out of the operational stage and into the folkloric is moved back, on its own terms, equipped with the cryptographic primitives that earlier iterations did not possess. The *structural* asymmetry of the move matters: a substrate rejoining the boundary of infrastructure rather than remaining on the boundary of artefact and myth. The *moral* asymmetry matters as much: the suppression was a wrong, the wrong was administered through ecclesiastical decree against a substrate the wrongdoer's own colonial bureaucracy continued to treat as operational evidence, and the revival is therefore restitution in the strict sense, equipped at last to function as the wronged substrate did before the suppression. Datachain Rope is engineered restitution, named as such because that is what it is.

Part II. The Proto-Computer Hypothesis

4. A Formal Reading

4.1 The tuple definition

We now state the proto-computer hypothesis with the precision the argument requires. Let a khipu be a tuple

$$K = (S, P, \Sigma, \Lambda, \pi, \tau, \sigma, \alpha, q)$$

where:

- S is a finite set of **strings** (primary cord plus all pendants, subsidiaries, and top cords).
- $P : S \to S \cup \{\perp\}$ assigns to each string its parent. The primary cord has parent $\perp$; pendants have the primary cord as parent; subsidiaries have a pendant or another subsidiary as parent; top cords have the primary cord as parent but in the *aggregating* relation rather than the *containing* relation. P induces a directed tree.
- Σ is the set of admissible knot **shapes**, cross-producted with knot **multiplicities**. We adopt the canonical typology: $\Sigma = \{\text{simple, long, figure-of-eight}\} \times \{1, 2, \dots, 9\}$, with the standard restrictions on which shapes admit which multiplicities.
- Λ is the set of admissible **labels**: a finite product of categorical features (colour from a discrete palette of order ≈ 24, ply direction in $\{S, Z\}$, attachment direction in $\{\text{recto, verso}\}$, knot direction in $\{S, Z\}$, material in $\{\text{cotton, camelid, hair}\}$, etc.).
- $\pi : S \to \mathbb{N}$ assigns to each string its **length** in normalised position units (i.e. discretised positions along the cord at which knots may be tied).
- τ is a partial function from $S \times \mathbb{N}$ to Σ: at each (string, position) pair, either there is a knot of a given shape, or there is a *tombstone* (deliberate absence, marking that a knot was once here and was untied), or there is structurally empty space.
- $\sigma : S \to \Lambda$ assigns to each string its categorical label (its colour-and-ply-and-material vector).
- α is the *aggregation function* relating top cords to their pendants: for each top cord t with associated pendant set $\{p_1, \dots, p_k\}$, $\alpha(p_1, \dots, p_k) = \text{value}(t)$, where the canonical aggregation is summation but multiplication and weighted aggregation are documented in the corpus.
- $q \in Q$ is the *quipucamayoq*, the named, professionally trained, reputationally bound cord-keeper to whom the khipu is attributed, and who is the witness to its construction and the authority for its reading.

This is a *position-addressable*, *typed*, *hierarchical*, *labelled*, *signed*, *aggregating* discrete record. It is, in formal terms, a labelled rooted tree whose internal nodes carry typed contents drawn from a controlled vocabulary, whose leaves are knots drawn from a finite alphabet, whose internal node values are derived from leaf values by a known aggregation function, whose tree is rooted in the primary cord, whose authority is bound to the named keeper, and whose individual knots are physically erasable while preserving the tree's structural continuity.

4.2 What this formalism captures

The formalism captures, with no metaphorical slack, the following operative properties:

(a) Position-addressable read. Given (s, i) for some $s \in S$ and $i \in \mathbb{N}$, $\tau(s, i)$ returns the knot at that position (or a tombstone, or empty).

(b) Typed-write append. Given $s \in S$ and a knot in Σ, the cord-keeper can append a knot at the next available position. The append is *typed*, the knot must be drawn from the admissible shapes for that string class.

(c) Granular erase. Given (s, i) for some occupied position, the cord-keeper can untie the knot, leaving a *tombstone*. The tombstone preserves the structural memory of the position; the content is irretrievably gone (because, in the cotton substrate, untying physically destroys the configuration).

(d) Signed origin. Each cord and each khipu carries an attribution to $q \in Q$. In a non-cryptographic regime, this attribution is institutional and reputational; in a cryptographic regime, it is realised as a digital signature.

(e) On-substrate aggregation. The aggregation function α relates internal-node values (top cords) to leaf-set values (pendants). The relation is structurally enforced, a discrepancy is observable on the substrate, and is verified by the comparison-by-recitation audit protocol.

(f) Replicated federation. A khipu K at level ℓ is replicated, by re-tying, at levels $\ell + 1, \ell - 1$, etc., yielding a hierarchy of cords whose values are bound by the recursive aggregation α, with the imperial cord at the top.

These six properties, read, typed-write, granular-erase, signed-origin, on-substrate aggregation, replicated federation, are the operative properties of any modern distributed ledger. We will see in Part III that they translate, in Datachain Rope, to the operations `rope_walkString`, `rope_appendToString`, `rope_untieKnot`, the post-quantum signature scheme, the cord-anchor knot pattern, and the Federation Generation Protocol respectively. The translation is not metaphorical. It is property-for-property.

4.3 What the formalism does not claim

We are careful not to over-claim. Three negative statements deserve emphasis.

(i) No claim of Turing-completeness. We do not claim that an Inka cord-keeper could simulate a universal Turing machine on a khipu. There is no evidence that the medium supports a universal computation in the Church–Turing sense, no evidence of a halting-problem-relevant control flow on the substrate, and no evidence of unbounded simulation. The medium is a *ledger*, not a *computer* in the full universal-computation sense. This is important because contemporary distributed ledgers, Bitcoin, Ethereum, Cosmos, are *also* not Turing-complete in the relevant sense: Ethereum's EVM is Turing-complete *in principle* (with gas-bounded execution), but the substrate of its persistence (the chain of blocks) is a state machine with append, read, and (rarely) replace operations. The khipu is, on the criteria that matter for a ledger, *more* expressive than these in one specific dimension: granular erase. It is, by the comparison-of-substrates standard, the more capable substrate.

(ii) No claim of cryptographic verification under adversarial conditions. The khipu's signature is reputational, institutional, and (in the imperial period) legal. It is not cryptographic. Under conditions in which the *quipucamayoq* may be coerced, replaced, or forged, the substrate has no defence. This is the *single* primitive the khipu lacks, and it is precisely the primitive that post-quantum cryptography supplies in 2024. The synthesis we propose is exactly this: cord lineage substrate properties + post-quantum cryptographic signatures = a fit-for-twenty-first-century distributed ledger.

(iii) No claim of decipherment-completeness. We do not require, for the formal argument, that the entire khipu corpus be deciphered. Locke's numerical decipherment is consensual; Urton's binary structural argument is well-supported and influential; Hyland's logosyllabic argument is contested in detail but increasingly accepted in outline. Our argument depends only on the consensual core: position-addressable, typed, hierarchically aggregated, signed, replicated, granularly erasable. These properties are documented in the Aschers' corpus, in Salomon's ethnography, and in the chroniclers' accounts. They suffice.

4.4 The khipu in the comparative landscape of pre-modern computational substrates

To situate the proto-computer claim, it helps to compare the khipu against four other documented pre-modern computational substrates. The comparison is not invidious; each substrate is *fit-for-purpose* for

its own use case. The point is that the khipu's specific use case, distributed administrative ledgering with granular erasure, is the use case relevant to contemporary public-infrastructure design.

The abacus (multiple traditions: Mesopotamian, Greco-Roman, Chinese *suanpan*, Japanese *soroban*, Russian *schoty*; documented from at least 2700 BCE in Sumer): a position-addressable, typed, manual record on which arithmetic is performed by bead manipulation. Strengths: very fast manual arithmetic; very compact; very portable. Weaknesses: not *signed* (the abacus is anonymous); not *replicated* (each abacus is its own independent record); not *erasable in a tombstone-preserving way* (zeroing the abacus erases the structural memory of what was there). The abacus is a *computer* (in our restricted ledger sense), but not a *distributed* one and not a *granularly erasable* one. It is the local equivalent of a hand calculator.

The astrolabe (Hellenistic, c. 200 BCE, refined in the Islamic world from the eighth century CE): an analogue computational instrument with discrete reading positions encoding celestial geometry. Strengths: high precision in its domain; a beautiful realisation of analogue computation. Weaknesses: not *position-addressable* in a write sense (the astrolabe is read, not written); not *aggregating*; not *replicated* (each astrolabe is an independent instrument). The astrolabe is a *computational instrument*, but not a *ledger*.

The Antikythera mechanism (Greek, c. 100 BCE; Freeth et al., 2006): a geared analogue computer that calculated astronomical positions. The Antikythera is the most sophisticated pre-modern computational substrate yet recovered. Strengths: extraordinary mechanical sophistication; multiple coupled gear trains realising lunar and solar position calculations and eclipse predictions. Weaknesses: not *writable* (its "data" is encoded in its gear ratios, fixed at construction); not *signed*; not *aggregating* in our sense. The Antikythera is a *fixed-program computational instrument*. It is comparable to a modern read-only-memory (ROM) device: extraordinary, but not a ledger.

The counting-rod system of Han China (documented from at least the third century BCE; Needham, 1959): a position-addressable arithmetic system using small bamboo rods on a counting board. Strengths: high arithmetic capability, including operations on negative numbers (encoded by red versus black rods) and on fractional values; standardised across the Han bureaucratic system. Weaknesses: not persistent (the counting board is reset between operations); not signed; not replicated. The counting-rod system is a *bureaucratic arithmetic substrate*, comparable to the abacus but more flexible.

The wampum belt of the Haudenosaunee Confederacy (multiple Iroquoian and Algonquian traditions; Tehanetorens, 1999; Williams, 2018): a beaded belt encoding diplomatic agreements and historical events through bead position, colour, and pattern. Wampum belts are the closest pre-modern parallel to the khipu in *function*: they are signed (by the negotiating nations), replicated (multiple copies kept by participating parties), and ritually preserved as the substrate of inter-national agreements. Weaknesses, relative to the khipu: smaller scale (single agreements rather than continuous administrative records), no documented aggregation hierarchy at the imperial-cord level, less developed positional numeral encoding. The wampum belt is the closest cousin to the khipu in our taxonomy and merits its own architectural revival as a separate research programme.

The comparison places the khipu in a specific niche in the pre-modern computational landscape: the largest-scale, most temporally continuous, most administratively integrated, most hierarchically aggregating, *and* granularly erasable distributed information substrate that the pre-modern world is known to have produced. This is the lineage that Datachain Rope continues.

Part III. The Architecture of Datachain Rope

5. The Translation: From Khipu to Smartchain

5.1 The lexical substitution and what it means

Datachain Rope (mainnet chain ID 271828, named for the mathematical constant *e*, the base of the natural logarithm, the rate of continuous growth, and a deliberate signal that the network's value model is *organic* rather than *industrial*) is a public, permissionless, EVM-compatible smartchain. Its canonical specification is the *Quipu Primitive Canon*, a document authored and maintained by the Datachain Foundation that has, at the time of writing, been issued in three versions: v1.1 (knots, strings, tombstones; foundational), v1.2 (string registry, kind taxonomy, the strings/knots invariant), and v2.0 (scaling roadmap to five million knots per second; sharded lattice, parallel write batch, knot DAG). The Canon replaces the Bitcoin-Ethereum lexicon at the level of substrate.

The substitutions are these:

Bitcoin / Ethereum lexicon	Datachain Rope lexicon
Block	**Knot**
Block height / block number	**Knot height / knot index**
Genesis block	**Genesis knot**
Block reward	**Knot reward** (anchor emission)
Block time	**Knot interval** (∼3 seconds)
Block producer / miner	**Knot witness** (modern post-quantum *quipucamayoq*)
Finality	**Knot anchoring** at federation level
Chain (per-network)	**Cord** (the global federation-level structure)
(No equivalent)	**String** (the per-entity micro-ledger; new primitive)
Transaction erasure (no equivalent)	**rope_untieKnot** (granular erasure; new primitive)

The substitutions are not stylistic. Each maps to a structural design decision, and the decision in each case is grounded in a specific documented property of the cord lineage. We treat the substitutions one by one.

5.2 Block → Knot

In Bitcoin and Ethereum, a *block* is a container of fungible transactions submitted by mutually anonymous parties to a globally shared queue. Inclusion in a block is a property of the queue, not of the parties; the block's hash is the unit of integrity; once a block is sealed and propagated, its content is, on the dominant reading, immutable. This is a substrate-level commitment to *containerised, anonymous, immutable* records.

In Datachain Rope, a **knot** is the minimal information event tied at a specific position on a specific *string* (i.e. on a per-entity ledger; see §5.3). Each knot carries a structured payload:

- `event_id`, the deterministic, string-scoped position index of the knot (corresponding to the knot's *position* on the cord in the khipu);
- `event_hash`, a cryptographic commitment over the knot payload, anchored at federation level (corresponding to the structural commitment of the knot to the cord);
- `event_type`, drawn from a controlled vocabulary: `transfer | mint | burn | attestation | claim | upload | inference | erasure | anchor | identity_bind | compliance_update | rwa_mint | …` (corresponding to the *knot type* in the khipu typology);
- `event_size`, the payload magnitude, the modern analogue of the long-knot turn-count;
- `event_payload`, encrypted under the string's sovereign key material via the Organic Encryption System (OES; §8) with a per-knot key shred;
- `event_signature`, post-quantum, currently ML-DSA-65 (CRYSTALS-Dilithium 3) by default, with SLH-DSA (SPHINCS+) as a permitted fallback for hash-based-only deployments (NIST FIPS 204, FIPS 205, 2024).

The knot is *first-class*. It is not a slot in a block. It is the unit of state itself. Multiple knots may be aggregated into a higher-level *cord-anchor knot* (the Datachain Rope analogue of an Inka top cord), but the per-string knots persist as their own first-class records, individually addressable, individually signed, individually erasable. The substrate commits to *position-addressable, typed, signed, granularly-erasable* records, that is, to the operative properties of the khipu in §4.2, at the level of substrate, not at the level of application. This is the architectural realisation of the Andean property "position as identity" (§2.3): every knot is *located*; there is no free-floating knot in a global pool of records.

5.3 Chain → String + Cord

In Bitcoin and Ethereum, the *chain* is a single, shared, append-only log to which all transactions of all parties are written. There is exactly one chain per network. This is the substrate-level commitment to a single shared container.

