Blockchain as Neutral Infrastructure in a Regulatory Environment

Architectural Principles for Public-Sector Relevant Use Cases

Working Paper – Discussion Draft - Published by the LEA Blockchain Project – 17 January 2026

> *This document is a technical and conceptual discussion paper.*
>
> *It does not claim completeness, legal assessment, or scientific proof.*

Abstract

Digital infrastructures are increasingly situated at the intersection of technological innovation and regulatory reality. In particular, blockchain systems have often been designed under assumptions that insufficiently account for public-sector requirements such as clear responsibility, liability, and long-term adaptability.

This paper argues that regulatory viability depends less on individual rules than on fundamental architectural decisions. Many existing blockchain architectures embed execution logic and responsibility at the protocol level, creating global dependencies that hinder local regulatory adaptation.

The paper presents an alternative architectural model that combines a neutral consensus layer with sovereign, application-specific execution domains. Regulatory logic is consistently shifted to the application layer without affecting the stability of the underlying infrastructure.

Using selected public-sector relevant use cases, the paper demonstrates why such separation is a structural prerequisite for long-term institutional adoption.

1. Introduction: Public Digitalization Under Regulatory Constraints

The digitalization of public-sector processes follows different principles than technological innovation in private or entrepreneurial contexts. While startups and technology companies often proceed iteratively, defer regulatory questions, or consciously accept risk, public institutions are bound by legal stability, clear responsibility, and accountability.

Technologies intended for public use must not only be functional but structurally resilient over time. They must support liability attribution, ensure data protection, remain maintainable in the long term, and integrate into existing institutional frameworks. These requirements are not optional and cannot be retrofitted without fundamentally altering the system.

Blockchain technologies are often discussed in political debates either as a panacea or as a risk vector. Both perspectives fall short. The decisive factor is not whether blockchain is suitable in principle, but which type of blockchain architecture is compatible with public-sector constraints.

This paper argues that many conflicts between blockchain systems and regulation do not arise from legal details but from technical design decisions that carry implicit political and regulatory consequences.

2. Structural Tension: Global Systems, National Rules

Legal systems are organized nationally or regionally. Jurisdiction, tax sovereignty, data protection requirements, and liability frameworks vary not only between states but also within federal systems. Digital infrastructures, by contrast, are often designed globally, operated uniformly, and technically difficult to fragment.

Blockchain systems intensify this tension. By definition, they are cross-border, consensus-driven, and designed to enforce uniform rules for all participants. What is technically advantageous quickly becomes problematic in a regulatory context.

This is particularly the case when:

- execution logic is globally fixed,
- economic mechanisms apply uniformly across applications,
- security assumptions can only be modified centrally.

Such conditions create dependencies that make local regulatory adaptation difficult or impossible.

Attempts to resolve these issues through downstream governance processes, exemptions, or legal constructs often result in uncertainty. Technical changes become politicized, regulatory interventions affect the entire system, and individual applications may impose systemic consequences on the network as a whole.

A central interim conclusion therefore follows:

The more rules are technically embedded at the protocol level, the higher the regulatory risk of the overall system.

3. Limitations of Existing Blockchain Architectures

3.1 Monolithic Architectures

In monolithic blockchain systems, consensus, execution, security mechanisms, and economic logic are tightly coupled. Changes to any of these elements inevitably affect the entire network.

For public or regulated use cases, this creates several issues:

- legal adaptations require protocol changes,
- governance decisions become systemic risks,
- heterogeneous regulatory requirements cannot coexist in parallel.

As a result, individual applications may implicitly impose regulatory consequences on all other participants, increasing both complexity and legal uncertainty.

3.2 Modular Stacks and Rollup Models

Modular blockchain approaches have achieved important progress, particularly in scalability. By separating data availability, settlement, and execution, technical bottlenecks can be reduced.

From a regulatory perspective, however, key challenges remain:

- execution logic remains globally defined within a rollup and applies uniformly to all applications,
- responsibilities are shifted to additional layers,
- regulatory adaptation occurs outside the base system.

Rather than structurally integrating regulation, it is often displaced into separate layers, organizations, or legal constructs. This may reduce short-term friction but does not resolve the underlying issue of clear technical responsibility.

3.3 Conclusion

Both monolithic and contemporary modular architectures primarily treat regulation as an external concern. The technical architecture itself remains unchanged, forcing regulatory requirements into downstream structures.

