Welcome Sudarshana Bandyopadhyay Sign out

Controller General of Patents, Designs & Trade Marks

सत्यमेव जयते

INTELLECTUAL PROPERTY INDIA
PATENTS | DESIGNS | TRADE MARKS
GEOGRAPHICAL INDICATIONS

G.A.R.6
[See Rule 22(1)]
RECEIPT

Docket No 22445

Date/Time 2025/09/30 22:27:10

To
Sudarshana Bandyopadhyay

UserId: SB2802

Flat No. 91, Sector A, Pocket C, Vasant
Kunj, New Delhi - 110070, India

CBR Detail:

Sr. No.	App. Number	Ref. No./Application No.	Amount Paid	C.B.R. No.	Form Name	Remarks
1	202531094304	TEMP/E-1/105450/2025-KOL	6560	12210	FORM 1	CAUSAL MECHANISM-BASED DISTRIBUTED COMPUTING ARCHITECTURE FOR EDGE AND FEDERATED ENVIRONMENTS
2	E-12/1940/2025/KOL	202531094304	2500	12210	FORM 9	----
3	E-106/2802/2025/KOL	202531094304	0	-----	FORM28	----

TransactionID	Payment Mode	Challan Identification Number	Amount Paid	Head of A/C No
N-0001762973	Online Bank Transfer	3009250083228	9060.00	1475001020000001

Total Amount : ₹ 9060.00

Amount in Words: Rupees Nine Thousand Sixty Only

Received from Sudarshana Bandyopadhyay the sum of ₹ 9060.00 on account of Payment of fee for above mentioned Application/Forms.

* This is a computer generated receipt, hence no signature required.

Print

Home About Us Contact Us

(12) PATENT APPLICATION PUBLICATION

(21) Application No.202531094304 A

(19) INDIA

(22) Date of filing of Application :30/09/2025

(43) Publication Date : 10/10/2025

(54) Title of the invention : CAUSAL MECHANISM-BASED DISTRIBUTED COMPUTING ARCHITECTURE FOR EDGE AND FEDERATED ENVIRONMENTS

(51) International classification	:G06F0021620000, H04L0009320000, G06N0020000000, G06N0007010000, H04L0009000000
(31) Priority Document No	:NA
(32) Priority Date	:NA
(33) Name of priority country	:NA
(86) International Application No	:
Filing Date	:01/01/1900
(87) International Publication No	: NA
(61) Patent of Addition to Application Number	:NA
Filing Date	:NA
(62) Divisional to Application Number	:NA
Filing Date	:NA

(71)**Name of Applicant :**
1)SRJX RESEARCH AND INNOVATION LAB LLP
Address of Applicant :Plot No - 3E/474, Sector-9, CDA, Post- Markat Nagar, Cuttack-753014, Odisha, India Cuttack Orissa India
(72)**Name of Inventor :**
1)DR SOUMYA RANJAN JENA
2)MR SANJOY SAHA
3)DR SOHIT AGARWAL

(57) Abstract :

The present invention provides a distributed computing architecture for real-time counterfactual reasoning and intervention-aware analytics across edge, gateway, and cloud nodes. The system elevates causal mechanisms to first-class cacheable units, where each mechanism kernel encapsulates a predictive function, stochastic noise model, invariance certificate, and manifest detailing inputs, training metadata, privacy state, and applicability contexts. Consistency across nodes is maintained by an intervention-aware coherence protocol that propagates updates as compact mechanism deltas keyed to interventions and invariance predicates, avoiding global cache flushes. A mechanism capability signature enables nodes to pre-check whether queries can be executed locally or require specific updates, facilitating low-bandwidth, secure federation. Counterfactual computation is performed by an Intervention Execution Unit that composes kernels, simulates proposed interventions, and returns outcomes with calibrated uncertainty and verifiable provenance. A selector optimizes reuse, local relearning, or delta fetching under constraints of latency, accuracy, energy, bandwidth, and privacy budgets. Privacy and compliance are enforced through purpose bindings, differential privacy ledgers, and cryptographically auditable proof-of-use receipts. The invention reduces data movement, supports heterogeneous hardware, and establishes a certified kernel ecosystem, providing an auditable, efficient, and privacy-preserving substrate for distributed “what-if” computation.

No. of Pages : 37 No. of Claims : 22

FORM 2

THE PATENTS ACT, 1970

(39 OF 1970)

AND

THE PATENTS RULES, 2003

COMPLETE SPECIFICATION

(*See* section 10; rule 13)

1. TITLE OF THE INVENTION

CAUSAL MECHANISM-BASED DISTRIBUTED COMPUTING ARCHITECTURE FOR EDGE AND FEDERATED ENVIRONMENTS

2. APPLICANT

(a) NAME: SRJX RESEARCH AND INNOVATION LAB LLP

(b) NATIONALITY: India

(c) ADDRESS: Plot No - 3E/474, Sector-9, CDA, Post- Markat Nagar, Cuttack-
753014, Odisha, India

3. PREAMBLE TO THE DESCRIPTION

The following specification particularly describes the invention and the manner in which it is to be performed.

FIELD OF THE INVENTION

The present invention relates to the field of distributed and edge computing systems, and more particularly to architectures, methods, and protocols for enabling counterfactual reasoning and causal inference in such systems. The invention specifically addresses the representation, caching, synchronization, and execution of causal mechanisms as first-class computational artifacts in edge–cloud environments, thereby supporting intervention-aware coherence, privacy-preserving portability, and efficient counterfactual-native computation under stringent latency, bandwidth, energy, and regulatory constraints.

BACKGROUND OF THE INVENTION

Modern distributed computing environments increasingly rely on edge devices and micro-datacenters to meet stringent latency, privacy, and cost requirements. Sensors, mobile phones, industrial controllers, and vehicular systems generate continuous observation streams that must be acted upon in milliseconds, often under regulatory restrictions that limit raw data movement. Conventional distributed architectures address scalability by centralizing analytics and machine learning in cloud platforms while pushing lightweight caches or model runtimes to the edge. Although these approaches reduce bandwidth and improve responsiveness for repeated lookups, they remain fundamentally designed for serving static content or point predictions rather than enabling reasoning about alternative decisions or policies.

Conventional caches and content delivery networks are optimized for byte-level reuse: they store files, blocks, or precomputed predictions and index them by URLs, addresses, or version identifiers. This paradigm works for static content and deterministic lookups, but it breaks down when a user asks counterfactual “what-if” questions that require changing the underlying data-generating process. Because these caches do not represent or manage causal mechanisms, they cannot update only the pieces of logic that change when an intervention is proposed. Coherence is enforced with coarse invalidations tied to object versions rather than to the specific assumptions that make a cached relationship valid. The result is either stale answers when conditions shift or costly, system-wide cache flushes that defeat the purpose of caching.

Model-serving stacks and feature stores extend caching to machine learning, but they retain the same limitations. They cache model binaries, embeddings, or inference outputs to reduce latency and compute, yet they lack semantics for interventions. When a pricing policy changes, a treatment guideline is updated, or a device firmware alters control loops, previously cached predictions become unreliable in ways version numbers cannot capture. Operators are forced into blunt strategies: retrain everything, redeploy full models, or accept degraded performance until nightly jobs run. This wastes bandwidth and energy, increases time-to-correctness, and provides no principled way to decide whether a local artifact remains valid for a new decision context.

Federated learning and on-device inference aim to keep raw data local, but they move large model updates and gradients that implicitly entangle many mechanisms at once. These updates provide no explicit handle on which causal relationships are stable across domains and which are context-specific. Without explicit invariance certificates or compact capability descriptors, a device cannot determine whether its local model is adequate to answer a particular intervention query. Moreover, gradient sharing can still leak information, and cross-device heterogeneity means the same global update may overfit some contexts and underfit others. The absence of mechanism-level modularity prevents shipping only the minimal change needed to adapt to a local intervention.

Privacy-enhancing technologies such as differential privacy, secure enclaves, and multiparty computation address data exposure but not intervention-centric computation. Differential privacy protects releases but adds noise that may be misallocated when only part of a mechanism changes, and it does not indicate whether a learned relationship still holds in a new environment. Enclaves centralize execution and require data ingress or trust in hardware roots, often conflicting with data localization rules and energy budgets. Cryptographic protocols can be too heavy for edge devices and still lack a representation for causal kernels as portable, auditable units. None of these approaches defines a coherence notion that tracks validity under specific interventions rather than under generic version drift.

