



[Home](#) [Licensing](#) [Patents](#) [Articles](#)

## **Zeebe Scaled Workflow Orchestration Horizontally. Governance Did Not Scale With It.**

by [Nick Clark](#) | Published March 28, 2026 | [PDF](#)

Zeebe, built by Camunda, provides horizontally scalable workflow orchestration through partitioned, replicated event sourcing. It handles millions of workflow instances concurrently across partitions. The scaling architecture is sound. But Zeebe scales execution throughput without scaling governance. Each workflow instance executes tasks without semantic agent governance, trust slope validation, or cryptographically bound policy enforcement. More throughput means more ungoverned execution. The structural gap is between scalable orchestration and governance that scales alongside the execution it governs.

---

Zeebe's event-sourced, partitioned architecture provides genuine horizontal scaling for workflow orchestration. The gap described here is about governance scaling, not about throughput scaling.

## Partitioned execution without partitioned governance

Zeebe partitions workflow instances across brokers for horizontal scaling. Each partition processes workflow steps independently. But governance is not partitioned. There is no mechanism for different partitions to apply different governance policies, validate trust slopes at partition boundaries, or ensure governance continuity across partitions.

Scaling the execution without scaling the governance means that governance becomes proportionally less present as throughput increases. More workflows execute. Each workflow receives the same absence of governance.

## Event sourcing without governance events

Zeebe uses event sourcing to record workflow state transitions. This provides auditability and replay capability. But the event stream records what happened. It does not record governance validation at each step. There are no governance events: no record that trust was validated, no record that policy was evaluated, no lineage connecting each step to the governance that authorized it.

## What a cognition-native execution platform provides

A cognition-native execution platform would scale governance alongside execution. Each partition would have governance anchors applying scope-local policy. The event stream would include governance events recording policy evaluation at every step. Trust slope validation would be a partitioned operation, scaling horizontally with the workflow throughput it governs.

[Execution Platform All 21 steps →](#)

The complete runtime for governed, persistent agents.

Patent

[US 19/230,933](#) · filed

Primary Technical Disclosure

[◦ A Cognition-Native Execution Platform for Distributed, Stateful, and Governable Agents](#)

Secondary Technical

[◦ Six-Field Canonical Agent Schema: Structural Definition of Autonomous Semantic Agents](#)[◦ Semantic Nest Instantiation: Dynamic Execution Environments From Agent Density and Entropy](#)[◦ Trust Zone Overlay Governance: Logical Policy Domains Independent of Network Topology](#)[◦ Scoped Quorum Mutation Validation: Independent Validators With Meta-Policy Escalation](#)[◦ Meta-Policy Override Resolution: Higher-Level Governance for Local Quorum Decisions](#)[◦ Semantic Router: Schema-Aware Propagation Replacing Address-Based Forwarding](#)[◦ Dynamic Agent Hash Derivation: Deterministic Identity From Memory and Mutation History](#)[◦ Dynamic Device Hash Derivation: Substrate Identity From Device-Local Entropy](#)[◦ Content Anchor Hash Derivation: Perceptual Identity for Non-Executing Digital Content](#)[◦ DAH-DDH Slope Entanglement: Binding Agent Identity to Host Device Lineage](#)[◦ Trust Slope Validation Across Zone Migration: Continuity Verification With Quarantine](#)[◦ Pseudonymous Propagation: Recognition by Slope Rather Than Global Identifier](#)[◦ Alias Slope-Band Indexing: Symbolic Resolution Through Trust-Slope Pathfinding](#)[◦ Fallback Rehydration: Recovering Partial Agents Through Contextual Policy Inference](#)[◦ Structural Validator With Fallback Routing: Schema Verification Before Execution](#)[◦ Execution Graph Manager: Structured Lineage of Agent Reasoning and Transformation](#)[◦ Full and Partial Agent Interoperability: Cross-Boundary Semantic Exchange Under Policy](#)[◦ Cross-Topology Substrate Deployment: Identical Agent Structure Across All Substrates](#)

Applications (General)

[◦ Multi-Cloud Agent Orchestration Without Centralized Schedulers](#)[◦ Autonomous Fleet Coordination Through Self-Governing Agents](#)[◦ Enterprise Workflow Without Orchestration Servers](#)[◦ Smart Contract Execution Without Blockchain Latency](#)[◦ Distributed Scientific Computing With Governed Agents](#)[◦ Supply Chain Autonomous Agents](#)[◦ Energy Grid Management Through Autonomous Agents](#)[◦ Disaster Response Coordination Without Central Command](#)

Applications (Specific)

[◦ Kubernetes Orchestrates Containers. It Does Not Know What They Are Doing](#)[◦ Temporal Solved Durable Workflows. The Workflows Have No Semantic Identity](#)[◦ Apache Airflow Orchestrates DAGs. The Tasks Inside Them Are Ungoverned](#)[◦ Prefect Made Data Workflows Pythonic. The Execution Model Is Still Task Scheduling](#)[◦ AWS Step Functions Made Serverless Orchestration Visual. The Steps Have No Semantic State](#)[◦ Azure Durable Functions Made Stateful Serverless Possible. The State Has No Governance](#)[◦ Nomad Schedules Any Workload. It Does Not Know What Those Workloads Are](#)[◦ Docker Swarm Simplified Container Orchestration. The Containers Are Still Opaque](#)[◦ Apache Mesos Managed Datacenter Resources. The Resources Had No Semantic Governance](#)[◦ Argo Workflows Orchestrates Kubernetes-Native Pipelines. The Pipeline Steps Have No Governance](#)[◦ Dagster Made Data Pipelines Software-Defined. The Pipeline Has No Governance Substrate](#)[◦ Luigi Defined Task Dependencies for Data Pipelines. The Tasks Execute Without Governance](#)[◦ Camunda Orchestrates Business Processes. The Process Engine Has No Semantic Agent Governance](#)[● Zeebe Scaled Workflow Orchestration Horizontally. Governance Did Not Scale With It.](#)

[Execution Platform overview →](#)

AQ

deterministic

autonomy

Legal

Subject to one or more pending U.S. and international patent applications, see [Patents](#) for the current list and status. No license, express or implied, is granted. Any use requires a separate written agreement—see [Licensing](#). Patent applications referenced on this site are pending. Claim scope, if any, is subject to examination and may issue in altered form or not at all. See [Legal](#) for terms and conditions.

Adaptive Query™ is a trademark of Nicholas Clark. U.S. federal registration is pending. federal registration. AQ™, AQ Inside™, Adaptive Index™, Adaptive Network™, Semantic Agent™, @AQ™, AQID™, and Adaptive Coin™ are used as trademarks in connection with the Adaptive Query platform and brand. Other names may be trademarks of their respective owners.

Platform operated by Adaptive Query LLC, which provides patent and trademark licensing services. Copyright © 2025-2026 Nicholas Clark. All rights reserved.



- [Inventive Steps](#)
- [Licensing](#)
- [Patents](#)
- [Articles](#)
- [Legal](#)
- [Opportunities](#)
- [Sitemap](#)



- 
- [nick@qu3ry.net](mailto:nick@qu3ry.net)
- 72 28 14 36 01



[Invented by Nick Clark](#) | Founding Investors: Devin Wilkie