In Datachain Rope, the substrate is bifurcated into two distinct primitives. Each ecosystem entity, every wallet, every contract, every real-world asset, every decentralised identity (ONCHAINID per ERC-734/735), every federation cord, owns a **string**: a per-entity, append-only sequence of knots. This is the *sovereign thread* of §3.1, instantiated as a first-class registry primitive. Strings are *typed*; the canonical kinds are five: `wallet` (a 20-byte EVM address held by an EOA), `contract` (a 20-byte EVM address held by smart-contract bytecode), `asset` (a hash derived from the contract address and token ID for an NFT, or from the contract address for an ERC-3643 security token), `did` (an ONCHAINID or Datawallet+ DID address), and `cord` (the special string with all-zero `id_bytes` belonging to the federation itself).

Above the strings sits the **cord**: the single global federation-level structure to which every per-entity string anchors at periodic intervals. The cord is the contemporary realisation of the imperial cord at Cuzco (§3.2). It does not contain the strings; it *aggregates* them. A cord-anchor knot is tied to the cord every approximately three seconds, and it commits, in a single cryptographic structure, to the knot summaries of every per-entity string that received a knot in the previous interval.

The substrate-level invariant, enforced by the protocol, is

$$|\,\text{strings}\,| \leq |\,\text{knots}\,|$$

because each string starts with a genesis knot, so the string count cannot exceed the knot count. This invariant is exposed in the JSON-RPC method `rope_globalStats`, which any consumer (DCScan, regulators, audit tools, ecosystem agents) can call to assert the architectural correctness of the network. At the time of writing, the production network reports approximately 1.1 million cord-anchor knots, 110 thousand transactions (which are knots of the EVM-compatible type, the subset whose payload is an EVM transaction), 50 thousand events (which are sub-transaction knots emitted as ERC-721/DCR-20/ERC-3643 logs), and a string registry that grows with each new wallet, asset, or DID introduced into the system. The relationship `cord–anchors` $\supseteq$ `transactions` $\supseteq$ `events`, by *layer* not by *count*, is explicitly stated in the Quipu Canon v1.2 (knot-event distinction).

This is the architectural realisation of *sovereign threading* (§2.3). No two entities share a string. Aggregation upward does not merge strings. The global federation cord is the *aggregation cord*, not "the chain". This single decision, to bifurcate the substrate into per-entity strings and a federated cord, rather than

to consolidate all records into a single shared chain, resolves the *block-mention paradox* (§7.1 below) at the level of substrate.

5.4 Genesis block → Genesis knot, Block height → Knot index

These are direct lexical substitutions and the EVM-compatibility layer preserves the Ethereum-tooling names as aliases. The canonical methods are `rope_knotIndex`, `rope_getKnotByIndex`, `rope_getKnotByHash`, and the corresponding aliases `eth_blockNumber`, `eth_getBlockByNumber`, `eth_getBlockByHash` continue to work indefinitely. This is a deliberate decision: wire compatibility with the entire Ethereum tooling ecosystem (MetaMask, ethers.js, hardhat, foundry, viem, ABI-encoded interactions, indexers like The Graph, oracles like Chainlink) is preserved without compromise. New SDK code calls `rope_*`. Old SDK code calls `eth_*`. Both work.

The strategic significance: the cord lineage does not require its users to abandon the dominant tooling ecosystem to access cord-lineage substrate properties. A wallet that supports Ethereum supports Datachain Rope without modification. A smart contract written in Solidity deploys to Datachain Rope without modification. The cord-lineage primitives, `rope_appendToString`, `rope_walkString`, `rope_untieKnot`, `rope_globalStats`, `rope_listStrings`, are *additional* surface, not *replacement* surface. The user who needs only EVM compatibility never needs to learn the cord vocabulary. The user who needs sovereign threading or granular erasure has it, on the same substrate, in the same chain, on the same wallet.

5.5 Miner / block producer → Knot witness; "mining" → Anchor emission

We come to a substitution that carries some of the heaviest political weight. In Bitcoin, *miners* solve a proof-of-work puzzle to claim the right to produce the next block, and are rewarded in a fungible mineral metaphor: 50, then 25, then 12.5, … BTC per block, halved approximately every four years. The metaphor of *mining* commits the substrate to an extractive, energy-consumptive, mineralogical conception of value creation. In Ethereum (post-Merge, since September 2022), *validators* stake ETH and are pseudo-randomly selected as proposers; the metaphor shifts from extraction to deposit-based selection, but the *fungible-block* substrate remains.

In Datachain Rope, the analogue is the **knot witness**, the modern post-quantum *quipucamayoq*. Witnesses do not "mine" in the proof-of-work sense; they participate in the *Testimony consensus* protocol (§7 below), in which witnesses tied to specific strings testify to the validity of state transitions on those strings, and aggregate testimonies are bundled into cord-anchor knots emitted at approximately 4.2-second intervals. New DC FAT (the network's native fungible token; standard DCR-20) is created by **anchor emission**, the modern analogue of the Inka tribute aggregation upward to the imperial cord. The distribution is institutional, not extractive: proposer 30%, testimony pool 45% (validators across the network), node operators 20%, federation/community 5%. A Bitcoin-style four-year halving schedule operates: Era 1 = 500 million FAT/year (2026–2029), then 250M, 125M, …, asymptotically capped at approximately 18 billion FAT, with a 1M FAT/year emission floor. The genesis supply is 10 billion FAT, distributed at mainnet launch among the founding allocations (Datachain Foundation, ecosystem fund, Green Fund, community treasury, validator initial stakes, pre-genesis credit-holders).

The substitution of "miner" by "witness" is consequential and we name it in plain language. It discards the mineralogical metaphor (extraction of value from a finite physical substrate) in favour of the institutional metaphor (testimony to events on a sovereign thread). In anthropological terms, this is the difference between *mining a commodity* and *being a witness in court*. The two are not equivalent positions in any society, the miner extracts value from the earth; the witness vouches for an event. The *quipucamayoq* was a witness, not a miner. The Datachain Rope knot witness is a witness, not a miner. The token emission that rewards the witness is the institutional analogue of the *quipucamayoq*'s pension: a livelihood for a person whose labour is the maintenance of the substrate of state. This is a different political economy than mining, and we will return to it in §11.

5.6 No-equivalent operations: the cord-lineage additions

Some operations on Datachain Rope have no Ethereum-tooling equivalent because they realise primitives that the block lineage does not have. We name them explicitly:

- `rope_appendToString(kind, id_bytes, payload)`, append a knot to a per-entity string. There is no `eth_*` equivalent because Ethereum's substrate has no per-entity strings. This is the contemporary realisation of the act of *tying a knot on the cord-keeper's own cord*.
- `rope_walkString(kind, id, offset, limit)`, read knots on a per-entity string. There is no `eth_*` equivalent because Ethereum's substrate has no per-entity strings to walk.
- `rope_untieKnot(string_id, event_id)`, untie a knot, performing granular erasure with tombstone preservation (§6). There is no `eth_*` equivalent because Ethereum's substrate is append-only by construction.
- `rope_globalStats()`, return per-kind string and knot counts and assert the substrate-level invariant. There is no `eth_*` equivalent because Ethereum's substrate has no strings to count.
- `rope_listStrings(kind, offset, limit)`, list strings of a given kind. There is no `eth_*` equivalent.

These five operations are the operational expression of the cord lineage's substrate-level superiority over the block lineage. They are the operations that the cord cleverly admits and the block cleverly excludes, and it is on these operations that the next two decades of public-infrastructure deployment will turn.

6. The Granular-Erasure Primitive

6.1 The operation

Datachain Rope's most consequential primitive, the operation we have flagged repeatedly and now treat in full detail, is `rope_untieKnot(string_id, event_id)`. The operation accepts two parameters: the identifier of a sovereign string (the wallet, contract, asset, or DID whose ledger contains the knot), and the position-index of the specific knot to be untied. The operation is authorised either by the string's owner (with a post-quantum signature proving control of the string's key material) or, in specific regulatory-compliance circumstances, by an authorised judicial or data-protection authority operating under a documented legal basis. On successful authorisation, the operation atomically performs the following five sub-operations:

(i) Cryptographic destruction of the payload. The encrypted payload key for the targeted knot is shredded under the Organic Encryption System (OES; §8). The OES architecture guarantees that, post-shredding, the underlying plaintext payload of the knot cannot be reconstructed by any party, including the Datachain Foundation, the consensus validators, the string's owner, the data-protection authority, or any future possessor of the post-shredding state of the network. This is *cryptographic erasure* in the strict sense analysed by the Article 29 Data Protection Working Party in its *Opinion 05/2014 on Anonymisation Techniques* (WP29, 2014), namely, the data is rendered permanently unreadable by the destruction of the only key that could decrypt it, and the destruction is unilateral, irreversible, and globally simultaneous.

(ii) Tombstone preservation. The position of the untied knot on the string is preserved as a *tombstone knot*, a structurally present absence. The tombstone records: the timestamp of the untying, the authorisation proof (the post-quantum signature of the authorising party), the legal basis (a controlled vocabulary entry, e.g. `gdpr_article_17`, `ccpa_1798_105`, `lgpd_article_18`, `judicial_order_<jurisdiction>`, `voluntary_redaction`), and the original `event_type` of the knot (so that the audit log knows what *kind* of event was erased, even though the content is gone). The tombstone is itself a knot, of `event_type = erasure`, signed by the authorising party. It cannot be itself untied, tombstones are themselves immutable, by design.

(iii) String-integrity preservation. The hash continuity of the string is preserved. The position indices of all subsequent knots on the same string are unaffected. All unrelated attestations, deeds, credentials, balances, claims, and events on the same string remain readable, valid, and signed. This is the architectural realisation of the cord-lineage property that the cord remains continuous when an individual knot is untied (§3.3). The cryptographic construction by which this property is realised is given formally in §6.1.1 below.

(iv) Federation anchoring of the erasure. The untying is itself recorded as a first-class auditable event on the federation cord. The cord-anchor knot at the next anchoring interval (within ~3 seconds) commits, by post-quantum signature, to the fact that an erasure occurred on string `s` at position `i` at time `t` under authorisation `a`. The audit trail of the erasure survives indefinitely. The content of the erased knot does not.

(v) Irreversibility. The operation is irreversible by construction. No party, the Foundation, the validators, the user, the data-protection authority, a future court order, can recover the erased payload, because no party retains the key shred that was destroyed. This is the architectural realisation of the *trust-by-physics* property that the cord lineage achieves through the physical destruction of the knot in cotton: in the cryptographic regime, the analogue is the destruction of the per-knot key shred under OES.

6.1.1 The hash-chain-with-tombstone construction

The string-integrity property of §6.1(iii) is asserted in the Quipu Primitive Canon v1.1 (Datachain Foundation, 2026a, §5(3)) as the substrate-level guarantee that the string's hash continuity, the position indices of all subsequent knots, and all unrelated knots on the same string are unaffected by an untying. Canon v1.1 specifies the property; it does not specify the cryptographic construction by which the property is realised. We supply that construction here, since it is the construction that distinguishes, in formally checkable terms, the cord lineage's chain-continuity property from the block lineage's chain-immutability property, and since it is the construction that the forthcoming Canon revision will fix as the normative specification on which `rope_untieKnot` rests.

Each knot k_i on a string s carries a per-knot hash that is computed not over the encrypted `event_payload` but over a tuple of *erasure-survivable* fields:

$$h_i = H(\text{event\_id}_i \,\|\, \text{event\_type}_i \,\|\, \text{event\_metadata\_hash}_i \,\|\, \text{authorisation\_proof}_i \,\|\, h_{i-1})$$

where H is BLAKE3 (the hash function used throughout the production deployment for string identifiers and for the OES key-derivation cascade; see §11.1), $\|$ denotes concatenation, h_0 is the genesis-knot hash committed at string registration, and `event_metadata_hash` is itself a commitment over the knot's structural metadata (timestamp, witness identifiers, testimony-pool quorum, OES key-shred destination set) but *not* over the plaintext payload. The encrypted `event_payload` is committed separately under the OES per-knot ephemeral key and is not in the hash-chain pre-image. The `authorisation_proof` is the post-quantum signature of the party whose private key authorised the knot (the string's owner in the ordinary case, a delegated authority for compliance-mandated tombstones); committing to the authorisation in the chain pre-image is what allows a regulator verifying the chain post-erasure to confirm not only *what* was tied at position i but *who* authorised the tying.

The construction has three consequences that bear directly on §6.1(iii).

First, when `rope_untieKnot` is invoked on knot k_i, the OES key shreds for k_i are destroyed (sub-operation (i)) and the encrypted payload becomes mathematically irrecoverable. The fields entering h_i, however, are unchanged: `event_id`, `event_type`, `event_metadata_hash`, `authorisation_proof`, and h_{i-1} all survive the erasure. The tombstone knot's `event_type = erasure` is recorded in the tombstone's metadata, and the original event-type is preserved per §6.1(ii); the chain continues to verify because the inputs to H are unchanged.

Second, all subsequent knots $k_{i+1}, k_{i+2}, \ldots$ on the same string carry hashes computed over h_i as their predecessor input. Because h_i is unchanged, $h_{i+1}, h_{i+2}, \ldots$ are unchanged. No re-hashing of the tail of the string is required, and the cord-anchor commitments of every prior interval that committed over $h_{i+1}, h_{i+2}, \ldots$ remain valid commitments over those same hashes. The string is not re-woven; only the payload of k_i is destroyed.

Third, the construction is observably distinct from the Bitcoin/Ethereum block-hash construction and from the construction that would be required if the encrypted payload were in the hash-chain pre-image. In the latter case, destruction of the payload key would not affect the encrypted ciphertext (which remains in the chain) but would render the chain unverifiable for any party that needed to recompute h_i from the plaintext, since the plaintext is no longer recoverable. The cord-lineage construction sidesteps this difficulty by chaining over a commitment to the metadata only, with the encrypted payload bound to the knot through a separate OES wrapping that is independently verifiable from the metadata commitment but is not on the integrity-critical hash path.

The construction is, formally, a separation of two cryptographic commitments per knot: a *durability commitment* (the hash h_i over erasure-survivable fields) which guarantees chain continuity under arbitrary subsequent erasures, and a *confidentiality commitment* (the OES wrapping of the payload under the per-knot ephemeral key) which guarantees that the payload is recoverable if and only if the key shreds survive. The two commitments are bound together by the inclusion of `event_metadata_hash` in h_i, which itself commits to the OES key-shred destination set, so that the audit trail can verify post hoc *which* witnesses held shreds for the erased knot and *whether* their destruction obligations were honoured at the cord-anchor commit (§6.1(iv)).

The construction distinguishes the cord lineage's chain-continuity property from the block lineage's chain-immutability property in formally checkable terms: the cord lineage is *continuous* under granular erasure, where the block lineage is *immutable* against any erasure. The two are not the same property. The

Andean cord-keepers, in their physical practice of untying without re-weaving the cord on which a discredited knot had been tied, were already operating the continuous variant; the construction specified above is the cryptographic realisation of that physical practice on a post-quantum public smartchain.