Causal inference offers tools for counterfactual reasoning by modelling domains as structural causal models whose mechanisms describe how variables influence each other. Yet in practice, causal tooling such as DAG learning, treatment effect estimation, and abduction–action–prediction pipelines are typically applied offline and monolithically. Analysts produce reports or export a whole-model artifact for batch evaluation. There is no standard to package an individual structural equation with its parents, uncertainty, and invariance scope as a small, deployable unit. Without such packaging, systems cannot perform targeted invalidation when an invariance breaks, nor can they exchange compact parameter deltas for only the affected mechanism. As a consequence, organizations either move raw data to centralized causal engines, inflating latency and privacy risk, or avoid counterfactual analytics entirely at the edge where decisions are made.

Digital twins and domain simulators promise “what-if” analysis, but they are often heavyweight, domain-specific, and centrally maintained. Updating a twin typically means applying a large configuration or model patch, not a minimal mechanism changes with provenance suitable for distribution across thousands of heterogeneous nodes. Their licensing and deployment models assume stable connectivity and ample compute, making them ill-suited to intermittent, low-power edge environments. Critically, digital twins rarely expose machine-verifiable invariance scopes or privacy guarantees for their internal mechanisms, making it difficult to share or reuse pieces across organizational or jurisdictional boundaries while satisfying audit requirements.

Data engineering and MLOps practices handle drift and schema evolution with coarse tools such as dataset versioning, concept-drift alarms, and shadow evaluations. These practices detect that “something changed” but seldom identify which structural relationship broke and only invalidate or relearn that part. The operational response is expensive: retrain pipelines, rebuild features, or push a full model update. This overreaction consumes bandwidth, compute, and energy—costs that are increasingly constrained by carbon targets. It also prolongs periods where production systems answer with stale assumptions because finely targeted repairs are impossible. The lack of mechanism-aware invalidation creates a persistent gap between data monitoring and decision-time correctness.

Existing coherence protocols whether for CPU caches, distributed key–value stores, or model registries synchronize based on addresses, keys, or opaque version vectors. They do not encode dependency graphs between mechanisms or attach validity to intervention descriptors (e.g., “policy P toggled variable X to value x”). As a result, systems cannot reason about partial correctness: a cache line or model version is either valid or not, forcing conservative invalidations. There is no standard way to propagate mechanism deltas whose scope is limited to an identifiable subgraph of the causal model. This absence of intervention-aware coherence causes unnecessary recomputation and bandwidth use and increases the risk of subtle, lingering inconsistencies.

Auditability and regulatory compliance in current stacks are centered on access logs and model version histories, not on verifiable proofs that only non-sensitive mechanisms moved and that queries were answered within a declared capability. In environments subject to data localization, health privacy, or industrial secrecy, it is not enough to say that “no raw data left the device”; systems must demonstrate that only approved mechanism kernels were disseminated and used. Prior art generally lacks purpose binding, geographic or temporal use constraints on model artifacts, and signed proof-of-use receipts that bind a specific intervention query to a specific cached mechanism. This gap undermines trust, complicates vendor collaboration, and slows adoption in regulated sectors.

Operational constraints at the edge—heterogeneous hardware, intermittent connectivity, strict latency targets, and tight power budgets—exacerbate all of the above. Centralized recomputation is slow or impossible when links are down; full-model updates are too heavy; and rule-based selection between local reuse, relearning, or remote fetch is myopic, wasting energy or missing deadlines. Prior systems do not optimize the multi-objective trade-off among latency, accuracy, privacy, and energy at mechanism granularity. Without a learned selector informed by mechanism metadata and drift signals, devices either overfetch from the network, burning joules, or over-recompute locally, missing real-time constraints.

Earlier inventions give us byte caches, predictive serving, federated updates, privacy wrappers, and offline causal analyses but they do not provide a way to represent causal mechanisms as portable, certifiable units; to synchronize their validity under specific interventions; to exchange only minimal deltas when a small part changes; or to produce auditable evidence that counterfactual queries were answered within local capability without exposing data. This constellation of shortcomings leads to stale or non-counterfactual answers, unnecessary data movement, higher energy consumption, brittle compliance postures, and slow recovery from change. Accordingly, there is a need for a computing architecture that seamlessly treats causal mechanisms as cacheable and portable units, validates their coherence under interventions, and operates under strict privacy and compliance constraints while minimizing recomputation, latency and energy use.

OBJECTS OF THE INVENTION

The primary object of the present invention is to provide a distributed computing architecture that enables real-time counterfactual reasoning and intervention-aware analytics at the edge, thereby reducing reliance on centralized data transfer and large model redeployment.

Another object of the invention is to encapsulate domain relationships into compact, portable, and auditable “mechanism kernels” with rich manifests that describe inputs, outputs, training contexts, confidence levels, invariance scopes, and privacy budgets.

A further object of the invention is to provide a mechanism builder that incrementally learns, refreshes, and validates causal mechanisms from local telemetry and permitted intervention logs, ensuring fidelity, applicability, and contextual stability.

Yet another object of the invention is to provide a causal-mechanism cache with content-addressable storage and compact capability summaries for efficient lookups, deduplication, and constant-time prechecks for answering what-if queries.

It is also an object of the invention to provide an intervention execution unit capable of composing relevant mechanisms, simulating proposed changes under individual or population conditions, and returning predictions with quantified uncertainty and complete provenance.

Another object of the invention is to maintain consistency across distributed nodes using an intervention-aware coherence protocol that synchronizes kernels and disseminates compact mechanism deltas keyed by intervention descriptors and invariance predicates.

A further object of the invention is to provide a selector that dynamically optimizes query execution strategies based on multi-objective costs involving latency, accuracy, energy consumption, bandwidth, and privacy budgets, while adhering to strict safety and policy constraints.

It is also an object of the invention to enforce privacy and compliance through a policy guard that applies purpose bindings, differential privacy accounting, and cryptographically verifiable receipts tied to specific mechanisms and policies.

Another object of the invention is to enable resilience in edge conditions through scoped invalidations, incremental updates, predictive prefetching, sandboxed execution, graceful fallback mechanisms, and auditable behavior.

A final object of the invention is to provide compact, standardized external interfaces that support kernel registration, capability exchange, counterfactual queries, delta dissemination, and scoped invalidation, thereby enabling broad applicability across industrial, healthcare, and network domains.

SUMMARY OF THE INVENTION

The present invention provides a distributed computing architecture that enables edge devices, gateways, and cloud services to perform counterfactual reasoning and answer “what-if” questions locally, quickly, and privately, without requiring large-scale data transfers or redeployment of full models. The system introduces modular components, including a causal-mechanism cache, a mechanism builder, an intervention execution unit, an intervention-aware coherence agent, a policy and privacy guard, and a selector for resource- and policy-aware decision-making.

Instead of moving datasets, the invention encapsulates domain relationships into compact, portable “mechanism kernels” with rich manifests describing inputs, outputs, training context, confidence, applicability conditions, and privacy budget states. These kernels are cached in a content-addressable layout, equipped with invariance certificates that define contextual validity, and maintained through scoped invalidations, deltas, and intervention-aware coherence updates.

The mechanism builder learns and refreshes kernels from local telemetry and permitted intervention logs, ensuring uncertainty calibration, provenance, and applicability tracking. The intervention execution unit composes kernels to simulate interventions, returning counterfactual predictions with quantified uncertainty and complete audit receipts. Consistency across nodes is maintained through a coherence protocol that disseminates compact mechanism deltas keyed by intervention descriptors and invariance predicates.

The selector dynamically determines whether to answer queries locally, relearn mechanisms, or fetch updates from peers, optimizing trade-offs between latency, accuracy, energy, bandwidth, and privacy budgets. Privacy and compliance are enforced by a policy guard that integrates purpose bindings, differential privacy accounting, and cryptographic receipts anchored to append-only logs.