For public-sector relevant use cases, this approach is not sustainable. It creates complex dependencies, unpredictable risk, and high barriers to institutional adoption.

This makes clear that an alternative approach is required, one in which regulation is not retrofitted but architecturally enabled.

4. Architecture as a Regulatory Factor

Regulatory questions are often treated in technical discussions as downstream concerns. Systems are built first, with legal requirements addressed later through additional rules, governance processes, or external controls. This perspective overlooks the fact that technical architecture itself already exerts normative force.

Architecture determines whether responsibilities can be clearly separated or whether changes inevitably have system-wide effects. In this sense, architecture is not a neutral background but a structural framework for regulatory agency.

In blockchain systems, this relationship becomes particularly evident. When execution logic, security assumptions, and economic rules are fixed at the protocol level, regulatory adaptation necessarily requires intervention at the core of the system. Each change creates uncertainty for all participants, regardless of whether they are affected by the specific use case.

Such designs generate two problematic effects. First, regulatory requirements are perceived as risks to network stability. Second, political pressure emerges to slow or halt technical evolution. Both outcomes conflict with the needs of public digitalization, which depends on long-term stability and controlled adaptability.

A regulatory-viable architecture must therefore satisfy three conditions. It must remain neutral with respect to application logic. It must clearly separate responsibilities. And it must enable local adaptation without triggering global effects.

These conditions cannot be achieved through policy or governance alone. They are the result of deliberate architectural separation between order and meaning, between infrastructure and application.

5. Reference Architecture: Neutral Consensus and Sovereign Execution Domains

Based on these requirements, a reference architecture can be defined that structurally enables public-sector relevant use cases rather than constraining them. The core concept is the strict separation between a neutral base system and application-specific logic.

The consensus layer fulfills exclusively infrastructural functions. It ensures unambiguous transaction ordering, finality, and data availability. It interprets no content, evaluates no rules, and contains no application logic. In this sense, it acts as a neutral ordering layer.

All substantive decisions are made within separate execution domains. These domains are application-specific, technically isolated, and bear full responsibility for their rules. They define transaction validity, accepted signatures, fee structures, and applicable regulatory requirements.

Within the LEA blockchain, this model is implemented through so-called PODs. PODs (Programmable Object Domains) are sovereign execution domains that encapsulate application logic, validation rules, and responsibility within clearly delineated technical units.

These execution domains are sovereign in a technical sense and can be independently designed, modified, and evolved. Changes within one domain do not affect other domains or the consensus layer. This enables local adaptability without introducing global instability.

Such a model allows highly diverse use cases to coexist on the same infrastructure. Regulation-intensive applications can implement strict requirements, while other domains remain intentionally open. Both rely on the same ordering layer and finality guarantees without interfering with one another.