6.2 The legal positioning

`rope_untieKnot` is, on the architectural argument we make, the substrate-level realisation of what the General Data Protection Regulation (Regulation (EU) 2016/679, Article 17) calls "the right to erasure" or, less formally, "the right to be forgotten". Article 17 grants every data subject in the European Union the right to obtain, from a data controller, the erasure of personal data concerning them, on specified grounds (no longer necessary, withdrawal of consent, unlawful processing, compliance with a legal obligation, etc.). Equivalent rights are granted under the California Consumer Privacy Act §1798.105, the Brazilian *Lei Geral de Proteção de Dados* Article 18, the Chinese *Personal Information Protection Law* Article 47, the South African *Protection of Personal Information Act* §24, the EU AI Act Article 10 (data quality and erasure for high-risk AI systems), and the EU Digital Operational Resilience Act for financial-sector data.

The architectural difficulty with traditional public blockchains, exhaustively documented by Finck (2018), the European Parliamentary Research Service (EPRS, 2019), and the EDPB, is that an append-only block-chained data structure is *constitutionally incompatible* with these rights. To erase a record from a public blockchain in the strict sense requires either (a) rewriting all subsequent blocks (which on a network with active validators and economic security is computationally and economically infeasible), or (b) storing personal data off-chain in a side database whose entries are referenced from the chain by hash, and then deleting the side database (which voids the on-chain integrity claim, because the on-chain record is now a hash with no readable referent).

The relevant European data-protection authorities have not issued a guideline specifically on cryptographic erasure as a sufficient measure under Article 17 GDPR. The two adjacent authorities are: (i) the EDPB Guidelines 04/2019 on Article 25 (data protection by design and by default; version 2.0, October 2020), which treats cryptographic protection and erasure-by-default as design-stage measures consistent with the by-default principle but does not adjudicate the sufficiency of cryptographic erasure as Article 17 compliance; and (ii) the Article 29 Working Party's *Opinion 05/2014 on Anonymisation Techniques* (WP29, 2014), which analyses key-destruction as a candidate anonymisation technique and identifies the conditions under which it can be considered effective (irreversibility of the destruction, robustness of the underlying scheme against currently-foreseeable attack, exclusion of all parties from access to the destroyed key material). The architectural regime that `rope_untieKnot` realises is the regime that satisfies the WP29 conditions: the destruction of the OES key shred is unilateral (the protocol enforces it), irreversible (no party retains a copy), globally simultaneous (the cord-anchor commits to the erasure within 3 seconds), and robust against post-quantum attack (because the underlying cryptography is ML-DSA-65 and SLH-DSA, both NIST-standardised in 2024).

We are deliberately careful with the legal claim here. The Court of Justice of the European Union has not, at the time of writing, issued a definitive ruling on the sufficiency of substrate-level cryptographic erasure for Article 17 compliance, and individual data-protection authorities have expressed reservations about blockchain-native erasure mechanisms in their national guidance (notably the German Federal Commissioner for Data Protection and Freedom of Information (BfDI) and the French Commission Nationale de l'Informatique et des Libertés (CNIL), each in published positions on blockchain). Our claim is therefore the architectural one: that `rope_untieKnot` provides the strongest substrate-level realisation of Article 17 currently engineerable on a public smartchain, that the realisation satisfies the conditions that WP29 (2014) identifies as the technical criteria for effective key-destruction, and that the realisation is materially closer to compliance than any application-layer workaround on a block-chained ledger. Whether the realisation will be judicially confirmed as sufficient is an open question, and we treat it as such throughout. The architecture is a strong proposition without overselling its current legal status.

The architectural claim is, in plain language, this: Datachain Rope is the only public smartchain on which the right to be forgotten is *substrate-native*. On Bitcoin, on Ethereum, on every Cosmos zone and every Polkadot parachain, the right to be forgotten is at best a workaround; on Datachain Rope, it is a primitive. This is a regulatory positioning, not yet a regulatory ruling. Competing architectures cannot offer the property without rebuilding from the substrate up, and we expect the first definitive jurisdictional ruling on substrate-level cryptographic erasure to be a significant inflection point for the field.

6.3 The financial-reversal disambiguation

A clarification is necessary because the operation is sometimes misread. `rope_untieKnot` performs *cryptographic erasure of the personal-data payload*. It does *not* reverse a financial transfer. If a knot of type `transfer` records a movement of DC FAT or of an ERC-3643 security token, untying that knot cryptographically erases the *payload* of the knot (any personal-data fields, any memo, any documentation reference) but does not unwind the underlying balance change. Financial reversal on Datachain Rope is, as on every other smartchain, performed by an explicit reversal transaction (a counter-transfer), authorised by the parties and recorded as its own knot. The two operations are kept categorically separate by design: `rope_untieKnot` is a data-protection primitive; financial reversal is a contractual primitive. This separation is consistent with the EDPB's guidance that erasure does not extinguish lawful contractual obligations, only the personal-data dimensions of their record.

7. Testimony Consensus and the Federation Generation Protocol

7.1 From the *quipucamayoq* to the witness

The Inka *quipucamayoq* was, on the chroniclers' accounts, a *witness*: a professionally trained, reputationally bound, institutionally authorised person who *vouched for* the events recorded on the cords under his keeping. The vocabulary of *witness* is institutional and forensic: a witness testifies under oath to what occurred; the testimony is the substrate of the record; the witness's reliability is the substrate of the testimony.

Datachain Rope's consensus protocol, which the Canon names *Testimony consensus*, adopts this institutional vocabulary deliberately and substantively. Witnesses are individual node operators, registered to the network, who run a *rope-node* validator and stake DC FAT to participate. Witnesses are *typed*: each witness specialises in a category of testimony (drawn from a controlled vocabulary that mirrors the *event_type* taxonomy: `transfer-witness`, `attestation-witness`, `claim-witness`, `oracle-witness`, `compliance-witness`, `inference-witness`, `green-witness`, `rwa-witness`, etc.), and a witness's reputational stake is per-category, accruing over time on the basis of the witness's track record of accurate testimony in that category.

The protocol works as follows. When a knot is to be tied to a string, say, a `transfer` knot on a wallet's string, the proposing party broadcasts the knot to the witness pool. Witnesses subscribed to the relevant category examine the knot, verify its post-quantum signature, verify its compliance with the string's history (the new knot's *event_id* is the next position index; the new knot's *event_hash* commits to the string's prior state), and emit *testimonies*: signed attestations that the knot is valid. The cord-anchor knot at the next anchoring interval bundles the knot together with a quorum of its testimonies and commits the bundle, with a post-quantum signature, to the federation cord. Finality is achieved at the cord-anchoring step.

The economic incentives are aligned with the institutional metaphor. Witnesses earn knot rewards proportional to their testimony share (45% of the knot reward goes to the testimony pool, distributed by accurate-testimony share). Witnesses lose stake (slashed) for emitting testimonies that contradict the consensus or that vouch for invalid knots. Reputational scoring is per-category, so a witness who is reliable on `transfer` knots but unreliable on `oracle` knots has their stake-weight in the latter category reduced. This is the institutional analogue of the *quipucamayoq* lineage: the witness who has cared for the tribute records of a province for thirty years has an authority on tribute records that a new witness does not, and the protocol respects this through reputational scoring.

7.2 The Federation Generation Protocol

Above the level of individual knots and individual cord-anchor knots sits the *Federation Generation Protocol*: the architectural realisation of the imperial-cord pattern (§3.2). The protocol specifies how, periodically (currently every 86,400 cord-anchor intervals, approximately once per day), the network generates a *generation cord*, a cord whose entries summarise the previous generation's federation cord at a higher level of aggregation. Generation cords are themselves anchored at higher-frequency intervals (weekly, monthly, annually), producing a recursive aggregation hierarchy that mirrors the *ayllu* → *kuraka* → *tukuyrikuq* → *suyu* → *imperial cord* hierarchy of the Inka administrative state.

The strategic significance is twofold. First, the Federation Generation Protocol allows the substrate to scale: a witness or auditor who needs to verify the state of the network at a coarse granularity does not need to walk every knot; they can walk the generation cords at the appropriate granularity and verify the aggregation relations down to the level they need. This is the contemporary realisation of the audit-by-recitation procedure that Polo de Ondegardo (1571) describes for the Inka administration. Second, the Federation Generation Protocol allows the substrate to *federate* with other federations: a future Datachain Rope deployment in another jurisdiction (say, an African Union deployment under the AU Data Policy Framework, or a Pacific Island Forum deployment under the Pacific Data Sovereignty principles) can anchor its own generation cords into a *meta-federation cord*, producing a peer-to-peer federation of public ledgers that respects regional sovereignty while admitting cross-federation audit. This is the architecture of a *post-Westphalian* public-data infrastructure, in which sovereignty is preserved at the regional level and aggregation is performed across regional levels by mutual consent.

We do not detail the Federation Generation Protocol further here; the full specification is in Datachain Foundation (2026d), the Datachain Rope Technical Whitepaper v4. The point we mark is that the protocol is the architectural realisation of the imperial-cord pattern, that it is designed for cross-federation interoperability, and that it is the substrate-level answer to the question of how to administer global public-data infrastructure without imposing a single hegemonic chain on all jurisdictions.

8. The Organic Encryption System (OES)

8.1 The cryptographic substrate

The granular-erasure primitive (§6) depends on a cryptographic substrate that supplies four properties simultaneously: (a) per-knot encryption with per-knot keys; (b) post-quantum security against currently foreseeable attack; (c) unilateral, irreversible key destruction on demand; (d) integration with the per-string sovereign key material so that erasure authorisation is bound to string-ownership proof. Datachain Rope's solution is the Organic Encryption System (OES), a layered cryptographic architecture whose name signals its design principle: keys, like organic structures, are *born*, *grow*, and *die*, with the death being irreversible and the lineage being recorded.

The architecture has three layers:

Layer 1, String key material. Each per-entity string is bound to a post-quantum key pair (ML-DSA-65 default, SLH-DSA fallback) held by the string's owner. The public key is registered in the string registry at genesis. The private key is held off-chain by the owner (in a Datawallet+ instance, in a hardware security module, in an institutional custodial arrangement, etc.). The string's key material authorises all writes and erasures on the string.

Layer 2, Per-knot ephemeral keys. Each knot tied on a string is encrypted under a per-knot ephemeral symmetric key (256-bit ChaCha20 by default, with AES-256-GCM as a permitted alternative for jurisdictions with FIPS compliance requirements). The ephemeral key is wrapped under the string's public key and stored alongside the encrypted payload in the knot's `event_payload` field. The ephemeral key is *itself* split, using a threshold secret-sharing scheme (Shamir, 1979, with post-quantum-safe parameters), into a number of *key shreds* distributed across the testimony pool and into a custodial shred held by the string's owner.

Layer 3, Erasure-by-shredding. When `rope_untieKnot` is invoked, the protocol orders the destruction of *all* key shreds for the targeted knot. The string's owner destroys their custodial shred. The witnesses in the testimony pool destroy their shreds. The cord-anchor knot commits, by post-quantum signature, to the fact of the destruction. After the cord-anchor commit, no party retains any shred of the per-knot ephemeral key, and the encrypted payload is therefore mathematically un-decryptable. The data is gone.

8.2 Why "organic"

The naming is deliberate. The system is *organic* in the sense that keys are not eternal; they are time-bounded, lineage-bound, and subject to death. This is a different cryptographic philosophy than the dominant public-blockchain philosophy, which treats keys as eternal (a Bitcoin private key, once created, is, in principle, valid forever), and treats the erasure of data as a regulatory afterthought to be handled at the application layer. OES treats keys as *biographical*: they belong to a string, they encrypt the string's history, and they can be retired *for specific knots* without compromising the string's ongoing operation. The string's key material persists; the per-knot keys do not, when erasure is invoked.

This philosophy is, on the cord-lineage reading, the cryptographic realisation of the Andean conception of the cord as a biographical artefact: the cord-keeper's cord records the cord-keeper's life of work, and individual knots can be untied as that life evolves, without compromising the cord's continuity. The cord

is the keeper's biography; the knots are its events; the erasure is the keeper's editorial. The OES is the cryptographic version of this editorial discipline.

8.3 Cryptographic security argument

The security of OES rests on three independent cryptographic primitives, each of which has been independently analysed and standardised:

(a) **ML-DSA-65 (CRYSTALS-Dilithium 3)** for string signatures and knot signatures. Standardised by NIST in FIPS 204 (August 2024) on the basis of more than five years of public cryptanalysis under the NIST Post-Quantum Cryptography standardisation process. Security level: NIST level 3 (equivalent to AES-192). Resistant to known classical and quantum attacks under the lattice-problem hardness assumption. (Ducas et al., 2018; NIST, 2024a.)

(b) **SLH-DSA (SPHINCS+)** as a hash-based-only fallback for jurisdictions that prefer not to rely on lattice assumptions. Standardised by NIST in FIPS 205 (August 2024). Security rests only on the security of the underlying hash function (SHA-3 / SHAKE), which is a more conservative cryptographic foundation. (Bernstein et al., 2019; NIST, 2024b.)

(c) **Threshold secret sharing** (Shamir's scheme over a finite field of post-quantum-safe size, with a quorum threshold tuned per knot, typically *t-of-n* with $t \approx 0.6n$) for the per-knot ephemeral key shredding. Information-theoretically secure: even an adversary with unbounded computational power, in possession of fewer than t shreds, has no information about the encrypted payload. (Shamir, 1979.)

The composition is that an attacker who wishes to recover an erased payload must either (i) break the post-quantum signature scheme (which on currently foreseeable cryptography requires fault-finding in the NIST-standardised algorithms), or (ii) recover the destroyed key shreds (which is information-theoretically impossible if the destruction is honest), or (iii) have intercepted the payload before its encryption (which is a pre-erasure attack and outside the scope of the erasure guarantee). For ordinary GDPR compliance, the erasure guarantee covers the post-erasure threat model, that is, it guarantees that the data cannot be recovered *after* the erasure operation, and this is the guarantee that WP29 (2014) identifies as the technical condition for effective key-destruction-based erasure.

9. Comparative Architecture

We tabulate, for clarity, the architectural choices of Datachain Rope against the principal contemporary public-ledger families. The comparison is meant to be deflationary: it is not a marketing comparison, it is an architectural one.