Learning and update operations are incremental and scoped, with resilience measures for cold-start, intermittent connectivity, and constrained devices. Interfaces include endpoints for kernel registration, capability exchange, counterfactual query handling, delta dissemination, and scoped invalidation. Representative applications include industrial control, mobile health, and network operations, where the system delivers rapid, auditable, and sustainable counterfactual analytics while preserving privacy and minimizing data movement.

By introducing portable, certified causal mechanisms and synchronizing them through intervention-aware coherence, the invention provides a practical and auditable substrate for real-time counterfactual reasoning across heterogeneous edge and cloud environments, meeting requirements of speed, privacy, resilience, and sustainability.

BRIEF DESCRIPTION OF DRAWINGS

Fig 1: Detailed flow chart of the invention

Start: Marks the entry point. Upon power-up, the node, which may comprise an edge device, gateway, or micro-service, loads trusted policies and initializes background monitors for telemetry collection and drift detection.

Telemetry (background lane): The node continuously collects local operational logs, including inputs, actions, outcomes, and contextual metadata such as location, firmware version, or timestamp. The collected telemetry remains local unless policy permits external transfer and is used to support learning, validation, and system health checks.

Receive counterfactual request (targets; interventions; fidelity): An application submits a counterfactual query or “what-if” request, specifying: (i) target variables to estimate, (ii) proposed interventions or changes to system parameters, and (iii) fidelity constraints including latency requirements, accuracy or uncertainty thresholds, and privacy budget classifications.

Retrieve mechanism capability signature (MCS) & cache summary The node loads a compact representation of its current capabilities, including supported variables and interventions for locally cached mechanism kernels. A fast pre-check structure, such as a causal Bloom filter, is employed to predict cache hits or misses prior to deeper evaluation.

Decision: Capability match? The runtime evaluates whether the cached mechanism kernels, their certified applicability as defined by invariance scopes, and applicable privacy or policy bounds are sufficient to satisfy the incoming request. If sufficient ("Yes" path), the node can execute the request locally. Otherwise ("No" path), the selector module determines further action.

(Yes path) Compose relevant mechanism kernels in Intervention Execution Unit (IEU): Only the subset of mechanism kernels required to answer the request is loaded into a sandboxed runtime. The IEU composes these kernels following the underlying causal graph, substitutes any kernels corresponding to proposed interventions, and prepares for counterfactual simulation.

(Yes path) Execute counterfactual simulation: The IEU executes the composed kernels to generate requested statistics, including mean, quantile, range, or individual treatment effects, while propagating uncertainty. Execution is constrained by allocated CPU, GPU, or time budgets to satisfy latency requirements.

(Yes path) Policy Guard: apply purpose, locale, and privacy controls Prior to release of results, the Policy Guard enforces purpose-specific, geographic, and temporal restrictions. When required, differential-privacy noise is applied in accordance with the remaining privacy budget of the kernels involved**.**

(Yes path) Generate proof-of-use receipt (signed): The node generates a cryptographically signed receipt binding the request, the identities and versions of mechanism kernels used, applied policies, privacy parameters, and timestamps. This receipt serves as auditable evidence that no raw data left the node and only authorized kernels were invoked.

(Yes path) Return answer: result + confidence + provenance: The node transmits the numerical results to the client along with confidence measures and machine-readable provenance, including a listing of the mechanism kernels used, their versions, invariance scopes, and the signed policy receipt.

(Yes path, post-answer) Invariance broken? Immediately following execution—or periodically—the node evaluates background monitors to determine whether any mechanism's invariance conditions are violated, such as due to firmware updates or environmental shifts. Affected kernels are selectively quarantined in accordance with the Intervention-Aware Coherence Protocol (IACP).

(No path) Selector evaluates options (fetch delta, local relearn, remote enclave): When cached capabilities are insufficient, the selector evaluates options under constraints of latency, expected error, energy consumption, bandwidth, and privacy, selecting one of three actions:

A) Fetch mechanism delta from peer (IACP get_delta): The node identifies a peer advertising compatible capability and requests only the minimal mechanism delta required to satisfy the request. Transport is conducted via the Intervention-Aware Coherence Protocol.

Verify signature, scope & invariance: Received mechanism deltas are verified for cryptographic integrity, checked against the local context (e.g., region, device type, timestamp), and accepted only if the delta's certified invariance scope aligns with the local environment.

Update Causal-Mechanism Cache (CMC): Upon acceptance, the node updates the Causal-Mechanism Cache with the new or modified kernel and associated invariance certificate, then re-evaluates the capability match. If the request is now satisfiable, execution proceeds along the "Yes" path.

B) Local relearn in Mechanism Builder: If fetching a delta is not permitted or feasible, the node retrains only the missing or invalidated mechanisms using local telemetry. Refreshed kernels with updated invariance certificates are stored in the cache, after which the request is reattempted.

C) Remote execution in enclave: If latency constraints are relaxed or privacy policies require isolated computation, the node forwards the request to a trusted remote enclave. The enclave executes the simulation without exposing raw source data and returns both the result and a signed proof-of-use receipt.

Receive answer + receipt: The node validates the enclave's receipt and either returns the results to the client or, if permitted, stores a derived mechanism or delta to enhance future local capability.

Online drift/change-point detection (periodic task): Lightweight monitors continuously track data streams for distributional shifts, control parameter changes, or sensor alterations that may invalidate cached mechanisms. Triggers are recorded with audit evidence.

Invariance broken? → IACP invalidate (scoped quarantine) When a monitor detects a violation, the IACP issues a scoped invalidation marking only the implicated mechanisms as unusable for specific contexts or interventions, thereby avoiding complete cache flushes and preserving unaffected kernels.

Periodic tasks: prefetch likely mechanisms based on schedule (optional) During idle periods, the selector predicts probable upcoming queries and prefetches relevant mechanism deltas, replenishes privacy budgets, rotates keys, and prunes stale cache entries.

Mechanism Builder (module box): A background service identifies immediate causes for each variable, trains portable mechanism kernels, validates them across contexts, and emits manifests containing confidence, applicability, lineage, and privacy ledger state.

Causal-Mechanism Cache (CMC) (module box): A content-addressable store indexing kernels by target variable, parent-set hash, intervention mask, and invariance scope. Each entry maintains a counterfactual TTL and a capability summary for fast prechecks.

IEU — Intervention Execution Unit (module box): A sandboxed runtime that composes kernels just-in-time for query execution, performs counterfactual simulations under resource constraints, and outputs sanitized results.

IACP — Intervention-Aware Coherence Protocol (module box) The messaging layer for distributing signed mechanism deltas and scoped invalidations keyed to interventions and invariance predicates rather than memory addresses or opaque versions. Graph-scoped versioning supports conflict resolution after intermittent connectivity.

Policy Guard (module box): Enforces purpose bindings, privacy budgets, geographic and temporal restrictions, key management, and receipt generation to ensure compliance without exposing raw data.

How the pieces cooperate in practice (mini trace): Upon submission of a "what-if" request, the node first evaluates its capability. If satisfied, the request is executed locally via the IEU, policies are applied, and results with signed receipt are returned. If capability is insufficient, the selector chooses one of: fetching a mechanism delta from a peer, local relearning, or remote enclave execution. Monitors continuously assess invariance, triggering scoped quarantines and optional delta fetches or relearning. Periodically, the node prefetches likely kernels and removes stale entries to maintain readiness.

Fig 2: The architecture of the invention

Client / Application: The client comprises any software entity, including but not limited to factory controllers, mobile applications, or network services, configured to request a counterfactual response from an edge node. The client formulates a "what-if" query by specifying target outputs, proposed interventions (e.g., policy modifications or parameter adjustments), and desired fidelity or latency constraints. Upon execution, the client receives a numerical result accompanied by a confidence bound and machine-readable provenance, enabling verification of the mechanisms employed and archival of the decision context.

Edge Node (Containerized Runtime Environment): An edge node is a trusted compute environment hosting all runtime modules required to process counterfactual queries locally. The node exposes minimal APIs, including capability inquiry, counterfactual request execution, mechanism delta management (put/get), and kernel invalidation. Isolation is enforced between components, and the node is configured to operate under intermittent connectivity while optimizing for energy efficiency, bandwidth conservation, and adherence to accuracy and privacy constraints.