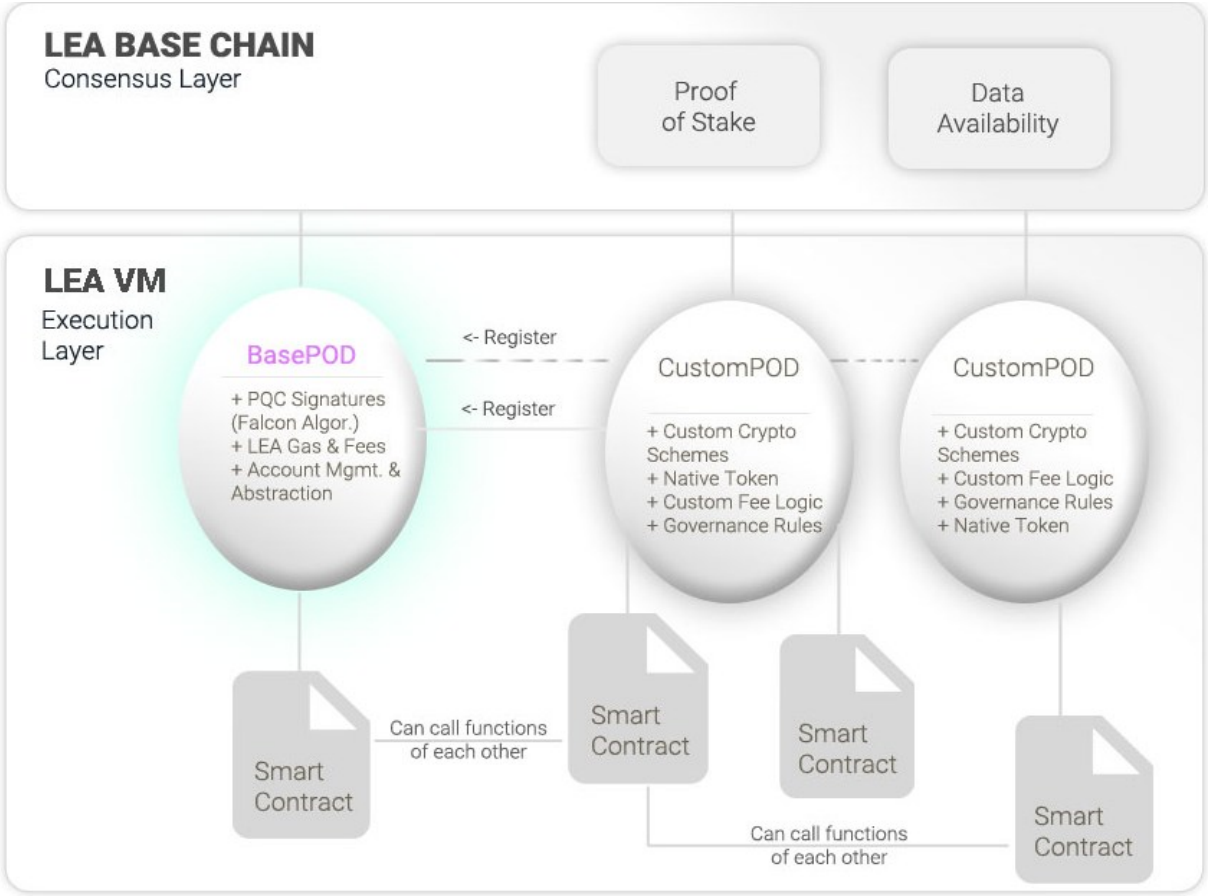

Image 1 LEA Rough architecture und Programmable Object Domains (PODs)

This architecture establishes clear responsibility. The operator of an execution domain is responsible for its rules. Regulatory requirements can be clearly attributed, technically enforced, and modified as needed. The base system remains unaffected and retains its neutrality.

The described reference architecture is concretely realized within the LEA blockchain. LEA implements a minimal consensus layer responsible solely for ordering and availability. All execution logic is delegated to clearly separated execution domains operated as independent technical units.

Crucially, LEA does not impose regulatory prescriptions. The system enforces neither openness nor restriction. It merely provides the structural conditions required to implement regulation where it belongs: at the application layer.

This approach fundamentally differs from systems that attempt to address regulatory concerns through global rules or retrospective constraints. Regulation is instead treated as a legitimate attribute of individual application domains without altering the character of the infrastructure.

6. Public-Sector Relevant Use Cases as Architectural Stress Tests

The robustness of a technical architecture is not demonstrated by abstract performance metrics but by its suitability for real institutional challenges. This chapter examines five use cases of current political and administrative relevance. The objective is not to evaluate specific policies but to assess architectural requirements.

Each subsection follows the same structure: description of the underlying problem, derivation of architectural requirements, and assessment of how a neutral-consensus-plus-sovereign-domains model can address them.

6.1 Value Added Tax and Tax Fraud

VAT fraud represents one of the largest recurring sources of fiscal loss. The mechanisms are well known. Complex supply chains, cross-border transactions, and delayed audits create opportunities for manipulation. A core issue is that tax-relevant information is fragmented and often only reviewed ex post.

Architecturally, this yields a clear requirement. Tax-relevant events must be recorded in a traceable, time-consistent, and machine-readable manner without exposing trade secrets or personal data. Responsibility for correctness must be clearly attributable.

A sovereign execution domain can address this requirement by recording only tax-relevant events rather than full business processes. Invoices, deliveries, or service milestones are anchored as verifiable references, while substantive details remain outside the infrastructure. Validation rules can be implemented directly within execution logic without affecting other applications.

The critical distinction is between infrastructure and tax logic. The consensus layer merely ensures ordering and immutability. Tax interpretation remains entirely within the respective domain and its operator.

6.2 Verifiable Facts in the Context of Increasing AI Use

As generative AI systems proliferate, societal focus shifts from content creation to verifiable provenance. For public institutions, the task is not to assess content but to establish when and under what conditions information was produced and whether it has been altered.