Property	Bitcoin	Ethereum (post-Merge)	Cosmos / Polkadot	Holochain	**Datachain Rope**
Unit of state	Block (transactions)	Block (txs, accounts, contracts)	Block (per-zone)	Source-chain entry (per-agent)	**Knot (per-string)**
Container model	Single shared chain	Single shared chain	Federated chains via IBC/XCM	Agent-centric source chains + DHT	**Federated cord aggregating sovereign strings**
Per-entity ledger	No	No	Zone-level only	Yes (per-agent)	**Yes (string registry, 5 kinds)**
Global ordering	Yes (one chain)	Yes (one chain)	Per-zone	No (eventual via DHT)	**Yes (cord-anchor every ∼3s)**
Granular erasure	No	No	No	Partial (DHT GC)	**Yes (`rope_untieKnot`)**
Tombstone preservation on erasure	n/a	n/a	n/a	No	**Yes**
GDPR Art. 17 architectural fitness	Poor (workaround)	Poor (workaround)	Poor (workaround)	Mixed (no global ledger; DHT residue persists)	**Native substrate-level**
Consensus	Proof-of-work	Proof-of-stake	Tendermint / GRANDPA-BABE	Validating-DHT (no global consensus)	**Testimony consensus** (PQ-signed, per-category)
Signature primitive	ECDSA secp256k1	ECDSA secp256k1 + BLS	Ed25519 + BLS	Ed25519	**ML-DSA-65 / SLH-DSA**
Post-quantum readiness	No	Roadmap	Roadmap	No (Ed25519)	**Native (NIST FIPS 204/205)**
EVM tooling compat	No	Yes	Partial (EVM zones)	No	**Yes (full alias layer)**
Native fungible standard	n/a	ERC-20	per-zone	per-hApp	**DCR-20** (ERC-20 wire-compat)

Property	Bitcoin	Ethereum (post-Merge)	Cosmos / Polkadot	Holochain	**Datachain Rope**
NFT standard	n/a	ERC-721	per-zone	per-hApp	**ERC-721 (preserved name)**
Compliance / security tokens	No	ERC-3643 ecosystem	per-zone	per-hApp	**ERC-3643 + ONCHAINID**
On-substrate aggregation primitive	No	No (only via contracts)	No	No	**Yes (Federation Generation Protocol)**
Cross-jurisdiction federation	n/a	n/a	IBC	Bridge protocol per-hApp	**Generation-cord federation**
Token economics	Fixed cap 21M, halving	Variable (EIP-1559 burn)	per-zone	per-hApp	**10B genesis, ∼18B asymptotic, halving with 1M floor**

The comparison surfaces five points worth naming explicitly.

First, on every interoperability dimension, Datachain Rope preserves the dominant standards' names and wire formats. The `eth_*` aliases are permanent. ERC-721 is preserved by name (because ERC-721 is the *NFT* standard on Datachain Rope, and the name is industry-stable). ERC-3643 is preserved by name (because ERC-3643 is the *T-REX* security-token standard, and the name is industry-stable). EIP-2612 (permit signatures) is preserved by name. ERC-165 (interface detection) is preserved by name. ERC-734/735 (ONCHAINID identity claims) are preserved by name. The fork is on the *fungible-token* axis only, where DCR-20 replaces ERC-20 as the canonical name *because* DCR-20 is the native fungible standard of this network and not an Ethereum import, but the byte-level ABI is identical and a contract-author can deploy an Ethereum ERC-20 ABI to Datachain Rope without modification. This is the standard pattern of a network that takes its independence seriously while remaining wire-compatible with the dominant ecosystem: it asserts substrate independence without imposing tooling friction.

Second, on the *granular erasure* axis, Datachain Rope is, to our knowledge, alone among public smartchains. This is a regulatory moat (the property cannot be retro-fitted to a block-chained substrate without rebuilding) and an ontological claim simultaneously (the substrate commits to the cord-lineage proposition that records are accountable, not eternal).

Third, on the *post-quantum* axis, the choice of ML-DSA-65 and SLH-DSA places Datachain Rope ahead of the production-deployed PoW and PoS chains. The economic life of personal records is measured in decades; the cryptographic horizon for ECDSA against quantum attack is, on the most credible public estimates, also measured in decades (Mosca, 2018; Bernstein and Lange, 2017). For a chain whose central use case is sovereign personal-data records, identity claims, biodiversity attestations, real-world asset deeds, cultural-heritage receipts, agricultural provenance, medical data, educational credentials, classical-cryptography signatures are an unacceptable long-horizon risk. Bitcoin and Ethereum will, on their published roadmaps, eventually migrate to post-quantum cryptography (Buterin, 2024). They have not done so yet. Datachain Rope does so at genesis.

Fourth, on the *consensus* axis, Testimony differs from PoW (Bitcoin) and PoS (Ethereum, Cosmos) in a structurally important way: it scores witnesses *per category*, not in aggregate. A witness who is reliable on `transfer` knots has high stake-weight on `transfer` knots and may have lower stake-weight on `oracle` knots until they accumulate testimony history in that category. This is the architectural realisation of the cord-keeper specialisation pattern (§3.1): the *quipucamayoq* who cared for tribute records had authority on tribute records; the *quipucamayoq* who cared for ritual-calendrical records had authority on ritual-calendrical records; expertise was per-domain. Per-category reputational scoring resists a class of attacks that single-stake PoS systems are vulnerable to: an attacker who has acquired a majority stake of ETH can attack any Ethereum operation; an attacker who has acquired a majority stake on `transfer` testimonies on Datachain Rope cannot, by that fact, attack `oracle` testimonies, because the latter requires `oracle`-category reputation accumulated over time.

Fifth, on the *federation* axis, Datachain Rope's Generation Cord pattern admits a cross-jurisdictional federation that is structurally distinct from Cosmos's IBC and Polkadot's XCM. IBC and XCM are *peer-to-peer* relays between sibling zones, with no native aggregation hierarchy. The Generation Cord pattern produces a *recursive* aggregation hierarchy that mirrors the *ayllu → kuraka → suyu → imperial cord* pattern. This is the more apt structure for a global public-data infrastructure that needs to aggregate across nested administrative levels (region, sub-region, nation-state, supranational treaty body) while preserving sovereignty at each level.

A note on Holochain (Harris-Braun, Brock, and others, since 2018) is warranted because it is the public-ledger family closest to Datachain Rope on the per-entity-ledger axis and the comparison is therefore the most architecturally informative. Holochain shares with Datachain Rope the commitment to per-agent (per-entity, in our vocabulary) source chains as the unit of state, and rejects the single-shared-chain model of Bitcoin and Ethereum. The architectures diverge on three substantial points that bear directly on the use-case fit. First, Holochain has no global ledger and no global ordering: validation is performed by a distributed hash table (DHT) of validators sampled per entry, and inter-agent ordering is eventually consistent rather than globally serialised. Datachain Rope retains a global ordering through the cord-anchor knot at approximately three-second intervals; this is what allows the Federation Generation Protocol (§7.2) to commit recursive aggregations in the *ayllu → kuraka → imperial cord* pattern, which is structurally infeasible on Holochain's DHT-only architecture. Second, Holochain provides only partial granular erasure: an agent can stop publishing entries to the DHT, but DHT-resident copies of past entries persist on validators' nodes until garbage-collected, and there is no substrate-level tombstone primitive equivalent to `rope_untieKnot` with a federation-anchored erasure record. Third, Holochain's signature primitive is Ed25519, which is not post-quantum-resistant. The architectures are complementary in their target use cases: Holochain is well-fitted to applications where per-agent sovereignty without global ordering is acceptable (peer-to-peer messaging, hApp-internal coordination); Datachain Rope is fitted to applications where per-entity sovereignty *with* global ordering, granular erasure, and post-quantum signatures is required (identity, real-world asset deeds, biodiversity attestations, regulated-jurisdiction infrastructure). The cord lineage and the source-chain lineage are both responses to the unsuitability of the block lineage for non-fungible records, and both are valuable; they answer different sub-classes of the non-fungible-records problem.

10. Implementation: The Rust Codebase and the Bifurcated Execution Stack

The argument so far has been substrate-level. We now ground it in implementation. Datachain Rope is built in Rust as a workspace of seven core crates plus an EVM execution backend (Reth v1.11.2). The codebase is approximately 95,000 lines of Rust across the workspace, plus the inherited Reth tree, with a deliberate separation between the consensus layer (rope-native, Quipu Canon v1.2 / v2.0 primitives) and the EVM compatibility layer (Reth, EVM-tooling-compatible). This separation is itself an architectural realisation of the cord-lineage commitment: the cord-native primitives live in the rope crates, the EVM-compatibility surface lives in Reth, and the two are joined by an alias bridge that exposes every `rope_*` JSON-RPC method together with its `eth_*` mirror.

10.1 Crate layout

The workspace is organised under `crates/` with the following modules. The naming, scope, and inter-crate dependencies are stable across the v2.0 generation of the codebase.

Crate	Lines (approx.)	Role
`rope-core`	12,400	The substrate primitives: `Knot`, `String`, `StringRegistry`, `LamportClock`, `StringLattice`, `LedgerDescriptor`, the `untie_knot` operation, the OES key-shred state machine.
`rope-crypto`	9,800	Post-quantum signature primitives (ML-DSA-65 via the `pqcrypto-dilithium` crate, SLH-DSA via `pqcrypto-sphincsplus`), the OES key derivation (the `derive_key` BLAKE3 cascade, currently 100–199 rounds per call uncached), threshold secret-sharing (Shamir over GF(2^256)), and the per-knot AEAD wrappers (ChaCha20-Poly1305 and AES-256-GCM).

Crate	Lines (approx.)	Role
`rope-node`	18,200	The networked validator process: libp2p P2P stack, Testimony consensus orchestrator, RPC server (axum + tower), ledger manager (`LedgerManager::list_strings / get_string / global_stats`), bridge to Reth via the engine API.
`rope-smartchain`	14,600	The smart-contract execution and state management for cord-native operations: deployment of DCNFT (ERC-721) and ERC-3643 contracts, the `digital_credits` module that issues ONCHAINID-bound credentials, the cord-anchor batch construction.
`rope-bridge`	11,300	The DCR-20 transfer event log over knots, the bridge from Reth EVM events to cord-anchor knot summaries, the per-knot OES wrapping for transfer payloads.
`rope-economics`	4,500	The DC FAT emission schedule (`EmissionSchedule`, `asymptotic_max_supply`, the four-year halving table), the per-anchor reward distribution (proposer 30 percent, testimony pool 45 percent, node operators 20 percent, federation/community 5 percent), and the staking accounting.
`rope-explorer`	16,400	The DCScan block (knot) explorer: a Rust HTTP service that aggregates `rope_globalStats`, `rope_listStrings`, `rope_walkString`, and the underlying Reth-served `eth_*` queries; serves the static frontend at `dcscan.io`.
`rope-loadgen` (v2.0 phase 1)	1,200	The benchmark harness used to validate the throughput targets in §11.

The crates are joined by a top-level `Cargo.toml` workspace manifest. The build is reproducible under `cargo build --release --workspace` and produces statically linked binaries whose only runtime dependencies are libssl (for TLS to peer nodes) and libpq (for the optional Postgres-backed indexer used by `rope-explorer`). Genesis configuration is held in `config/networks/mainnet.json` with the chain ID `271828`, the genesis hash `0x6a425482c849cfa07ec253c272a661375469b71c47f499048218a377a24d7203`, the DCR-20 standard declaration, and the initial allocation schedule.

10.2 The bifurcated execution stack

A recurring pattern in the literature on EVM-compatible chains is to fork an existing client (Geth, Erigon, Reth, Nethermind) and patch it to add new operations. We took a different approach. Datachain Rope keeps Reth v1.11.2 unmodified as the EVM execution backend (port 8595, internal-only) and runs `rope-node` as a separate process (port 8545, public-facing, behind nginx) that proxies all `eth_*` calls to Reth and serves all `rope_*` calls itself. The two communicate over the standard Engine API (`engine_newPayload`, `engine_forkchoiceUpdated`) plus a custom `rope_payload` extension that allows `rope-node` to attach a cord-anchor commitment to each EVM block before Reth seals it.

The benefit is decoupling. Reth's EVM correctness, its mempool, its state-trie management, its DevP2P implementation, its EIP-1559 fee market: none of these need to be re-audited by us, because we did not modify them. The cord-native operations (the per-string append, the per-knot erasure, the testimony consensus, the OES key shredding) live entirely in `rope-node`, where we have full control and full responsibility. When Reth ships a security fix, we adopt it by version bump. When we ship a Quipu Canon v2.0 phase-1 sharded-lattice change, Reth is unaffected. The two evolve on independent release cadences.

The cost is one additional process per node (about 240 MB resident memory for `rope-node` on top of Reth's 800 MB to 2.5 GB working set, depending on chain age) and one extra network hop for `eth_*` calls (about 0.3 ms on a localhost loop). For the throughput regime we operate in (currently 1,500 to 4,000 knots per second sustained, see §11) this overhead is invisible.

10.3 The mainnet topology (architectural pattern)

The production network operates a multi-node, multi-provider, multi-region high-availability mesh behind a single public RPC endpoint, structured to survive single-node, single-data-centre, and single-cloud-provider failures. Three architectural points are worth naming, with the operational specifics deliberately omitted from this paper and held in an internal SRE runbook.

First, the failover topology is hierarchical: a primary sequencer produces cord-anchor knots, one or more hot-warm secondaries receive periodic state replicas from the primary, and an upstream load balancer routes traffic to whichever node is currently healthy. The cord-anchor commit interval (approximately three seconds) sets the upper bound on the secondaries' state-replica drift relative to the primary, which is well within the bounds required for transparent failover from the perspective of clients submitting `rope_appendToString` or `eth_sendRawTransaction` operations.

Second, the cord-native state (the StringRegistry, the OES key-shred state, the per-witness reputational scores) is replicated independently of the EVM state via an IPFS content-addressable mesh peered across the production nodes. This separation is deliberate: the EVM state is replicated by mdbx-aware state-snapshot tooling at the storage layer, while the cord-native state is replicated by content-hash addressing at the substrate layer. The IPFS mesh is the substrate-level continuation of the Inka inter-replica audit-by-comparison protocol (§3.1): a content-addressable replication of the cord-native state across geographically and administratively independent custodians.

Third, the topology is designed to be cloud-provider-agnostic and replicable. Future deployments by ecosystem federations (an African Union deployment, a Pacific Island Forum deployment, a Caribbean RWA deployment, etc.) are intended to operate analogous topologies in their own jurisdictional clouds and to anchor their generation cords into the meta-federation cord (§7.2). The architectural pattern, not any specific cloud or hostname, is the part of this paper that matters.

10.4 Validator hardware floor and the Knot Witness baseline

A validator on Datachain Rope ("knot witness" in the Quipu Canon vocabulary) runs `rope-node` and Reth on a single machine. The current minimum specification is 4 vCPU, 8 GB RAM, 200 GB SSD, with chain-state growth bounded at the order of tens of gigabytes per year under current transaction volumes and the Phase 1 sharded-lattice optimisations (§11.1). The minimum staked amount is denominated in DC FAT (the network's native fungible token, standard DCR-20). The witness runs continuously; the slashing condition for a missed cord-anchor testimony is denominated in DC FAT per missed commitment with an exponential-backoff curve (consecutive misses incur progressively larger penalties, which discourages

large-scale forfeit-and-rejoin attacks). Specific stake-minimum and slashing-coefficient values are set by governance and adjusted across network phases; we do not pin them in this paper because the specific values are not the architectural point.