Causal-Mechanism Cache (CMC): The Causal-Mechanism Cache stores modular, portable "mechanism kernels" representing causal relationships available to the node. Each kernel is associated with an invariance certificate specifying valid contexts, including geographic region, device class, firmware version, and temporal window, as well as a counterfactual Time-To-Live (TTL) enforcing periodic revalidation. The cache maintains a capability summary enabling rapid hit/miss determination. The modular design permits selective composition of kernels to serve a query, avoiding reliance on a monolithic model.

Mechanism Builder: The Mechanism Builder generates and updates mechanism kernels from local telemetry, comprising inputs, actions, outcomes, and approved interventional logs. For each variable, it identifies immediate causal contributors, estimates response functions, calibrates uncertainty, and generates or refreshes invariance certificates. Upon detection of distributional drift or receipt of new data, the builder produces targeted mechanism deltas, avoiding full retraining and minimizing computational and bandwidth overhead.

Intervention Execution Unit (IEU): The IEU is a sandboxed runtime environment that dynamically composes only the kernels necessary to satisfy a given request. Proposed interventions are applied virtually by substituting affected kernels, and the IEU executes a counterfactual simulation to produce statistical outputs, including expectations, quantiles, or individual effects. Resource governance mechanisms enforce strict time and memory limits. The IEU outputs both results and a provenance summary detailing kernel versions, applicability scopes, and any applied privacy noise.

Policy and Privacy Guard: The Policy and Privacy Guard enforces purpose limitations, geographic and temporal constraints, and privacy budgets at the kernel level. Differential privacy is applied via calibrated noise when required. The guard verifies jurisdictional permissions for each request and prevents unauthorized data egress. Prior to result release, the guard generates a cryptographically signed proof-of-use receipt, binding the query, the mechanisms employed, and the policies enforced.

Selector: The Selector module evaluates, per request, whether to utilize cached kernels, retrieve mechanism deltas from peers, perform local relearning, or delegate computation to a remote enclave. Selection is based on a multi-objective optimization accounting for expected error, latency, energy and carbon cost, bandwidth usage, and privacy budget consumption. The Selector dynamically adapts its strategy based on observed performance metrics, including accuracy, queueing delays, and energy consumption, to achieve efficient service targets.

IACP Agent (Intervention-Aware Coherence Protocol): The IACP Agent maintains inter-node coherence using intervention semantics rather than opaque versioning. It disseminates scoped invalidations when invariance conditions are violated (e.g., firmware updates) and propagates compact mechanism deltas for partial domain updates. Capability handshakes enable peer nodes to determine delta applicability without exchanging raw data, thereby minimizing network load and compliance risk.

Audit and Receipts Journal: An append-only journal records cryptographically signed proof-of-use receipts, capability determinations, kernel invalidations, and delta applications. The journal provides a verifiable audit trail demonstrating that all responses were produced within declared capability and policy bounds, without transferring raw data. Regulatory entities or partner systems may perform audits by reviewing receipts rather than accessing sensitive data or executable code.

Scheduler / Prefetcher: The Scheduler or Prefetcher is a background service configured to anticipate upcoming what-if queries based on maintenance schedules, policy testing, or observed traffic patterns. During periods of low network activity, it prefetches relevant mechanism deltas to the edge node. The service additionally performs cache maintenance, including revalidating entries approaching expiration, evicting low-utility kernels, and staging local relearn tasks, thereby ensuring the node is prepared to meet peak demand efficiently.

Energy / Carbon Meter: The Energy/Carbon Meter monitors local energy consumption and, where available, consults grid carbon intensity metrics. This information is supplied to the Selector to inform low-energy operational decisions, such as reusing existing kernels versus fetching deltas, and scheduling non-urgent relearn or prefetch operations during periods of reduced carbon intensity. Such measures enable measurable reductions in energy expenditure per counterfactual response without compromising accuracy or correctness.

Peer Nodes: Peer nodes comprise neighboring edge devices or gateways that advertise compatible capability summaries. They act as sources and recipients of mechanism deltas and receive scoped invalidations propagated via the IACP Agent. Because exchanges occur at the granularity of individual kernels, each accompanied by signed manifests and invariance scopes, inter-node collaboration remains efficient, verifiable, and privacy-preserving.

Mechanism Registry / Exchange: The Mechanism Registry is a curated service for publishing certified mechanism kernels and associated capability descriptions. Edge nodes may retrieve vetted kernels compatible with their operational context, accelerating cold starts and promoting consistency of invariance claims across organizational or vendor boundaries. Raw datasets remain protected, and only kernels and metadata are exchanged.

Remote Confidential Enclave: A Remote Confidential Enclave serves as a fallback computational resource when local capability is insufficient or policy requires isolated execution. The enclave executes the counterfactual query without exposing source data, returning solely the final result and a cryptographically signed receipt. Where permitted, the enclave may additionally provide derived mechanism deltas to enhance local node capability for future queries.

Flow Semantics (Legend): In system diagrams, solid lines represent data or result flows, such as telemetry to the Mechanism Builder, kernels to the IEU, or answers to the client. Dotted lines denote control and policy flows, including Selector decisions, Policy Guard checks, and capability handshakes. Dashed lines indicate audit and receipt paths from the IEU and

Policy Guard to the append-only journal. This separation of data, control, and audit flows ensures correctness, governance, and trac

Fig 3: Device Diagram

Compute Module: The compute module provides general-purpose processing for control, scheduling, and orchestration of node operations. In one embodiment, the compute module comprises a multi-core CPU with vector extensions and virtualization capabilities, enabling isolated execution domains. The CPU executes supervisor functions, the policy engine, capability matching, and lightweight analytics, and exposes hardware counters for latency and energy monitoring. Trusted firmware enforces privilege separation so that untrusted application code cannot access mechanism parameters, privacy keys, or audit buffers.

AI Accelerator: An AI accelerator, such as a GPU, NPU, or DSP, offloads numerical kernels for mechanism learning and counterfactual simulations. The accelerator supports standard model formats and a sandboxed runtime to execute compiled mechanism kernels with bounded memory and without file or network access. Direct-memory-access is restricted via an IOMMU, ensuring that only whitelisted buffers are read or written, thereby preventing extraction of sensitive state.

Memory: System memory stores active mechanism kernels, capability summaries, and intermediate results. Error-correcting mechanisms protect against bit-flip corruption during extended simulations. Optional memory encryption prevents physical attacks on removable modules. A reserved memory region holds the privacy ledger and audit buffers, ensuring they are not paged or mapped into untrusted contexts.

Storage: Non-volatile storage (e.g., NVMe, eMMC, SSD) persists the mechanism cache, invariance certificates, receipts, and software images. A dedicated encrypted partition with authenticated metadata ensures tamper detection upon boot. Wear-leveling and background scrubbing preserve auditability and verify receipt integrity for regulatory compliance.

Security Module: A hardware root of trust enforces secure boot and device attestation. Keys are generated and stored in a TPM or secure element and released only when measured boot values match an approved configuration. The module enables remote attestation to peers or a registry, ensuring that mechanism deltas are accepted solely from devices operating under verified firmware and policies.

Data-Processing Unit / Smart-NIC (Optional): A network-attached data-processing unit accelerates compression and signing of mechanism deltas and validates scoped invalidation messages inline. The DPU exposes a limited API to the host, providing rate-limiting of outbound traffic and isolating intervention-aware coherence protocol traffic from general application networking.

Networking: The device supports one or more network interfaces (TSN-capable Ethernet, Wi-Fi 6/6E, and cellular). A policy layer enforces link-level priorities, ensuring critical flows, such as receipts and invalidations, are not starved by bulk traffic. Cellular interfaces may employ eSIM profiles to restrict roaming and enforce data localization compliance.

I/O Interfaces: A peripheral hub exposes serial and digital buses (USB, UART, SPI, I^{2}C, CAN, GPIO, PWM) for attachment to plant controllers, medical peripherals, or vehicular subsystems. Permission tables define which peripherals may provide telemetry to the mechanism builder and which may receive actuation commands derived from counterfactual results.