Architecturally, this requires separating content from integrity proof. Systems that conflate both introduce either privacy risks or unnecessary disclosure.

A dedicated execution domain can function as an integrity layer. Documents, statements, or AI-generated outputs are cryptographically referenced and time-stamped without exposing content. Multiple trusted issuers or validators can participate without central authority.

Again, the consensus layer remains neutral. It does not judge truth or relevance. It merely ensures temporal ordering and immutability. Meaning is defined exclusively at the application layer.

6.3 Emerging AI Economies and Tax Attribution

Autonomous or semi-autonomous AI systems increasingly generate revenue, trigger contracts, or deliver services. Existing tax frameworks are only partially equipped to address such arrangements.

From a technical perspective, the immediate requirement is clear attribution. Who operates the system, who receives revenue, and who bears tax responsibility. If these questions cannot be addressed technically, future regulation becomes significantly more difficult.

A sovereign execution domain enables AI-driven processes to be clearly assigned to an operator or legal entity. Revenue flows are technically traceable without imposing legal interpretation at the infrastructure level. Regulatory changes can be implemented within the domain without modifying the base system.

This does not solve regulatory questions but removes structural barriers. The technology does not obstruct the application of law; it enables it.

6.4 Development Cooperation and Public Transfers

In development cooperation, transparency, traceability, and cost control are critical. At the same time, legitimate security concerns limit disclosure of operational detail. Traditional systems often result in either opacity or new dependencies on central actors.

Architecturally, the challenge is to ensure traceable fund movements without revealing sensitive context. Purpose binding must be verifiable without restricting operational flexibility.

A dedicated execution domain can represent transfers as verifiable state changes. Allocations, earmarking, and disbursement become auditable, while operational details remain external. Roles can be clearly separated without requiring centralized oversight.

The consensus layer again functions solely as a neutral ordering layer. No political or operational judgments are made.

6.5 Digital Credentials Between Usability and Data Protection

Digital credentials such as certificates, diplomas, or authorizations are increasingly important. At the same time, strict requirements exist regarding data protection, revocability, and purpose limitation. Immutable storage of personal data is incompatible with these constraints.

The architectural solution lies in separating proof from content. An execution domain can attest to the existence, validity, or revocation of a credential without exposing personal data. Validity periods, updates, and revocations can be technically represented.

Here as well, the consensus layer has no knowledge of content. It ensures integrity of state changes only. Data protection is not bypassed but architecturally respected.

Interim Conclusion

Although the examined use cases differ substantially, they share a common requirement: regulatory logic must be implementable

locally without compromising shared infrastructure stability.

An architecture based on neutral consensus and sovereign execution domains fulfills this requirement structurally. It neither replaces legal processes nor makes political decisions. It provides the technical precondition for realistically implementing public-sector relevant applications.

7. Deliberate Scope Limitations and Non-Goals

Clear boundaries are necessary to avoid misinterpretation and to realistically position the role of technical infrastructure. The architecture described here is explicitly not a legal, political, or administrative instrument, but a technical foundation that enables or constrains certain actions.

The architecture makes no legal judgments. It does not determine applicable rules but provides the structural conditions to implement diverse rules in a technically coherent manner. Responsibility for substantive requirements remains with domain operators and competent authorities.

It does not replace public-sector processes. Auditing, supervision, approval, and enforcement remain institutional tasks. Infrastructure may support but not substitute them.

The architecture is not an instrument of political control. The neutral consensus layer evaluates no content, prioritizes no applications, and intervenes in no execution logic. Its content-agnostic design ensures long-term public usability.

Finally, the architecture is not a guarantee of compliance. It prevents neither misuse nor governance failure. It reduces structural risk by localizing regulatory adaptation and avoiding system-wide effects.

These deliberate non-goals are essential for properly understanding the architecture and its long-term applicability.

8. Blockchain as Public Infrastructure

Public infrastructures share certain characteristics. They must be stable, durable, and independent of individual use cases. At the same time, they must remain flexible enough to accommodate new requirements without constant structural modification.

Historical examples demonstrate that infrastructures fail when overly tailored to specific scenarios. Successful systems maintain clear separation between infrastructure and use.

Applied to blockchain systems, it becomes evident that many existing architectures do not consistently fulfill their infrastructural role. By coupling order with substantive assumptions, they become vulnerable to political and regulatory change. A clearly separated infrastructure avoids this coupling and remains adaptable.