The early-mainnet validator set is deliberately small (single digits) and the per-witness anchor-emission share is correspondingly large in DC FAT terms. This is a bootstrap-phase parameterisation, not a steady-state one. The Quipu Canon v2.0 Phase 2 plan grows the validator set toward a 21-witness target (with mechanical room for further growth as Phases 3 and 4 ship), and the per-witness emission share scales down accordingly. The architectural point is that anchor emission flows institutionally to witnesses-of-record under reputational stake, in contrast to the proof-of-work extraction model of Bitcoin (where emission flows to whoever wins a hash-rate competition). The architectural metaphor is stewardship rather than mining (§5.5). We avoid translating the FAT-denominated emissions into fiat-denominated figures here because the canonical reference price published by the Foundation is an internal valuation rather than a continuously-traded market price, and because the validator-economics question is more soberly assessed at the steady-state validator-set size than at the early-mainnet bootstrap size.

11. Throughput, Scaling, and the Quipu Canon v2.0 Phase Plan

11.1 The current ceiling

The production network's measured ceiling at the time of writing is approximately 1,500 to 4,000 knots per second sustained, against a five-million-knots-per-second target under the Quipu Canon v2.0 design (Datachain Foundation, 2026f). The current bottleneck is well-characterised and lives in three specific code locations:

1. The single global `parking_lot::Mutex<LamportClock>` in `crates/rope-core/src/clock.rs` that serialises every knot append to enforce monotonic ordering across the substrate. At write rates above approximately 4,500 ops per second on contemporary hardware, this lock becomes the dominant CPU consumer.
2. The four-way `RwLock` write cascade in `StringLattice::add_string` at `crates/rope-core/src/lattice.rs` lines 274 to 285. Each new string acquires four write locks in sequence (registry, kind-index, parent-link, OES key-binding), which on profiling consumes approximately 18 percent of total CPU time at the production load.
3. The 100 to 199 BLAKE3 rounds per `OES::derive_key` call in `crates/rope-crypto/src/oes.rs` lines 766 to 801, called once per knot uncached. Profiling shows OES key derivation is approximately 32 percent of per-knot CPU.

The ceiling is therefore not an architectural limitation of the cord lineage. It is an engineering bottleneck of three identifiable, fixable, code-level points. The Phase 1 plan below addresses all three.

11.2 The five-phase scaling roadmap

The Quipu Canon v2.0 (Datachain Foundation, 2026f) specifies a five-phase engineering programme to reach 5 million knots per second aggregate throughput. The phases are sequenced so that each phase ships behind a versioned RPC and v1.2-emitting clients keep working through Phase 3.

Phase	Headline change	Per-node TPS	Aggregate TPS	Eng-months	Canon-breaking?
1	Sharded lattice (16-way), per-wallet head-lock, cached OES key derivation, RocksDB persistence replacing the current in-memory `HashMap<...>` storage	50K to 100K	50K to 100K	3	No
2	Real consensus turned on (`verify_signatures: true` in `consen-sus_orchestrator.rs`) with batched/aggregated post-quantum signatures (Dilithium batch verification), validator set grown from 6 to 21	100K (held)	100K	4	No
3	Horizontal node sharding (16 wallet-prefix clusters, content-addressable routing)	100K per shard	**1.6M aggregate**	8	No (additive RPC namespace)
4	DAG-of-knots replaces strict per-wallet head-string serialisation; the canon advances from v1.2 (per-string append) to v2.0 (per-string causal partial order)	300K	**5M+**	12	**Yes**, but the projection layer keeps v1.2 emitters working forever

Phase	Headline change	Per-node TPS	Aggregate TPS	Eng-months	Canon-breaking?
5	GPU/ASIC post-quantum-signing offload (off-loading Dilithium signing to AMD MI300 / NVIDIA H100 / dedicated PQ-ASIC accelerators)	600K	10M+	4 (after P3)	No

The total design-to-five-million budget is approximately 12 to 14 calendar months at a 16-node fleet size, with an estimated run-cost between EUR 4,000 and 12,000 per month for the cloud infrastructure. Phase 1 alone delivers a thirty-fold improvement over the current ceiling without any canon break, which is sufficient headroom for the next 18 to 24 months of organic growth even if Phases 2 to 4 slip.

11.3 Phase 1 benchmark results

The Phase 1 benchmark harness (`tools/rope-loadgen/`) was completed and run against a single-node deployment in March 2026. The methodology was 30-minute sustained writes; the test was configured to fail if any sample exceeded 1 second of append latency (so the reported p99 is honest rather than truncated, and a single sample above 1 second would have caused the run to be discarded); zero knot loss was verified by `rope_globalStats` invariant assertion at start and end; and the untie path was exercised at 1 percent of write traffic so that the granular-erasure primitive is tested under load. Two methodological caveats deserve explicit statement before the numbers, so that the reader is not misled.

First, **Phase 1 isolates the storage, lattice, and OES improvements; it does not measure full-consensus signature-verification overhead.** The benchmark was run with `verify_signatures: false` in the consensus orchestrator, which is the configuration in which the storage and lattice optimisations can be measured without the cryptographic-verification cost dominating. Phase 2 brings the consensus-verification path online (with batched Dilithium verification) and will report end-to-end production-representative throughput separately. The 33.6-times improvement reported in this section is therefore honest *as a measurement of the targeted Phase 1 optimisations* but is not a representation of post-Phase-2 production throughput, which will land lower than 51,800 knots per second per node and which will be reported in a separate technical note when Phase 2 completes.

Second, **the benchmark is single-node**. The aggregate throughput targets in §11.2 (1.6M and 5M knots per second) require Phase 3 horizontal sharding and Phase 4 knot-DAG topology, neither of which is exercised in the Phase 1 measurement. The Phase 1 numbers should be read as evidence that the substrate's per-node ceiling is well above the current production load, not as evidence of aggregate scalability.

With these caveats stated, the Phase 1 results (Datachain Foundation, 2026g, *Phase 1 Benchmark Results*) are:

Metric	Pre-Phase-1	Post-Phase-1	Improvement
Sustained knots per second	1,540	51,800	33.6 times
p50 append latency	18 ms	0.9 ms	20 times
p99 append latency	240 ms	11 ms	21.8 times
OES derive_key cost (per knot)	1,840 microseconds	89 microseconds	20.7 times (cache hit)
Memory footprint (sustained)	2.4 GB	3.1 GB	+29 percent (acceptable)
Untie path correctness	100 percent	100 percent	unchanged

Metric	Pre-Phase-1	Post-Phase-1	Improvement
Cord-anchor commit interval	3.0 s	3.0 s	unchanged

The improvements come from four code-level changes: (i) replacing the global Lamport mutex with per-shard hybrid logical clocks (HLC, after Kulkarni et al., 2014), (ii) splitting `StringLattice` into 16 shards keyed by the first nibble of `string_id`, with per-shard read-write locks, (iii) caching `OES::derive_key` outputs in an LRU of size 65,536 entries (cache hit rate measured at 99.4 percent on the benchmark workload), and (iv) replacing the in-memory storage with a RocksDB column-family-per-kind persistence layer. None of these changes alters the substrate-level semantics, the canonical RPC surface, or the cord-lineage commitment. They are pure engineering wins on top of the unchanged architecture.

11.4 Hardware-acceleration horizon

The Phase 5 horizon (hardware acceleration for post-quantum signing) is best treated as an open research frontier rather than a settled engineering plan. ML-DSA-65 (Dilithium 3) signing on commodity x86-64 CPUs takes approximately 0.18 ms per signature on AVX2-vectorised builds (Ducas et al., 2018), which already places per-host signing throughput in the same order of magnitude as the Phase 4 aggregate target. Hardware-accelerated implementations of CRYSTALS-Dilithium have been demonstrated on FPGAs and dedicated ASIC designs (Beckwith et al., 2021), and active research on GPU-batched lattice-signature pipelines is ongoing in the post-quantum cryptography community. Whether the Phase 5 figures we project are reachable on commodity GPUs or require dedicated PQ-ASIC hardware will depend on the maturation of these implementation lines over the next two to three years; we do not commit to a specific accelerator in this paper. The architectural point that survives any choice of accelerator is that per-host signing is unlikely to be the binding constraint on aggregate throughput at the 10-million-knots-per-second horizon; the binding constraint will be the on-host network transit of the signature objects (each Dilithium 3 signature is approximately 3,300 bytes) and the cross-shard ordering protocol of Phase 4, neither of which has a substrate-level dependency on cord-versus-block lineage.

Part IV. The Carrier Substrate, the Stakes, and the Future

12. The Carrier Substrate: Datawallets, CARE, and the Political Economy of the Cord

12.1 Strings as songs, wallets as singers, generations as inheritance

The deeper architectural claim of this paper is not principally about lexicon (*knot* rather than *block*) nor principally about programmatic positioning. It is about *carriers*. Across the documented record of pre-modern and non-Western information systems at scale, critical information has not been transmitted across generations by depositing it into anonymous public archives. It has been transmitted by *being carried in named persons*. The substrate that fits this practice is not the file, not the database row, not the block in a chain of strangers' records. It is the *signed thread* held by a carrier whose authority to act on that thread is itself accountable to the institutional or kinship structure within which the thread acquires meaning. The khipu is one substrate of this kind. It is not the only one, and the architectural argument we make in this section does not require it to be the only one.

The ethnographic record supplies the carrier model in detail. The Andean *amauta* taught the imperial princes of Tawantinsuyu their dynastic line as a recited genealogy whose authority was inseparable from the named person who carried it (Garcilaso de la Vega, 1966 [1609]; Murúa, 2008 [c. 1590]). The West African griot, in the long Mande-speaking tradition documented by Hale (1998), held the genealogy of a royal house back twenty generations, not in a centralised archive but in the named living body of a hereditary office-holder, who was paid, witnessed, and audited by the kinship structure that the genealogy bound together. The Polynesian navigator, on the model documented by Lewis (1972) and operationalised in the contemporary Hōkūle'a voyages of the Polynesian Voyaging Society, carried thousand-mile sea routes as remembered chants whose precision exceeded the dead-reckoning instruments available to European navigators of the same period. The Haudenosaunee wampum keeper, on the documented record of Tehanetorens (1999) and Williams (2018), carried the terms of inter-national treaties in a beaded belt whose authority was inseparable from the office of the keeper, with copies replicated in the keeping of every party to the agreement. The Tupicocha *huayrona*, on Salomon's (2004) ethnography, carries the community's administrative cord in a body of named cord-keepers whose collective authority is renewed at each public *huayrona* assembly, and whose deliberate untying and re-tying of knots is performed in front of witnesses in a way that, in computational terminology, anchors the erasure on a public audit trail at the moment of erasure (§3.3, §6). In every case the structure is the same. The vessel was a body. The content was a specific named string of meaning, a song, a genealogy, a route, a treaty, a community ledger. The persistence mechanism was the apprenticeship by which the next carrier inherited the string from the carrier who taught them.

The **Datawallet+** is the modern realisation of this carrier primitive in software. A Datawallet+ is a sovereign personal vessel, an application on a phone, a hardware secure element, an institutional custody arrangement, a multi-signature configuration, that holds the post-quantum key material for its owner's strings on Datachain Rope. Architecturally, the wallet does not contain the strings; the strings live on the network as the per-entity ledgers indexed by the v1.2 string registry (§5; Datachain Foundation, 2026b). The wallet is what makes the strings *signable, walkable, untie-able, and inheritable*. When the wallet appends a knot, the wallet's ML-DSA-65 signature, anchored on the federation cord at the next anchoring interval, constitutes the *voice* that authenticates the next verse on the string. When the wallet calls `rope_walkString`, the wallet exercises the carrier's right to recite its own threads in their entirety.

When the wallet calls `rope_untieKnot`, the wallet exercises the cord-keeper's right to drop a verse from a song under post-quantum cryptographic conditions, leaving the structural memory of the position as a tombstone but rendering the content irrecoverable. The wallet is the singer; the strings are the songs; the post-quantum signature is the voice; the federation cord is the public *huayrona* in front of which the recitations are witnessed.

Multi-generational transmission then becomes a protocol rather than a hope. A Datawallet+ is *designed to admit* authenticated inheritance: a string's authority can be transferred, by a signed instruction from the prior holder, to a designated successor, with the federation cord anchoring the transfer event as a knot tied under a reserved `event_type = inheritance` extension in the open event-type namespace specified in the Quipu Primitive Canon (Datachain Foundation, 2026a, §4). The inheritance event-type is not, at the time of writing, a shipped Canon-named primitive; it is the architectural target of the v1.3 Canon revision, scheduled to follow the v1.2 string registry (Datachain Foundation, 2026b) and prior to the Canon v2.0 sharding (Datachain Foundation, 2026f). When shipped, the successor will inherit the *capacity to act* on the strings, the right to append, walk, and untie, the way an apprentice inherits a song from the singer who taught them, or the way a Tupicocha *quipucamayoq* office is renewed at the *huayrona*. The successor will not inherit the substrate (the strings remain on the network, owned by the wallet identity, accountable to the federation cord); the successor will inherit the carrier's role with respect to the substrate. Granular reversibility will be preserved across the inheritance: individual knots on an inherited string remain untie-able by the inheritor under their own authority, just as a song's later carriers have always been free to drop verses that no longer fit the world they sing into. The ML-DSA-65 signatures on the inheritance knot, anchored on the federation cord, will constitute the cryptographic analogue of the public witnessing that the bardic and cord-keeping traditions have always required: the inheritance will be publicly observed by the federation and rendered tamper-evident under post-quantum signatures, exactly as the apprenticeship was publicly observed by the assembled cord-keepers in Tupicocha or by the assembled lineage-elders in the Mande. Until v1.3 ships, multi-generational transmission on the production network proceeds under the existing key-rotation primitives; the carrier-substrate argument is therefore an architectural claim about the substrate's *intended trajectory*, not a description of a feature already in user hands.

The substrate that emerges is the substrate that *fits* what human cultures at scale have always done with critical information. An identity is a song. A medical history is a song. A territorial boundary, a real-world asset deed, a biodiversity attestation, an educational credential, a cultural-heritage receipt, a marriage contract, a will, a record of land stewardship: each is a song, sung by a named wallet, owned by a named person, inheritable by a named successor, untie-able on the holder's authority, federated upward to a cord that audits without seizing. The cord lineage of Datachain Rope plus the per-person carrier vessel of Datawallet+ realises in cryptography what the oral lineage realised in the human voice: a population-scale infrastructure for critical information that does not flatten the songs into a database row, does not impose anonymity on what was always named, does not impose immutability on what was always editable, and does not impose extraction on what was always stewarded. We submit that this is the substantive architectural claim that the contemporary literatures on indigenous data sovereignty, on substrate-political accountability, and on the right to be forgotten have been seeking to ground but have not previously had a substrate adequate to ground in. The substantive claim is not a programme of values; it is a substrate of carriers, equipped at last with the cryptographic primitives that earlier civilizational iterations did not possess.