Sensor Array: The sensor array aggregates environmental, inertial, and optional audio/visual inputs. Calibration and timestamping services align heterogeneous signals for mechanism learning. On-device privacy filters (e.g., face or speech masking) remove identifying content prior to feature extraction.

Power and Power Management IC (PMIC): The PMIC accepts AC/DC or PoE input and may incorporate a battery for temporary power continuity. Monitored current and voltage values feed the energy/carbon estimator to support energy-aware operational decisions. Protection circuitry preserves storage and system integrity during brownouts or surges.

Thermal Management and Enclosure: Thermal elements, including heat sinks, fans, and ducts, maintain safe operating temperatures for compute and storage components under sustained loads. The enclosure provides ingress protection and EMI shielding, mitigating environmental hazards and side-channel leakage of model parameters.

Clock and Timing: A real-time clock with battery backup timestamps receipts and privacy-budget events. Optional GNSS or PTP synchronization enables multi-node reproducibility of counterfactual simulations and deterministic ordering of deltas and invalidations. A watchdog timer triggers recovery if the IEU or policy guard becomes unresponsive.

Secure Boot Chain (Firmware Layer): Boot ROM validates a signed first-stage loader, which in turn validates the operating system and container images. Only allowlisted software hashes may execute, and rollback protection prevents loading vulnerable images. Boot logs are sealed into the security module for subsequent attestation.

Operating System / Hypervisor: The operating system enforces process isolation and resource control. In alternate embodiments, a micro-hypervisor provides stronger partitioning between the IEU sandbox and other services. Kernel parameters disable unnecessary drivers, and mandatory access control policies restrict inter-process communication to essential channels.

Container Runtime: A container runtime hosts micro-services including the IEU, mechanism builder, policy guard, and IACP agent. Each container receives only the secrets and file mounts necessary for its function. Image provenance is enforced through signature verification upon deployment.

Intervention Execution Unit (Software): The IEU composes the minimal set of mechanism kernels required for a given request, applies proposed interventions, and executes the counterfactual simulation with bounded resources. The unit returns results with an associated uncertainty measure and a list of kernel versions used, which are verified by the policy guard prior to release.

Causal-Mechanism Cache (Software): The cache stores mechanism kernels with invariance certificates, counterfactual TTLs, and capability summaries. A content-addressable layout deduplicates identical kernels, and manifests link entries to their provenance. A constant-time capability precheck reduces latency for common queries.

Mechanism Builder (Software): The builder learns and refreshes kernels from authorized telemetry. When conditions change, it produces minimal mechanism deltas and updates invariance certificates. Builder outputs are cryptographically signed prior to peer distribution.

IACP Agent (Protocol): The intervention-aware coherence protocol propagates scoped invalidations and signed mechanism deltas. Messages carry intervention descriptors and invariance predicates, enabling peers to quarantine only affected kernels. Graph-scoped version vectors facilitate conflict detection and deterministic merges after intermittent connectivity.

Policy and Privacy Guard: The guard enforces purpose binding, geographic and temporal restrictions, and per-kernel differential privacy budgets. It validates receipts from remote enclaves, applies calibrated noise when necessary, and signs proof-of-use receipts binding results to mechanisms and policies. The guard serves as the final checkpoint before any data leaves the node.

Telemetry and Drift Monitors: Monitors track distributional shifts and configuration changes that may compromise kernel invariance. When triggers occur, the monitors notify the IACP agent to issue scoped invalidations and schedule local relearn or delta fetch operations. Monitored metrics also inform selector decisions and energy/carbon estimations.

Audit / Receipts Journal: An append-only log stores signed receipts, capability decisions, invalidations, and deltas. The journal enables independent audit of privacy compliance and system correctness without exposing raw data or kernel internals. Optional anchoring to a consortium ledger provides cross-organizational non-repudiation.

Client / Application Interface: The northbound interface accepts what-if requests and returns results with confidence and provenance. Requests breaching policy or requiring uncertified mechanisms are rejected. A "capability" endpoint may be exposed to allow clients to adapt queries prior to submission.

Peers / Cloud Interface: The southbound interface connects to peer nodes, mechanism registries, or confidential enclaves. It supports exchange of deltas, scoped invalidations, and fallback computation through mutually authenticated channels. Policy rules restrict cross-border transfers and limit bandwidth to protect local workloads.

System Overview: The described hardware and software layers collectively enable the device to locally learn, cache, and execute causal mechanisms, validate and update kernels securely, and provide auditable, privacy-preserving counterfactual answers with minimal energy, bandwidth, and latency overhead.

DETAILED DESCRIPTION OF INVENTION

The present invention relates to a distributed computing architecture that introduces causal mechanisms as the fundamental cacheable unit, supported by novel coherence, execution, and update protocols to enable counterfactual-native computation at the edge and across federated environments.

Core Concept: A first distinguishing feature of the invention lies in what it chooses to cache. Instead of storing bytes, feature tensors, or monolithic model binaries, the invention elevates causal mechanisms which are portable structural equations with explicit parent sets, parameterizations and uncertainty, to first-class, distributable artifacts. Each mechanism kernel carries an invariance certificate that encodes the contexts such as geography, device class, time window or policy regime under which the relationship is empirically stable. This packaging permits fine-grained reuse and targeted replacement of mechanisms, which has no direct analogue in conventional content delivery networks, model registries, or feature stores. By defining a machine-interpretable unit of causal computation, the invention makes “how it works” a cacheable object with explicit scope and validity.

Intervention-Aware Coherence Protocol: A second novel component is the intervention-aware coherence protocol (IACP), which synchronizes caches using the semantics of interventions rather than opaque versions or memory addresses. Coherence keys incorporate descriptors of the proposed do-operator (edges to cut, variables to clamp) and the invariance certificates of the relevant mechanisms. When drift or a policy change is detected, the protocol invalidates only those kernels whose invariance predicates fail under the intervention, thereby avoiding global flushes. This reframing of coherence from location/version keyed to intervention/invariance keyed yields a principled, minimal update frontier aligned with decision semantics.

Mechanism capability signature: The invention also introduces a mechanism capability signature (MCS), which is a compact, cryptographically hashable summary of a node's ability to answer a given counterfactual. The MCS may be implemented as an extended adjacency matrix combined with typed variable metadata, intervention support flags, and an invariance bitmap. Optionally, a causal bloom filter may be employed to allow probabilistic pre-checks for capability matches across large graphs. Through this capability handshake, peers can decide prior to any data or model transfer whether a query is answerable locally or whether a specific mechanism delta is required. This design differs fundamentally from version checks or model identifiers and enables low-bandwidth, safe federation across organizational boundaries.

Counterfactual Computation as a Cache Operation: Counterfactual computation is treated as a first-class cache operation. The system provides an Intervention Execution Unit (IEU) that performs abduction–action–prediction by composing cached kernels, abducting latent noise when necessary, substituting intervened mechanisms, and simulating outcomes to produce expectations, quantiles, or individual treatment effects with calibrated uncertainty. Unlike predictive serving layers that treat inference as a black box, the IEU is explicitly causal and compositional, stitching together only the mechanisms implicated by the query, propagating their uncertainty, and documenting provenance. This renders "what-if" analysis as efficient as a cache hit while maintaining scientific traceability.

Mechanism Delta Dissemination: The invention further minimizes update traffic through mechanism delta dissemination. Rather than transmitting entire models, peers exchange compressed, scoped deltas such as low-rank parameter updates, sparse structure edits (adding or removing parents), or symbolic patches to transformation functions. Each delta is signed and tied to the affected mechanisms and their invariance scopes, enabling deterministic replay and audit. This approach allows large networks of edge nodes to remain current with kilobyte-scale changes rather than full redeployments of megabyte- or gigabyte-scale models. Unlike gradient sharing or whole-model updates in federated learning, mechanism deltas align with the causal graph and reduce both bandwidth and carbon cost.

Privacy and Compliance: Privacy and compliance are achieved by binding privacy budgets and purpose limitations to mechanisms themselves rather than to datasets or endpoints. Each kernel maintains a differential-privacy ledger and explicit use constraints (for example, "only for policy evaluation in region R until date T"). The runtime generates proof-of-use receipts that cryptographically bind an answer to the specific mechanisms and policies invoked. These receipts enable verifiable assurance that no raw data or disallowed mechanism left the device while supporting audit across organizations without revealing sensitive content. Prior systems record access logs; the invention instead inseparably binds permitted uses to actual uses at the mechanism level.