For public contexts, this distinction is decisive. Only a neutral infrastructure can operate across differing legal and political frameworks. Only localized adaptation ensures long-term viability.

In this sense, the architecture described here represents less a technological innovation than a structural prerequisite for sustainable use.

9. Conclusion

Discussions around blockchain and public-sector adoption often focus on individual use cases or regulatory details. This paper advances a different perspective, arguing that the decisive question lies at the architectural level.

Technical systems determine which forms of regulation are feasible. When execution logic is globally embedded, regulation becomes a systemic risk. When responsibility cannot be clearly separated, uncertainty arises. When adaptation requires intervention at the core, innovation becomes politicized.

An architecture that consistently separates order from execution addresses these challenges structurally. It enables regulatory diversity without compromising infrastructure stability. It establishes clear responsibility without imposing central control. And it allows innovation without undermining institutional requirements.

The question is therefore not whether blockchain can be used in public contexts. The question is which architectures make such use possible.

Glossary

The following glossary clarifies key terms as they are used in this document. It does not claim to provide generally applicable definitions beyond this context.

Application Logic	Rules and validation mechanisms defined within an execution domain that determine its functional purpose. Application logic is separated from the consensus layer.
Execution Domain	A clearly delineated technical application space within a blockchain architecture with its own logic and responsibility.
Blockchain	A digital infrastructure for ordered and immutable documentation of state changes. In this document, understood as a neutral ordering layer rather than an application or financial instrument.
Finality	The state in which a transaction or state change is considered irreversible.
Integrity Proof	A technical attestation that a given state, document, or piece of information existed at a specific time and has not been altered.
Consensus Layer	The base layer of a blockchain system responsible for ordering, finality, and data availability, without interpreting application logic.
Infrastructure Neutrality	The principle that the base layer of a system makes no substantive or regulatory assumptions and does not privilege applications.
PODs (Programmable Object Domains)	Project-specific term for sovereign execution domains within the LEA architecture, encapsulating application logic and responsibility.
Regulatory Viability	The structural capability of a system to accommodate regulatory requirements without altering core infrastructure.

Rollup	A separate execution system in which transactions are processed off-chain and aggregated onto a base chain, primarily for scalability.
Execution Domain Sovereignty	The ability of an execution domain or POD to modify its rules independently of the base system.
Transaction Ordering	The unambiguous temporal sequence of state changes provided by the consensus layer.
Responsibility	Clearly attributable accountability for rules and decisions within an execution domain.

References and Further Reading

Lessig, Lawrence (1999).
Code and Other Laws of Cyberspace. Harvard University Press.

Yeung, Karen (2018).
Algorithmic regulation: A critical interrogation.
Regulation & Governance, 12(4), 505–523.

Brownsword, Roger (2019).
Law, Technology and Society. Routledge.

Werbach, Kevin (2018).
The Blockchain and the New Architecture of Trust. MIT Press.

De Filippi, Primavera; Wright, Aaron (2018).
Blockchain and the Law: The Rule of Code. Harvard University Press.

Buterin, Vitalik (2017).
On-chain governance: Why it's hard. Ethereum Research.

Buterin, Vitalik (2021).
A rollup-centric Ethereum roadmap. Ethereum Foundation.

Al-Bassam, Mustafa et al. (2018).
Fraud proofs: Maximising light client security and scaling blockchains with dishonest majorities.
IEEE Symposium on Security and Privacy.

OECD (2017).
Technology Tools to Tackle Tax Evasion and Tax Fraud. OECD Publishing.

OECD (2020).
The OECD Digital Government Policy Framework. OECD Publishing.

European Commission (2019).
VAT Fraud and the Digital Economy. Publications Office of the European Union.

Finck, Michèle (2018).
Blockchain and the General Data Protection Regulation.
European Parliament Study.

W3C (2022).
Decentralized Identifiers (DIDs) and Verifiable Credentials.
World Wide Web Consortium Recommendation.

World Economic Forum (2020).
Blockchain Deployment Toolkit for Governments.

Mazzucato, Mariana (2018).
The Value of Everything. Penguin Random House.

LEA

LEA is a blockchain infrastructure project focused on modular execution domains and regulatory viability. The project develops a neutral consensus layer for public and institutional use cases.

Website: https://getlea.org
Kontakt: info@getlea.org
Version: Discussion Draft 1.0
Date: Januar 2026