12.2 The CARE principles, indigenous data sovereignty, and the substrate

The CARE Principles for Indigenous Data Governance (Carroll et al., 2020), Collective Benefit, Authority to Control, Responsibility, Ethics, are the most influential framework for indigenous data sovereignty internationally. They were developed by the Global Indigenous Data Alliance and adopted by the Research Data Alliance in 2020. The principles are *governance* principles: they specify how data *about* indigenous communities should be handled, by whom, under what authority, with what accountability. They have been increasingly adopted by funding agencies (e.g. Australian National Health and Medical Research Council, Canadian Tri-Council policy), by repositories, and by editorial standards.

The CARE principles are *not*, in their authors' formulation, a specification of *substrate*. They specify the *operations* that should be supported on the substrate, collective consent, individual withdrawal, authori-

sation per use, audit of access, and they leave the substrate question open. Datachain Rope's argument is that the cord lineage, instantiated as the *string registry*, is the substrate that *natively supports* the CARE operations: a per-entity string is the natural unit of "authority to control" (the string's owner authorises all writes); `rope_untieKnot` is the natural realisation of "responsibility to honour withdrawal" (the substrate makes withdrawal a primitive); the testimony pool is the natural realisation of "ethics of collective audit" (witnesses vouch for events under reputational stake). The block lineage cannot natively support these operations; it can support them only at the application layer, with the substrate-level append-only-ness as a hostile background. The cord lineage can support them natively. This is the substrate-level meaning of CARE.

We are explicit that this argument generalises beyond the Andean case. The cord lineage is, we propose, a *family* of substrates whose other members include the wampum belt of the Haudenosaunee, the *kipu*-related cord systems of the Polynesian Pacific (e.g. the *lukasa* memory boards of the Luba in the Congo basin, which are functionally analogous though structurally distinct; Reefe, 1977), the lineage records of the Yoruba *aroko* signal system, and the seasonal-calendar cords of the Tibetan plateau pastoralists. We do not claim equivalence among these substrates. We claim that they share a family resemblance, sovereign threading, signed origin, granular reversibility, federated aggregation, and that the cord-lineage substrate of Datachain Rope is hospitable to all of them. A future Pacific Federation deployment might draw its categorical taxonomy more from wampum than from khipu; a future Congo Basin Federation deployment might draw more from *lukasa* than from khipu. The substrate is the same; the cultural specifications layer on top of it.

12.3 The political economy of the substrate

The substrate choice is not politically neutral. We have argued repeatedly that the *block* lineage carries an industrial, mineralogical, extractive vocabulary, and we will now name what this vocabulary entails politically. Bitcoin's block-mining metaphor commits the substrate to an extractive value-creation model in which value is *won* through energy expenditure and computational work. This is structurally adequate for a peer-to-peer electronic-cash use case, but it has produced, as a side-effect, the largest decentralised energy-consumptive industry in human history (Bitcoin mining at peak consumed approximately 0.5% of global electricity production; de Vries, 2018). It has also produced a financialised speculation that has, on the documented record, been an instrument of capital flight, money laundering, and rent-extraction in regulatory-arbitrage geographies (Howson, 2020; FATF, 2021).

The cord lineage, by contrast, commits the substrate to an institutional value-creation model in which value is *recognised* through testimony and accountability. The witness is paid for vouching; the cord-keeper is paid for keeping. Value flows from the *labour of stewardship* rather than from the *expenditure of energy on cryptographic puzzles*. This is a different political economy. It is, we propose, a more apt political economy for a public-data infrastructure that aspires to administer identity, biodiversity, real-world assets, and cultural heritage at the scale of billions of subjects. The aptness is not aesthetic. It is operational: a substrate whose value model is stewardship is a substrate that *trains* its operators in stewardship behaviours; a substrate whose value model is extraction *trains* its operators in extraction behaviours. The substrate's metaphor becomes, over time, the substrate's culture.

The argument here is not that "indigenous values are good and Western values are bad". It is more careful and more interesting. It is that *every* substrate's metaphor becomes its culture, that the dominant-block-lineage metaphor has produced (predictably, given its metaphor) an extractive-financial-speculation culture, that the cord lineage's metaphor would predictably produce a stewardship-institutional-accountability culture, and that the latter is the one we need for the substrate of the next half-century of public-data infrastructure. The choice of substrate is therefore a constitutional choice about the kind of society the substrate's operators will, by training, become.

13. The Three Paradoxes Resolved, and the Stakes

13.1 The block-mention paradox

Statement: in a sovereign-thread architecture, two strangers cannot share a "block", because a block presupposes a shared container into which both deposit records, but each thread is, by definition, the property of one named keeper. The paradox arises because the dominant blockchain literature *assumes* the block primitive and then attempts to deduce sovereign-thread properties from it. The deduction fails: shared containers cannot host sovereign threads, except by structural concession (off-chain side databases, application-layer per-user queues, etc.) that voids the substrate's integrity claim.

Resolution in Datachain Rope: the architectural primitive is the *knot* on a *string*, not the block. The federation cord exists, but its role is *aggregation across signed strings*, not *containerisation of strangers' records*. Every event on the network is the event of a specific named string. There is no shared container of strangers; there is a federated aggregation of sovereigns. The paradox is dissolved at the level of substrate: the *block-mention paradox* is a paradox *for block-lineage substrates*; on cord-lineage substrates, it does not arise.

13.2 The erasure paradox

Statement: append-only block-chained data structures are constitutionally incompatible with personal-data erasure obligations under Article 17 GDPR (and analogues under CCPA, LGPD, PIPL, POPIA, the EU AI Act, DORA). Erasing a record on such structures requires either rewriting the chain (impossible on a public network with active validators) or storing personal data off-chain in a side database (a workaround that voids the on-chain integrity claim and exposes the data to side-channel and custodial risks at the off-chain database).

Resolution: `rope_untieKnot` (§6) realises granular erasure as a substrate primitive. Cryptographic destruction of the OES key shreds renders the payload irrecoverable; tombstone preservation maintains the audit trail; string-integrity preservation ensures that all unrelated records on the same string are unaffected; federation anchoring of the erasure preserves the audit trail at the federation level. The paradox is dissolved at the level of substrate: cord-lineage substrates are *natively* compatible with the right to be forgotten because the cord lineage was always granularly erasable, at Tupicocha, in cotton, by the cord-keepers' own hand. The cryptographic version of `rope_untieKnot` is the contemporary realisation of an information-substrate property that has been continuously practised in the Andes for at least four hundred and seventy years.

13.3 The commoditisation / speculation-impossibility paradox

Statement: the moment a ledger is structured around fungible blocks of fungible transactions submitted by anonymous parties, the records on that ledger become commodities, substitutable, tradeable, speculative. This is structurally adequate for monetary systems (Bitcoin) but structurally inadequate for systems whose records are *not fungible*: identity claims, biodiversity attestations, real-world asset deeds, cultural-heritage receipts, sovereign personal histories. The paradox arises because the same substrate that enables peer-to-peer electronic cash *imposes commoditisation on every other record on the substrate*, including those for which commoditisation is structurally inappropriate.

Resolution: a knot tied on a sovereign string is not a fungible block. It is a record on a specific named thread. The medium does not invite commoditisation of records; it invites stewardship of records. DC FAT (the network's native fungible utility token) is fungible, but it is one *kind* of knot among many, not the structural unit of the medium. The medium thus accommodates monetary fungibility (a wallet's balance is a number on a wallet's string) without imposing it on non-fungible records (an identity attestation is a claim on a DID's string, not a tradeable block). We call this the *speculation-impossibility thesis* in its weak form: the architecture does not preclude monetary commoditisation, but it does preclude the commoditisation of identity records, biodiversity claims, real-world asset deeds, and personal histories. These remain stewarded threads, not tradeable blocks.

The strong form of the thesis, which we do not require for the architectural argument but which we offer as a hypothesis, is that the medium's metaphor will, over time, *culturally* discourage the commoditisation of records that the medium *technically* permits to be commoditised. A wallet whose owner thinks of their string as their *biographical thread* (rather than as their *trading account*) will be less inclined to engage in speculative trading on that string. This is an empirical claim about the cultural effect of substrate metaphors and we propose it as a research programme rather than as an established result. But the architectural claim, that the cord lineage substrate does not *impose* commoditisation, where the block lineage substrate does, is not a hypothesis. It is a property of the substrates as they are designed.

13.4 The stakes named in plain language

We have argued, across the body of the paper, that the substrate choice for public-ledger infrastructure is a constitutional choice with consequences across multiple registers. We name them, in plain language, here.

Technical stakes. The block lineage is structurally incompatible with Article 17 GDPR and its global analogues. The cord lineage is structurally compatible. As more jurisdictions adopt advanced data-protection regimes, more public-ledger deployments will need to be substrate-compatible with granular erasure. Substrate-incompatible deployments will hit a regulatory ceiling that no application-layer engineering can lift. The cord lineage has more architectural runway.

Legal stakes. A regulator or court that rules on the compatibility of a public-ledger deployment with data-protection law is, in effect, ruling on the substrate. Bitcoin and Ethereum will, sooner or later, face such rulings in jurisdictions that take Article 17 seriously. Datachain Rope is designed to pass such rulings on first principles. The first such ruling, when it arrives, will be a regulatory inflection point for the entire field; the cord-lineage chains will be on the right side of it.

Economic stakes. The block lineage's value-creation model rewards extraction. The cord lineage's value-creation model rewards stewardship. The two produce different industries. The block-lineage industry is, on the empirical record, energy-consumptive, speculation-prone, and rent-extractive. The cord-lineage industry, on its design intentions, is institutionally compensated, stewardship-trained, and accountability-cultured. The two industries will compete for talent, for capital, and for political legitimacy over the next decade. The question of which prevails is partly a question of which substrate the regulators, the funders, and the operators choose.

Cultural-political stakes. The substrate-political argument is not decorative, and we register it with moral spine rather than technocratic neutrality. The cord lineage was suppressed by an ecclesiastical campaign that began with the 1583 Council of Lima and continued through the seventeenth century. The suppression was substantively effective: the bulk of the imperial archive at Cuzco was lost, most regional collections were lost, and the *quipucamayoq* profession was driven underground or extinguished. The suppression was wrong on the very criterion the colonial administration itself implicitly conceded each time it kept reusing khipu testimony in tribute records and in colonial litigation while in parallel ordering its destruction. The structural reading and the moral reading do not compete. Successions of civilizational iterations carry implicit redrawings of the boundary between *infrastructure* and *cultural artefact*, and a substrate that has been pushed across that boundary in one direction can be returned, by a successor iteration, in the other. That structural lens is the way of seeing what the suppression *was*; it is not a softening of the historical fact. The cord lineage's revival is the substantive case: a substrate that an earlier civilizational iteration relegated out of the infrastructural register, equipped now with the cryptographic primitives unavailable to that earlier iteration, returns to the register in which it was operational. The per-person Datawallet+ vessel returns the carrier role to the named, accountable, inheriting subject that the bardic and cord-keeping traditions always took for granted. The revival is

therefore not a neutral technical choice; it is a position-taking on which information substrates and which carrier offices are recognised as legitimate computational lineages, and on what restitution looks like when the wronged tradition's substrate is engineered to be operational again. This position-taking matters for the constitutional politics of the next decade of public-data infrastructure, particularly for the African Union Data Policy Framework deployment, for the Pacific Data Sovereignty programme, for the Caribbean and Amazon Basin RWA deployments through Tanastok, and for the Congo Basin biodiversity-credit deployments through NaturaProof.

Scientific stakes. The architectural lineage is documented in the peer-reviewed literature on three independent shelves: anthropology of the Andes (Locke, the Aschers, Urton, Salomon, Brokaw, Hyland), post-quantum cryptography (NIST FIPS 204 and 205, Bernstein and Lange, Mosca), and distributed systems (Lamport, Castro and Liskov, Haber and Stornetta, Nakamoto). The contribution of Datachain Rope is the *synthesis* across those shelves. The synthesis is the kind of inter-disciplinary engineering that the next decade of computer science, anthropology, and law will increasingly need to produce. Datachain Rope is one of the early instances of this kind of synthesis. Its scientific stake is the demonstration that this synthesis is possible, that it is technically realisable, and that it produces a substrate more adequate to its problem-set than the dominant alternative.

14. Limitations, Future Work, and a Visionary Close

14.1 Limitations

We name four limitations explicitly so that the paper's claims are not over-read.

First, the decipherment of the khipu remains incomplete. Locke's numerical layer is consensual; Urton's binary structural argument is well-supported and influential; Hyland's logosyllabic argument is conservative in its claim and increasingly accepted in outline but contested in detail. The architectural argument of this paper does not depend on the contested layers; the position-addressable, typed, hierarchically-aggregated, signed, replicated, granularly-erasable structure is consensual. But future archaeological and linguistic findings will refine the picture, and Datachain Rope's design must remain open to incorporating new structural lessons. If, for example, the phonemic layer is shown to support a richer alphabet of structural choices, the per-knot `event_type` enumeration may be extended in line with that richer alphabet.

Second, the production deployment of Datachain Rope at the time of writing is at the early-mainnet stage. Throughput is currently bounded at the 1,500–4,000 knots per second sustained range under realistic write-load (Datachain Foundation, 2026f, Quipu Canon v2.0, Scaling Roadmap), against a five-million-knots-per-second target under design. The architectural translation of the khipu lineage outlined in this paper does not, *per se*, dictate any specific throughput floor or ceiling; it dictates the *kind* of operations the substrate must support (per-string append, granular erasure, federated aggregation). The throughput question is a matter of engineering, sharded lattices, batched signatures, RocksDB persistence, knot-DAG topology, not of substrate. We treat that separately in the Quipu Canon v2.0 roadmap (Phase 1: 50K–100K per node; Phase 2: 100K with consensus on; Phase 3: 1.6M aggregate via horizontal sharding; Phase 4: 5M+ aggregate via knot DAG; Phase 5: 10M+ via PQ-signing offload).

Third, the legal status of cryptographic erasure as sufficient under Article 17 GDPR is an evolving area of European data-protection law (the deeper hedging is in §6.2). No EDPB guideline addresses cryptographic erasure as Article 17 compliance directly; the closest authorities are the EDPB Guidelines 04/2019 on Article 25 (data protection by design and by default, version 2.0, 2020) and the Article 29 Working Party's *Opinion 05/2014 on Anonymisation Techniques* (WP29, 2014), and the European Court of Justice has not yet issued a decision specifically on substrate-level cryptographic erasure. Datachain Rope's claim is that the architecture provides the *strongest available substrate-level realisation* of the right to be forgotten currently engineerable and satisfies the WP29 (2014) technical conditions for effective key-destruction, not that it has been judicially confirmed as sufficient. The first ECJ decision on substrate-level cryptographic erasure will be a significant moment for the field; we will follow the proceedings and adjust the architecture if the decision identifies additional requirements.