Energy- and Carbon-Aware Selector: The invention also incorporates an energy- and carbon-aware selector that optimizes at mechanism granularity where and how to satisfy a request. Using contextual bandits or reinforcement learning, the selector evaluates reuse, local relearning, or delta fetching under constraints of latency, error tolerance, privacy budgets, and real-time grid carbon intensity. Because mechanisms are small and richly annotated, the selector can attribute costs and accuracy to specific graph regions and learn Pareto-efficient policies over time. This fine-grained selection contrasts with rule-based scaling heuristics of existing serving stacks and directly addresses energy proportionality and sustainability objectives.

Invariance-Centric Monitoring for Robustness: Robustness to change is delivered by invariance-centric monitoring. Lightweight change-point tests and transport metrics operate at the mechanism level to detect when assumed invariances no longer hold locally. Upon detection, the system performs a partial rollback that quarantines only the affected kernels, preserving the remainder of the graph, and requests targeted relearning or delta retrieval. This confinement of corrective action to a subgraph while maintaining coherent counterfactual semantics across the system is unique to the mechanism-first design and reduces mean time to correctness.

Deployment Characteristics: Deployment of the invention is hardware-agnostic yet offload-friendly. Mechanism kernels can be compiled into small ONNX or TorchScript modules accompanied by metadata sidecars, while the coherence protocol and IEU can be accelerated on DPUs or NICs for in-line delta compression and secure composition. Sensitive domains may use enclaves to host the IEU without centralizing full datasets. Because the invention transmits small, typed kernels with clear interfaces, heterogeneous environments with CPUs, GPUs, and accelerators become interoperable without sharing monolithic models or raw data.

Ecosystem Layer: Finally, the invention establishes an ecosystem layer. Registries and exchanges for mechanism kernels are keyed by capability signatures; policy templates standardize invariance claims; and developer tooling converts domain code—such as control laws, decision trees, or micro-ML models—into certifiable kernels. This fosters a marketplace in which organizations can trade "how-it-works" components with privacy-preserving guarantees and precise applicability, thereby enabling collaborative counterfactual analytics across jurisdictions.

The core architecture of the mechanism kernels in the system are enumerated in detail below:

The kernels, together with their associated metadata, serve as the foundational elements of a coordinated runtime that can execute counterfactual queries, adapt to environmental changes, and maintain compliance with privacy and policy constraints.

Core Architecture

Each node participating in the system, whether an edge device, gateway, or cloud service, hosts a set of coordinated modules:

1. **Causal-Mechanism Cache** – A content-addressable store that maintains mechanism kernels and their manifests. It supports primary and secondary indexing, pre-check structures for capability queries, and journaled logging of additions, removals, validations, and uses. Each entry carries a counterfactual time-to-live and an applicability bitmap, ensuring timely revalidation and proper reuse.

2. **Mechanism Builder** – A module responsible for learning, refreshing, and validating mechanism kernels from telemetry and permitted intervention logs. For each variable, it identifies direct causes, estimates a predictive mechanism, learns stochastic fluctuations, and generates an invariance certificate that encodes the contexts in which the mechanism remains valid. The builder serializes the kernel together with training metadata, error calibration, privacy budget state, and scope of applicability.

3. **Intervention Execution Unit (IEU)** – A sandboxed runtime that composes kernels to simulate counterfactual scenarios. For population-level questions, it propagates effects across the mechanism graph; for individual-level questions, it reconstructs latent conditions, applies interventions, and predicts outcomes with uncertainty. It returns results with a provenance bundle that enumerates kernel versions, contexts, and applied privacy parameters.

4. **Intervention-Aware Coherence Agent** – A protocol engine that ensures consistency across nodes. Instead of conventional versioning or address-based invalidations, coherence operates on intervention descriptors and invariance predicates. Scoped quarantines are issued when invariance breaks, and updates are disseminated as compact mechanism deltas (low-rank parameter edits, sparse structural changes, or symbolic patches). Graph-scoped version vectors maintain ordering and resolve conflicts during reconciliation.

5. **Selector** – A decision-making component that optimizes how each request is satisfied, balancing latency, accuracy, privacy, energy, and bandwidth constraints. It may reuse cached kernels, trigger local

relearning, or fetch deltas from peers. Implemented as a contextual bandit or constrained reinforcement learner, it adapts through continuous feedback and enforces strict safety rules such as geographic boundaries and latency caps.

6. **Policy and Privacy Guard** – A compliance layer that enforces privacy budgets, purpose bindings, and jurisdictional constraints directly at the mechanism level. Each kernel maintains its own differential-privacy ledger, and answers consume from this ledger with calibrated noise when necessary. Signed receipts cryptographically bind every answer to the mechanisms, policies, and privacy parameters used, enabling auditable compliance without disclosing raw data.

Mechanism of the Kernels: Each mechanism kernel is a portable, self-describing unit consisting of:

- A predictive function linking a variable to its direct causes.
- A stochastic noise model for uncertainty representation.
- An invariance certificate specifying certified contexts (e.g., geography, device class, firmware, season, or policy regime).
- A manifest with input hashes, training metadata, calibration scores, privacy state, and scope bitmaps.

By representing “how it works” as a cacheable and certifiable artifact, the invention elevates causal mechanisms to first-class computational units, enabling fine-grained reuse, traceable updates, and distributed auditability.

Intervention-Aware Coherence Protocol (IACP): The IACP ca replaces global flushes with targeted quarantines. When invariance breaks due to events such as software updates or policy changes, only the implicated kernels are invalidated. Updates are exchanged as mechanism deltas that describe scoped parameter changes, structure edits, or symbolic patches, thereby minimizing bandwidth. Graph-scoped version vectors allow nodes to merge deltas deterministically and maintain coherent counterfactual semantics.

Selector and Optimization: The selector continuously evaluates whether to answer locally, relearn a mechanism, or fetch a delta. Its cost function accounts for expected error, latency, privacy budget consumption, bandwidth usage, energy draw, and live grid carbon intensity. By attributing costs and uncertainties at the mechanism level, the selector learns Pareto-efficient strategies that balance responsiveness with sustainability.

Privacy and Compliance: The system enforces privacy and policy through mechanism-level bindings. Each kernel maintains its own differential-privacy ledger, and receipts bind answers to kernels, policies, and applied noise. Purpose and scope constraints prevent unauthorized use, and receipts are anchored to append-only logs or consortium ledgers for verifiable audit. Hardware-backed key stores and optional remote attestation strengthen compliance guarantees.

Learning and Delta Dissemination: Mechanism learning and updates are incremental and scoped. The builder refreshes only affected kernels, validates proposed edits through stability and counterfactual tests, and packages signed deltas describing semantic changes and updated capability flags. Receiving nodes verify scope overlap and either accept, restrict, or reject deltas, with all actions logged for transparency.

Resilience and Edge Operation: The invention supports cold-start nodes, intermittent connectivity, and constrained environments. Nodes can start with empty capability signatures, forward high-risk queries to enclaves, and queue invalidations locally. Predictive prefetching prepares kernels for likely interventions, while runtime guards ensure safe execution under uncertainty.

Interfaces: External interfaces are standardized for interoperability:

- **Registration endpoint** for kernel manifests.
- **Capability endpoint** for node signatures and summaries.
- **Counterfactual endpoint** for intervention queries and auditable responses.
- **Delta endpoint** for exchanging signed updates.
- **Invalidate endpoint** for applying scoped quarantines.

Declarative policy templates define operational rules, and client libraries expose high-level what-if APIs for application developers.

Representative Use Cases

Factory operations: A controller asks the throughput effect of reducing conveyor speed under high humidity. Local kernels simulate the change and return results within milliseconds. If lubricant updates invalidate a jam-rate mechanism, only that kernel is quarantined and a delta retrieved from a peer.

Mobile health: A treatment recommender evaluates dosage adjustments directly on-device, consuming a small privacy budget and issuing a receipt proving compliance.

Network operations: A router evaluates the effect of a peering path change, executing locally when carbon intensity is low and latency constraints are tight.

Technical Advantages

The invention introduces a new substrate for distributed counterfactual reasoning by combining:

- Mechanism-level packaging and certification.
- Intervention-aware coherence.
- Capability handshakes.
- Delta-based updates.
- Privacy-bound receipts.
- Energy-aware selection.
- Scoped quarantines and partial rollbacks.

We claim:

1. A distributed computing system for performing counterfactual and intervention-aware computation across edge, gateway, and cloud nodes, the system comprising:

a. a causal-mechanism cache configured to store mechanism kernels as first-class cacheable units, each kernel comprising:

   i. a predictive function linking a variable to its direct causes;

   ii. a stochastic noise model for uncertainty representation;

   iii. an invariance certificate specifying contexts of validity; and

   iv. a manifest describing input data hashes, training metadata, privacy state, and context applicability;

b. a mechanism builder configured to learn, refresh, and validate mechanism kernels from local telemetry and permitted intervention logs, and to generate invariance certificates that specify contexts in which each kernel remains valid;

c. an Intervention Execution Unit configured to compose mechanism kernels, simulate interventions, and compute expected outcomes, quantiles, or individual treatment effects with calibrated uncertainty, and to generate provenance bundles enumerating kernel versions, contexts, and applied privacy parameters;

d. an intervention-aware coherence agent configured to maintain consistency across nodes using intervention descriptors and invariance predicates, issuing scoped quarantines for only affected kernels, and disseminating mechanism deltas including parameter edits, structural changes, or symbolic patches;

e. a selector configured to decide, for each incoming query, whether to reuse cached kernels, relearn locally, or fetch deltas from peers based on latency, accuracy, privacy, energy, and bandwidth constraints; and

f. a policy and privacy guard configured to enforce privacy budgets, purpose bindings, and jurisdictional constraints at the mechanism level, and to generate cryptographically auditable proof-of-use receipts for each query.

2. The system as claimed in claim 1, wherein further comprising a selector configured to decide, for each incoming query, whether to reuse cached kernels, relearn locally, or fetch mechanism deltas from peer nodes based on latency, accuracy, privacy, energy, and bandwidth constraints.

3. The system as claimed in claim 1 or 2, wherein each mechanism kernel is portable and distributable, allowing fine-grained reuse and targeted replacement without transmitting full models or raw datasets.

4. The system as claimed in claim 1-3, wherein the intervention-aware coherence agent keys cache validity based on descriptors of proposed interventions and invariance predicates, and propagates updates as scoped mechanism deltas.

5. The system as claimed in claim 1-4, wherein the intervention execution unit performs abduction–action–prediction by abducting latent noise, substituting intervened mechanisms, and propagating uncertainty from constituent kernels to generate scientifically traceable counterfactual outcomes.

6. The system as claimed in claims 1–5, wherein mechanism deltas are compressed and signed, and include low-rank parameter updates, sparse structural edits, or symbolic patches, enabling deterministic replay and low-bandwidth updates across nodes.

7. The system as claimed in claims 1–6, wherein the policy and privacy guard maintains per-kernel differential-privacy ledgers, purpose and scope bindings, and generates signed receipts anchored to append-only logs or consortium ledgers to enable verifiable audit without revealing sensitive data.

8. The system as claimed in claims 1–7, further comprising external interfaces including a registration endpoint for kernel manifests, a capability endpoint for node signatures, a counterfactual endpoint for intervention queries, a delta endpoint for exchanging signed updates, and an invalidate endpoint for scoped quarantines.

9. A method of performing distributed counterfactual computation using the system of any preceding claim, the method comprising the steps of:

a. learning mechanism kernels from local telemetry and permitted intervention logs;

b. storing mechanism kernels in a causal-mechanism cache with invariance certificates and manifests;

c. executing counterfactual queries using the intervention execution unit by composing relevant kernels and simulating proposed interventions;

d. maintaining consistency across nodes using an intervention-aware coherence agent by quarantining only affected kernels and disseminating mechanism deltas;

e. selecting a strategy to satisfy each query using a selector based on latency, accuracy, privacy, energy, and bandwidth constraints; and

f. enforcing privacy and compliance via a policy and privacy guard and generating auditable proof-of-use receipts.

10. The method as claimed in claim 9, wherein the selector continuously learns Pareto-efficient strategies for balancing responsiveness, sustainability, and accuracy by attributing costs and uncertainties at the mechanism level.

11. The method as claimed in claim 9 or 10, further comprising performing partial rollbacks and targeted delta retrievals upon detection of invariance violations, thereby preserving unaffected kernels and maintaining coherent counterfactual semantics.

12. The method as claimed in any of claims 9–11, further comprising incrementally updating only affected mechanism kernels using warm starts, stability tests, and counterfactual sanity checks prior to dissemination as signed deltas.

13. 13. The system as claimed in claim 1, wherein each node comprises a compute module including a multi-core CPU with vector extensions and virtualization capabilities, the compute module being configured to execute supervisor functions, policy enforcement, capability matching, and telemetry analysis.

14. The system as claimed in claim 1 or 13, further comprising an AI accelerator, the accelerator being selected from a GPU, NPU, or DSP, configured to offload numerical kernels for mechanism learning and

counterfactual simulations, and operating in a sandboxed runtime with restricted memory and I/O access.

15. The system as claimed in any of claims 1–14, wherein system memory stores active mechanism kernels, capability summaries, and intermediate results, and includes a reserved region for privacy ledger and audit buffers protected from untrusted access.

16. The system as claimed in any of claims 1–15, further comprising non-volatile storage configured to persist mechanism kernels, invariance certificates, proof-of-use receipts, and software images, wherein storage includes an encrypted partition with authenticated metadata for tamper detection.

17. The system as claimed in any of claims 1–16, further comprising a security module providing a hardware root of trust for secure boot, device attestation, key management, and verification of mechanism deltas received from peers.

18. The system as claimed in any of claims 1–17, further comprising a network interface supporting one or more of TSN-capable Ethernet, Wi-Fi, or cellular connectivity, the interface being controlled by a policy layer for prioritizing critical flows such as proofs-of-use and scoped invalidations.

19. The system as claimed in any of claims 1–18, further comprising a peripheral hub configured to interface with sensors or actuators, wherein telemetry inputs are filtered for privacy before being processed by the mechanism builder.

20. The system as claimed in any of claims 1–19, further comprising a power management module and optional battery, the module monitoring energy consumption and providing real-time metrics to the selector for energy- and carbon-aware decision-making.

21. The system as claimed in any of claims 1–20, further comprising a thermal management system and enclosure configured to maintain safe operating temperatures and protect against EMI and environmental hazards.

22. The system as claimed in any of claims 1–21, wherein the intervention execution unit and coherence agent are optionally offloadable to hardware accelerators, network-attached DPUs, NICs, or secure enclaves for sensitive domains.

Dated this 29th day of September 2025

Sudarshana

Sudarshana Bandyopadhyay

Regn. No.: IN/PA 2802

Agent for the applicant

Phn No. 9748818235

Email: bandyopadhyay.sudarshana@gmail.com

APPLICATION NO.____________

ABSTRACT

CAUSAL MECHANISM-BASED DISTRIBUTED COMPUTING ARCHITECTURE FOR EDGE AND FEDERATED ENVIRONMENTS

The present invention provides a distributed computing architecture for real-time counterfactual reasoning and intervention-aware analytics across edge, gateway, and cloud nodes. The system elevates causal mechanisms to first-class cacheable units, where each mechanism kernel encapsulates a predictive function, stochastic noise model, invariance certificate, and manifest detailing inputs, training metadata, privacy state, and applicability contexts. Consistency across nodes is maintained by an intervention-aware coherence protocol that propagates updates as compact mechanism deltas keyed to interventions and invariance predicates, avoiding global cache flushes. A mechanism capability signature enables nodes to pre-check whether queries can be executed locally or require specific updates, facilitating low-bandwidth, secure federation. Counterfactual computation is performed by an Intervention Execution Unit that composes kernels, simulates proposed interventions, and returns outcomes with calibrated uncertainty and verifiable provenance. A selector optimizes reuse, local relearning, or delta fetching under constraints of latency, accuracy, energy, bandwidth, and privacy budgets. Privacy and compliance are enforced through purpose bindings, differential privacy ledgers, and cryptographically auditable proof-of-use receipts. The invention reduces data movement, supports heterogeneous hardware, and establishes a certified kernel ecosystem, providing an auditable, efficient, and privacy-preserving substrate for distributed "what-if" computation.