Fourth, the carrier-substrate-pluralisation claim must not be over-stated. The khipu is one of several non-European information substrates that could, under appropriate analysis, inform contemporary infrastructure (the Yoruba *aroko* signal system, the Iroquois wampum belt, the Nsibidi script of the Cross River Igbo, the Indus Valley signs, the Chinese counting-rod system, the Polynesian *kipu*-analogue cords, the Luba *lukasa* memory boards, the Tibetan plateau pastoralists' seasonal-calendar cords). The argument of this paper is that the *Andean cord* lineage fits a particular contemporary problem-set particularly well. Other lineages would fit other problem-sets. The pluralisation of computational lineages is the larger programme; this paper is one entry in it. We see Datachain Rope as the *cord-lineage* entry; we

expect, and hope for, *wampum-lineage*, *lukasa-lineage*, *aroko-lineage* entries from other research and engineering communities.

14.2 Future work

We propose six directions for future work, organised in increasing scale.

(a) Formal ethnomathematical analysis of khipu aggregation. A close re-reading of the Aschers' corpus, with contemporary computational tools, to identify further structural primitives that might be promoted into the Datachain Rope substrate. Candidates include weighted-aggregation functions documented in the corpus, the *checksum* knot patterns reported in some specimens, and the *segmentation marker* patterns that may correspond to per-string-genesis indicators.

(b) Comparative legal analysis. A systematic comparative analysis of `rope_untieKnot`-style mechanisms against Article 17 GDPR, CCPA §1798.105, LGPD Article 18, PIPL Article 47, POPIA §24, EU AI Act Article 10, DORA, the African Union Data Policy Framework data-subject rights articles, and the emerging Pacific Data Sovereignty principles. The aim is a published article, in a venue like *European Data Protection Law Review* or the *International Data Privacy Law* journal, that establishes the substrate-level erasure claim on a thoroughly comparative basis.

(c) Empirical study of substrate-cultural effects. A multi-year empirical study of operator behaviour on Datachain Rope as the network matures, testing the speculation-impossibility thesis against observed transaction patterns. The hypothesis: that wallets whose owners have explicit string-ownership awareness exhibit a measurable difference in transaction patterns (more attestation knots relative to transfer knots, lower trading frequency on the wallet's primary holdings, higher engagement with `rope_untieKnot`) than wallets that operate under a block-lineage mental model.

(d) Cross-lineage substrate-level federation. Engineering work on the Federation Generation Protocol's cross-jurisdictional and cross-lineage capabilities. The aim is a working federation, by 2028, between Datachain Rope (cord-lineage substrate) and at least one other public-ledger deployment in a different lineage, with cross-lineage audit and bridged-but-sovereign asset transfers. This is the architectural realisation of the post-Westphalian public-data infrastructure thesis (§7.2).

(e) Andean-studies and indigenous-data-governance dialogue. This paper proposes a research programme that, to be conducted with appropriate institutional accountability to the cord-keeping community whose lineage the architecture draws on, requires partnerships that we are at the moment of writing actively building rather than able to evidence. We name this honestly: we have not yet signed memoranda of understanding with the Tupicocha *quipucamayoq* community, nor have we co-authored any published material with named Andean-studies scholars, nor have we adopted the CARE Principles for Indigenous Data Governance (Carroll et al., 2020) into the Foundation's governance charter beyond a current statement of commitment. The work we are committing to over the next eighteen to twenty-four months is: (i) to invite at least two named Andean-studies scholars onto the Foundation's research advisory board, with publicly visible affiliation and remuneration; (ii) to seek a formal MoU with the Tupicocha community council under terms of free, prior, and informed consent (Salomon, 2004) covering both academic citation practice and any future Foundation engagement that touches on the village's khipu practice; (iii) to commission an external review, by a scholar of indigenous data sovereignty and substrate-political accountability, of the Foundation's vocabulary, governance, and IP policy; and (iv) to publish the outputs of (i)–(iii) before submitting any successor to this paper to a peer-reviewed venue at which the indigenous-data-sovereignty claim is load-bearing. Until those four artefacts exist, the §12 carrier-substrate positioning should be read as a *programme* rather than an established practice, and we accept that reviewers at *Social Studies of Science* and *Big Data & Society* are entitled to press exactly here.

(f) Indigenous Federation deployments. Concrete deployment partnerships with indigenous-data-sovereignty bodies (for example, Te Mana Raraunga in Aotearoa, Maiam nayri Wingara in Australia, the Global Indigenous Data Alliance, the African Union Data Policy implementing committees, regional Pacific Forum bodies). The goal is *Indigenous Federations* of public-ledger infrastructure, sovereign at the regional level, federated across regions through the Generation Cord pattern, with data-governance under CARE-aligned protocols and substrate under cord-lineage architecture.

14.3 Conclusion

The argument of this paper has been that the dominant historiography of computing has bracketed the cord lineage out, that the bracketing is not, on the evidence, defensible, and that the substrate properties supplied by the cord lineage (per-entity ledgering, signed origin, hierarchical aggregation, granular erasure, federated audit) are exactly the substrate properties most acutely needed by the next phase of internet-scale public-data infrastructure. We have argued further that the historical contingency that prevented the cord lineage from being equipped with adversarial cryptographic verification, the absence of public-key cryptography in pre-Columbian Andean society, has been removed by the standardisation of post-quantum cryptographic primitives in 2024 (NIST FIPS 204, FIPS 205), and that Datachain Rope is one engineering realisation of the resulting architectural opportunity.

We have advanced this argument under three deliberate restraints. First, we do not claim that the khipu *is* a distributed ledger in any metaphysical sense; we claim that it *instantiates the operative properties of* one. Second, we do not claim that substrate-level cryptographic erasure is a settled legal answer to the GDPR Article 17 problem; we claim that it is the strongest substrate-level realisation of Article 17 currently engineerable on a public smartchain, with the judicial confirmation an open empirical question. Third, we do not claim that the architectural revival is, at the time of writing, a fully realised partnership with the contemporary cord-keeping community; we claim that it is a research programme that, to be conducted with appropriate institutional accountability, requires partnerships with Andean-studies scholarship and with the present-day cord-keeping community of Tupicocha that we are at the moment of writing actively building (§14.2(e)).

Six empirical questions follow from the architecture and merit study by the distributed-systems, post-quantum-cryptography, anthropology-of-computing, and law-and-technology communities. (i) Does Phase 2 (consensus-verification online with batched Dilithium) sustain throughput within engineering tolerance of the Phase 1 ceiling reported in §11.3? (ii) Does the Federation Generation Protocol (§7.2) admit cross-jurisdictional federation under the regulatory constraints of GDPR, the EU AI Act, and DORA without protocol-level modification? (iii) Will a court of competent jurisdiction (most plausibly the Court of Justice of the European Union) confirm the sufficiency of substrate-level cryptographic erasure for Article 17, and if not, what residual application-layer measures must accompany `rope_untieKnot` to satisfy the residual legal requirement? (iv) Under what economic conditions does the per-witness reputational stake produce a robust adversarial-resistance threshold equivalent to or stronger than the current proof-of-stake economic-security model? (v) Does the architectural revival admit re-export of substrate properties to ecosystem federations (an African Union deployment, a Pacific Island Forum deployment, a Caribbean RWA deployment) without architectural modification? (vi) What is the appropriate IP and benefit-sharing arrangement between the Datachain Foundation and the Tupicocha community council under a regime of free, prior, and informed consent and the CARE Principles for Indigenous Data Governance?

We close on the registration of a position. The cord lineage was operational. It administered an empire of approximately ten million subjects across approximately two million square kilometres of altitude-spanning terrain. It was suppressed by an ecclesiastical campaign that began with the 1583 Council of Lima and continued through the seventeenth century, and a substantial portion of the inherited corpus was destroyed. The structural reading (a succeeding civilizational iteration reclassifying a still-operational substrate from administration into superstition) is not a softening of that fact; it is a way of seeing what the suppression *was* and of identifying what restitution would *require*. It was preserved, in residual form, by a small community in the Lima highlands. It admits revival. Its revival, equipped with post-quantum cryptography and engineered as a public smartchain, is the architectural proposition that this paper invites the reader to take seriously. Whether the revival is, in the long arc, *successful* in the sense that matters (whether the resulting substrate becomes the substrate of choice for the public-data infrastructure of the next century) is an empirical question whose answer this paper does not pre-empt. What this paper has tried to establish is the weaker claim: that the question is well-posed, that it is technical, and that it deserves the considered attention of the communities to which it is addressed.

References

Acosta, J. de. 2002 [1590]. *Historia natural y moral de las Indias*. Translated by F. M. López-Morillas. Durham, NC: Duke University Press.

African Union. 2022. *African Union Data Policy Framework*. Addis Ababa: African Union Commission.

Article 29 Data Protection Working Party (WP29). 2014. *Opinion 05/2014 on Anonymisation Techniques*. WP216. Adopted 10 April 2014. Brussels: European Commission.

Ascher, M., and R. Ascher. 1981. *Code of the Quipu: A Study in Media, Mathematics, and Culture*. Ann Arbor: University of Michigan Press. Reprinted 1997, Mineola, NY: Dover.

Aspray, W. 1990. *John von Neumann and the Origins of Modern Computing*. Cambridge, MA: MIT Press.

Back, A. 2002. *Hashcash, A Denial of Service Counter-Measure*. Technical report.

Beckwith, L., D. T. Nguyen, and K. Gaj. 2021. "High-Performance Hardware Implementation of CRYSTALS-Dilithium." In *Proceedings of the 2021 International Conference on Field-Programmable Technology (ICFPT)*. Auckland: IEEE.

Bernstein, D. J., A. Hülsing, S. Kölbl, R. Niederhagen, J. Rijneveld, and P. Schwabe. 2019. "The SPHINCS+ Signature Framework." In *ACM CCS 2019*. New York: ACM.

Bernstein, D. J., and T. Lange. 2017. "Post-Quantum Cryptography." *Nature* 549 (7671): 188–194.

Bowker, G. C., and S. L. Star. 1999. *Sorting Things Out: Classification and Its Consequences*. Cambridge, MA: MIT Press.

Brokaw, G. 2010. *A History of the Khipu*. Cambridge: Cambridge University Press.

Buterin, V. 2014. *Ethereum White Paper: A Next-Generation Smart Contract and Decentralized Application Platform*. Self-published.

Buterin, V. 2024. "How to Hard-Fork to Save Most Users' Funds in a Quantum Emergency." *Ethereum Research* (ethresear.ch), 5 March 2024. https://ethresear.ch/t/how-to-hard-fork-to-save-most-users-funds-in-a-quantum-emergency/18901.

Carrión Cachot, R. 1948. "La cultura Chavín: Dos nuevas colonias, Kuntur Wasi y Ancón." *Revista del Museo Nacional de Antropología y Arqueología* 2 (1): 99–172.

Carroll, S. R., I. Garba, O. L. Figueroa-Rodríguez, et al. 2020. "The CARE Principles for Indigenous Data Governance." *Data Science Journal* 19: 43.

Castro, M., and B. Liskov. 1999. "Practical Byzantine Fault Tolerance." In *Proceedings of the Third Symposium on Operating Systems Design and Implementation (OSDI '99)*, 173–186. New Orleans: USENIX.

Charles, J. 2010. *Allies at Odds: The Andean Church and Its Indigenous Agents, 1583–1671*. Albuquerque: University of New Mexico Press.

Cieza de León, P. 1998 [1553]. *The Discovery and Conquest of Peru: Chronicles of the New World Encounter*. Edited and translated by A. P. Cook and N. D. Cook. Durham, NC: Duke University Press.

Cobo, B. 1979 [1653]. *History of the Inca Empire*. Translated and edited by R. Hamilton. Austin: University of Texas Press.

Conklin, W. J. 1982. "The Information System of Middle Horizon Quipus." *Annals of the New York Academy of Sciences* 385: 261–281.

Conklin, W. J. 2002. "A Khipu Information String Theory." In *Narrative Threads: Accounting and Recounting in Andean Khipu*, edited by J. Quilter and G. Urton, 53–86. Austin: University of Texas Press.

Datachain Foundation. 2026a. *Quipu Primitive Canon v1.1, Knots, Strings, Tombstones*. Internal canonical specification. Geneva: Datachain Foundation SAS.

Datachain Foundation. 2026b. *Quipu Primitive Canon v1.2, String Registry*. Internal canonical specification. Geneva: Datachain Foundation SAS.

Datachain Foundation. 2026c. *Quipu Canon v1.2, Knot, Transaction, Event Distinction*. Internal canonical specification. Geneva: Datachain Foundation SAS.

Datachain Foundation. 2026d. *Datachain Rope Technical Whitepaper v4*. Geneva: Datachain Foundation SAS.

Datachain Foundation. 2026e. *DC FAT Technical Tokenomics Specification v1.2*. Geneva: Datachain Foundation SAS.

Datachain Foundation. 2026f. *Quipu Canon v2.0: Scaling Roadmap to 5M+ Knots per Second*. Geneva: Datachain Foundation SAS.

Datachain Foundation. 2026g. *Quipu Canon v2.0 Phase 1 Benchmark Results: Sharded Lattice, Cached OES, RocksDB Persistence*. Internal technical report. Geneva: Datachain Foundation SAS.

de Vries, A. 2018. “Bitcoin’s Growing Energy Problem.” *Joule* 2 (5): 801–805.

Diffie, W., and M. E. Hellman. 1976. “New Directions in Cryptography.” *IEEE Transactions on Information Theory* 22 (6): 644–654.

Ducas, L., E. Kiltz, T. Lepoint, V. Lyubashevsky, P. Schwabe, G. Seiler, and D. Stehlé. 2018. “CRYSTALS-Dilithium: A Lattice-Based Digital Signature Scheme.” *IACR Transactions on Cryptographic Hardware and Embedded Systems* 2018 (1): 238–268.

European Data Protection Board (EDPB). 2020. *Guidelines 04/2019 on Article 25: Data Protection by Design and by Default*, version 2.0. Adopted 20 October 2020. Brussels: EDPB.

European Parliamentary Research Service (EPRS). 2019. *Blockchain and the General Data Protection Regulation*. Study PE 634.445. Brussels: European Parliament.

Financial Action Task Force (FATF). 2021. *Updated Guidance for a Risk-Based Approach to Virtual Assets and Virtual Asset Service Providers*. Paris: FATF.

Finck, M. 2018. “Blockchains and Data Protection in the European Union.” *European Data Protection Law Review* 4 (1): 17–35.

Freeth, T., Y. Bitsakis, X. Moussas, et al. 2006. “Decoding the Ancient Greek Astronomical Calculator Known as the Antikythera Mechanism.” *Nature* 444 (7119): 587–591.

Garcilaso de la Vega, El Inca. 1966 [1609]. *Royal Commentaries of the Incas, and General History of Peru*. Translated by H. V. Livermore. Austin: University of Texas Press.

Guaman Poma de Ayala, F. 1980 [c. 1615]. *El primer nueva corónica y buen gobierno*. Edited by J. V. Murra and R. Adorno. Mexico City: Siglo XXI.

Haas, J., and W. Creamer. 2006. “Crucible of Andean Civilization: The Peruvian Coast from 3000 to 1800 B.C.” *Current Anthropology* 47 (5): 745–775.

Haber, S., and W. S. Stornetta. 1991. “How to Time-Stamp a Digital Document.” *Journal of Cryptology* 3 (2): 99–111.

Hale, T. A. 1998. *Griots and Griottes: Masters of Words and Music*. Bloomington: Indiana University Press.

Hodges, A. 1983. *Alan Turing: The Enigma*. London: Burnett Books.

Howson, P. 2020. “Climate Crises and Crypto-Colonialism: Conjuring Value on the Blockchain Frontiers of the Global South.” *Frontiers in Blockchain* 3: 22.

Hyland, S. 2014. “Ply, Markedness, and Redundancy: New Evidence for How Andean Khipus Encoded Information.” *American Anthropologist* 116 (3): 643–648.

Hyland, S. 2017. “Writing with Twisted Cords: The Inscriptive Capacity of Andean Khipus.” *Current Anthropology* 58 (3): 412–419.

Kulkarni, S. S., M. Demirbas, D. Madappa, B. Avva, and M. Leone. 2014. “Logical Physical Clocks.” In *Principles of Distributed Systems (OPODIS 2014)*, edited by M. K. Aguilera, L. Querzoni, and M. Shapiro, 17–32. Lecture Notes in Computer Science 8878. Cham: Springer.

Lakoff, G., and M. Johnson. 1980. *Metaphors We Live By*. Chicago: University of Chicago Press.

Lamport, L. 1978. “Time, Clocks, and the Ordering of Events in a Distributed System.” *Communications of the ACM* 21 (7): 558–565.

Lewis, D. 1972. *We, the Navigators: The Ancient Art of Landfinding in the Pacific*. Honolulu: University of Hawaii Press.

Locke, L. L. 1923. *The Ancient Quipu, or Peruvian Knot Record*. New York: American Museum of Natural History.

Medrano, M., and G. Urton. 2018. “Toward the Decipherment of a Set of Mid-Colonial Khipus from the Santa Valley, Coastal Peru.” *Ethnohistory* 65 (1): 1–23.

Mignolo, W. D. 2003. *The Darker Side of the Renaissance: Literacy, Territoriality, and Colonization*. 2nd ed. Ann Arbor: University of Michigan Press.

Mignolo, W. D. 2011. *The Darker Side of Western Modernity: Global Futures, Decolonial Options*. Durham, NC: Duke University Press.

Mosca, M. 2018. “Cybersecurity in an Era with Quantum Computers: Will We Be Ready?” *IEEE Security & Privacy* 16 (5): 38–41.

Murúa, M. de. 2008 [c. 1590]. *Historia general del Pirú*. Edited by T. Cummins and B. Anderson. Los Angeles: J. Paul Getty Museum.

Nakamoto, S. 2008. *Bitcoin: A Peer-to-Peer Electronic Cash System*. Self-published whitepaper.

National Institute of Standards and Technology (NIST). 2024a. *Module-Lattice-Based Digital Signature Standard*. FIPS 204. Gaithersburg, MD: NIST.

National Institute of Standards and Technology (NIST). 2024b. *Stateless Hash-Based Digital Signature Standard*. FIPS 205. Gaithersburg, MD: NIST.

Needham, J. 1959. *Science and Civilisation in China, Volume 3: Mathematics and the Sciences of the Heavens and the Earth*. Cambridge: Cambridge University Press.

Pärssinen, M., and J. Kiviharju. 2004. *Textos Andinos: Corpus de textos khipu incaicos y coloniales*. Vol. 1. Madrid: Instituto Iberoamericano de Finlandia.

Polo de Ondegardo, J. 1916 [1571]. *Informaciones acerca de la religión y gobierno de los Incas*. Edited by H. H. Urteaga. Lima: Sanmartí.

Quijano, A. 2000. “Coloniality of Power, Eurocentrism, and Latin America.” *Nepantla: Views from South* 1 (3): 533–580.

Quilter, J., and G. Urton, eds. 2002. *Narrative Threads: Accounting and Recounting in Andean Khipu*. Austin: University of Texas Press.

Reefe, T. Q. 1977. “Lukasa: A Luba Memory Device.” *African Arts* 10 (4): 48–50, 88.

Regulation (EU) 2016/679 (General Data Protection Regulation). 2016. *Official Journal of the European Union* L 119/1 (4 May 2016).

Rivest, R. L., A. Shamir, and L. Adleman. 1978. “A Method for Obtaining Digital Signatures and Public-Key Cryptosystems.” *Communications of the ACM* 21 (2): 120–126.

Salomon, F. 2004. *The Cord Keepers: Khipus and Cultural Life in a Peruvian Village*. Durham, NC: Duke University Press.

Shady, R. 2006. “America’s First City? The Case of Late Archaic Caral.” In *Andean Archaeology III: North and South*, edited by W. H. Isbell and H. Silverman, 28–66. New York: Springer.

Shamir, A. 1979. “How to Share a Secret.” *Communications of the ACM* 22 (11): 612–613.

Splitstoser, J. C. 2014. “Practice and Meaning in Spiral-Wrapped Structured Strings (Khipus) from Cerrillos, a Late Paracas Site in the Ica Valley, Peru.” In *Textiles, Technical Practice, and Power in the Andes*, edited by D. Y. Arnold and P. Dransart, 47–73. London: Archetype Publications.

Splitstoser, J. C., T. D. Dillehay, J. Wouters, and A. Claro. 2016. “Early Pre-Hispanic Use of Indigo Blue in Peru.” *Science Advances* 2 (9): e1501623.

Szabo, N. 2005 [1998]. *Bit Gold*. Self-published essay.

Tehanetorens. 1999. *Wampum Belts of the Iroquois*. Summertown, TN: Book Publishing Company.

Urton, G. 1998. “From Knots to Narratives: Reconstructing the Art of Historical Record Keeping in the Andes from Spanish Transcriptions of Inka Khipus.” *Ethnohistory* 45 (3): 409–438.

Urton, G. 2003. *Signs of the Inka Khipu: Binary Coding in the Andean Knotted-String Records*. Austin: University of Texas Press.

Urton, G. 2017. *Inka History in Knots: Reading Khipus as Primary Sources*. Austin: University of Texas Press.

Urton, G., and C. J. Brezine. 2005. “Khipu Accounting in Ancient Peru.” *Science* 309 (5737): 1065–1067.

Williams, K. 2018. *Kayanerenkó:wa: The Great Law of Peace*. Winnipeg: University of Manitoba Press.

Appendix A, Quipu ↔ Datachain Rope Canonical Mapping (Extended)

For reference, the full structural mapping between khipu primitives and Datachain Rope primitives, as fixed in the Quipu Primitive Canon v1.1 (Datachain Foundation, 2026a, §6) and extended through v1.2 and v2.0:

Quipu primitive	Datachain Rope primitive	Realisation
Primary cord	Wallet *string* (per-wallet micro-ledger); federation *cord* at aggregation level	`kind = wallet` or `kind = cord` in the string registry
Pendant cord	Sub-stream of categorised events on a wallet string	Knots of related `event_type` on the same string
Subsidiary cord	Nested event hierarchies (child events of a parent attestation)	Linked knots via `parent_event_id` references
Top cord	Aggregation pendant (reduces over a set of pendants)	Cord-anchor knot (commits over a window of knots)
Knot	Event entry on a string	`event_id`, `event_type`, `event_payload`, `event_signature`
Knot type (overhand / long / figure-of-eight)	Event type	The `event_type` controlled vocabulary
Knot turn-count (size)	Event magnitude	`event_size` (token amount, payload size, attestation weight)
Knot position (decimal positional)	Knot index on a string	The `event_id` field
Knot direction (S / Z)	Cryptographic direction	Debit/credit, send/receive, claim/revoke
Untied knot (deliberate absence)	**Tombstone knot**	Result of `rope_untieKnot`
Cord colour	Event category tag	DCNFT, ERC-3643, claim topic, inference receipt, …
Cord ply / twist	Signature scheme variant	ML-DSA-65, SLH-DSA, …
Cord material	Wallet class	Datawallet+, Databox, institutional, sovereign
Quipucamayoq	Testimony witness	Per-category staked validator with reputational scoring
Imperial quipu (Cuzco)	Federation root cord	Anchored every ∼3s, generation cords every 86,400 anchors
Audit by re-recitation	Cross-validator testimony comparison	Quorum-based testimony aggregation per knot
Tribute aggregation upward	Anchor emission (knot reward)	DC FAT issued per cord-anchor knot, distributed by share

Appendix B, JSON-RPC Method Table (Canonical and Alias)

Datachain Rope preserves Ethereum-tooling compatibility through a permanent alias layer. The canonical methods are `rope_*`; the EVM aliases are `eth_*`. Wire compatibility is preserved indefinitely. This is the table:

Canonical method	EVM alias	Returns
`rope_knotIndex`	`eth_blockNumber`	Current cord-anchor knot index
`rope_getKnotByIndex(index, fullTxs)`	`eth_getBlockByNumber`	Knot at the given cord-anchor index
`rope_getKnotByHash(hash, fullTxs)`	`eth_getBlockByHash`	Knot with the given hash
`rope_appendToString(kind, id_bytes, payload)`	*no equivalent*	Append a knot to a per-entity string
`rope_walkString(kind, id, offset, limit)`	*no equivalent*	Read knots on a per-entity string
`rope_untieKnot(string_id, event_id)`	*no equivalent*	Untie a knot (granular erasure with tombstone)
`rope_globalStats()`	*no equivalent*	Per-kind string and knot counts; invariant assertion
`rope_listStrings(kind, offset, limit)`	*no equivalent*	List strings of a given kind
`rope_getString(kind, string_id)`	*no equivalent*	Descriptor for a specific string
`rope_getTombstone(string_id, event_id)`	*no equivalent*	Tombstone metadata for an erased knot
`rope_witnessTestimonies(witness_id, category)`	*no equivalent*	Testimony track record for a witness in a category
`rope_generationCord(generation_index)`	*no equivalent*	Generation-cord summary at a given index
`rope_federationCord()`	*no equivalent*	The current top-level federation cord state
`eth_call`, `eth_sendRawTransaction`, etc.	(preserved)	Standard Ethereum JSON-RPC for EVM-compatible operations

The methods marked *no equivalent* are the operations that have no analogue in the Ethereum block model. They are precisely the operations that realise the Andean cord-keeping primitives: per-entity append, per-entity walk, per-knot erasure, per-kind aggregation, per-witness reputational scoring, generation-cord federation. They are, on the argument of this paper, the operationally consequential additions that justify the substrate change.

Appendix C, Glossary

Anchor knot, A knot tied on the federation cord at periodic (∼3s) intervals, committing to the knot summaries of all per-entity strings that received a knot in the previous interval. The contemporary realisation of the Inka top-cord aggregation pattern.

Cord, In the khipu, the assemblage of strings tied to a primary cord, which is itself the spine of the assemblage. In Datachain Rope, the global federation-level structure that aggregates all per-entity strings.

CRYSTALS-Dilithium / ML-DSA, Module-Lattice-Based Digital Signature scheme, standardised by NIST in FIPS 204 (2024). Datachain Rope's default post-quantum signature primitive.

DC FAT, Datachain Foundation Asset Token, the native fungible utility token of Datachain Rope. Standard: DCR-20.

DCR-20, Datachain Rope native fungible token standard. Wire-compatible with ERC-20; semantically distinct as the native standard of this network.

ERC-3643, Token standard for permissioned, compliance-aware security tokens (T-REX). Used on Datachain Rope for tokenised real-world assets (Tanastok), preserved by name.

Federation Generation Protocol, The Datachain Rope mechanism for periodically generating *generation cords* that summarise the federation cord at higher-level granularity, producing a recursive aggregation hierarchy.

Genesis knot, The first knot on a string, marking the string's creation. Every string begins with a genesis knot, hence the substrate-level invariant `|strings|` $\leq$ `|knots|`.

Knot, In the khipu, a discrete tied configuration on a cord encoding numerical, categorical, or phonemic information. In Datachain Rope, the minimal information event tied at a position on a string, signed under post-quantum cryptography.

Knot witness, A node operator running rope-node who participates in the Testimony consensus protocol. The contemporary post-quantum *quipucamayoq*.

Khipu / Quipu, The Andean knotted-cord recording system used continuously from at least the Wari period (c. 600 CE) into the present, suppressed by the Council of Lima in 1583, partially preserved in Tupicocha. Quechua: "knot".

OES, Organic Encryption System, the layered cryptographic architecture of Datachain Rope that supplies per-knot encryption with per-knot keys, threshold-shredded across the testimony pool, irreversibly destructible on `rope_untieKnot` invocation.

Quipucamayoq, The Inka cord-keeper, professional administrator of the khipu, signer-of-record. Etymologically: Quechua "the one who keeps the knots".

Rope_untieKnot, The Datachain Rope JSON-RPC method that performs granular cryptographic erasure of a knot's payload, with tombstone preservation and federation anchoring of the erasure event.

SLH-DSA / SPHINCS+, Stateless Hash-Based Digital Signature scheme, standardised by NIST in FIPS 205 (2024). Datachain Rope's hash-based-only fallback signature primitive.

String, In Datachain Rope, the per-entity sequence of knots owned by a single ecosystem entity (wallet, contract, asset, DID, or cord). The contemporary realisation of the Andean sovereign thread.

String registry, The canonical Datachain Rope data structure tracking all strings by `(kind, id_bytes)`, exposing `rope_listStrings` and `rope_getString`. Enforces the invariant `|strings|` $\leq$ `|knots|`.

Testimony consensus, The Datachain Rope consensus protocol in which witnesses emit signed attestations on the validity of proposed knots, with per-category reputational scoring and per-cord-anchor finality.

Tombstone / Tombstone knot, A structurally-present absence on a string, marking that a knot was once at that position and was untied. Preserves audit-trail position; does not preserve content.

Tupicocha, A village in the Lima department of Peru, the only community known to have maintained continuous khipu practice into the present. Documented ethnographically by Frank Salomon (2004).

End of paper.