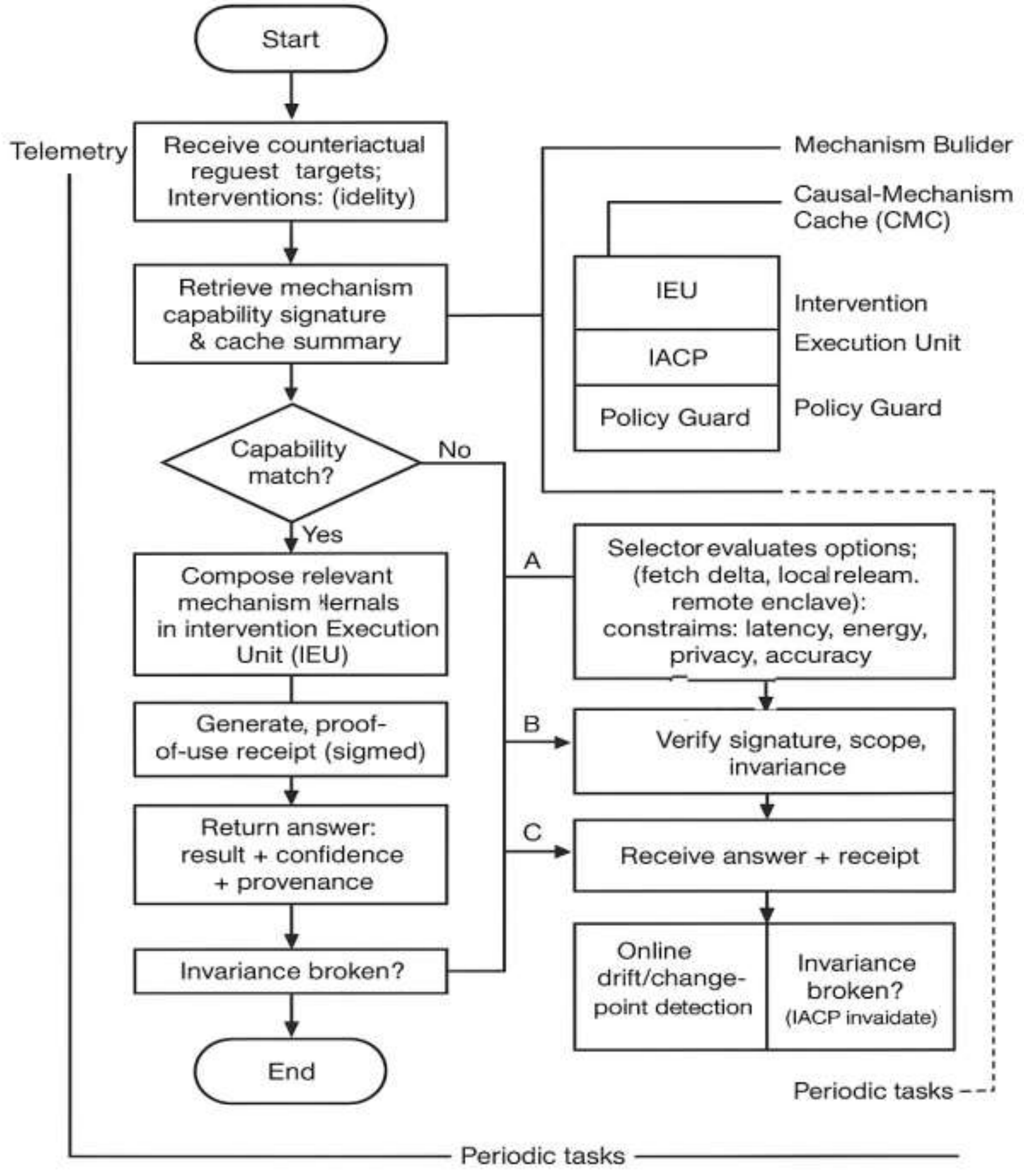
Fig 2

Applicant Name: SRJX RESEARCH AND INNOVATION LAB LLP

Total No. of Sheets: 3

Appl No. -

Sheet 1 of 3


Start
Telemetry
Receive counteriactual reguest targets; Interventions: (idelity)
Retrieve mechanism capability signature & cache summary
Capability match?
No
Yes
Compose relevant mechanism kernals in intervention Execution Unit (IEU)
Generate, proof-of-use receipt (sigmed)
Return answer: result + confidence + provenance
Invariance broken?
End
Mechanism Bulider
Causal-Mechanism Cache (CMC)
IEU
IACP
Policy Guard
Intervention
Execution Unit
Policy Guard
A
B
C
Selector evaluates options; (fetch delta, local releam. remote enclave): constraims: latency, energy, privacy, accuracy
Verify signature, scope, invariance
Receive answer + receipt
Online drift/change-point detection
Invariance broken? (IACP invaidate)
Periodic tasks
Periodic tasks

Detailed Flowchart

Figure 1

Sudarshana

Sudarshana Bandyopadhyay
Regn No.: IN/PA 2802
Agent for the Applicants

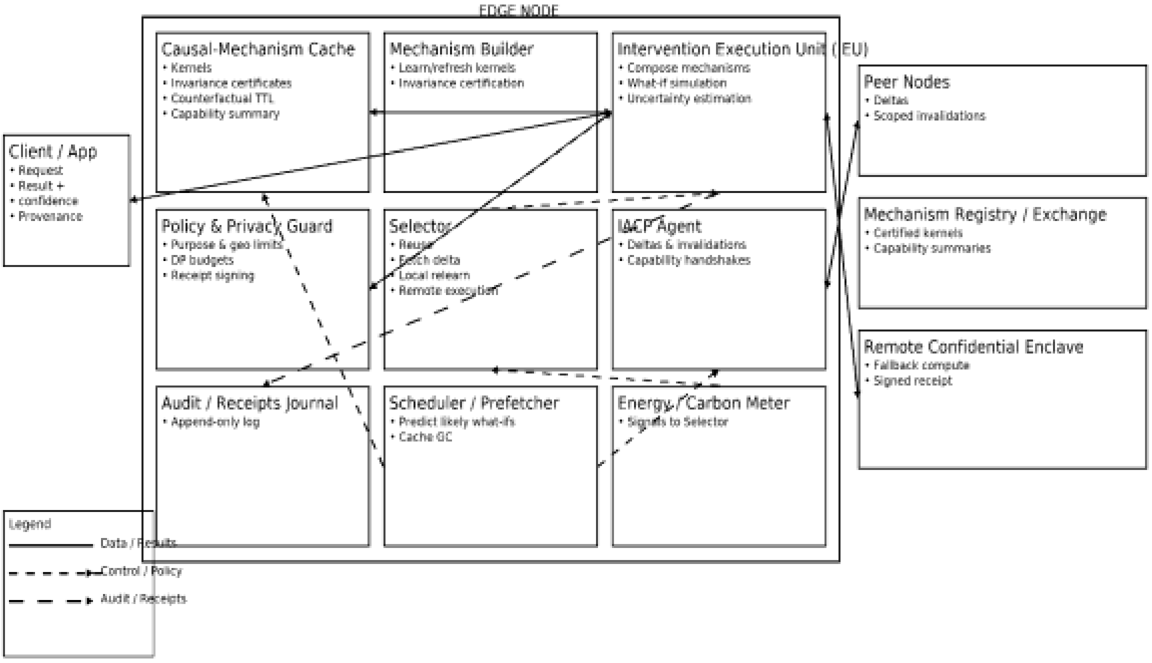
Applicant Name: SRJX RESEARCH AND INNOVATION LAB LLP

Total No. of Sheets: 3

Appl No. -

Figure 2 of 3

Intervention-Aware Causal-Mechanism Cache



Detailed Architecture

Figure 2

Sudarshana

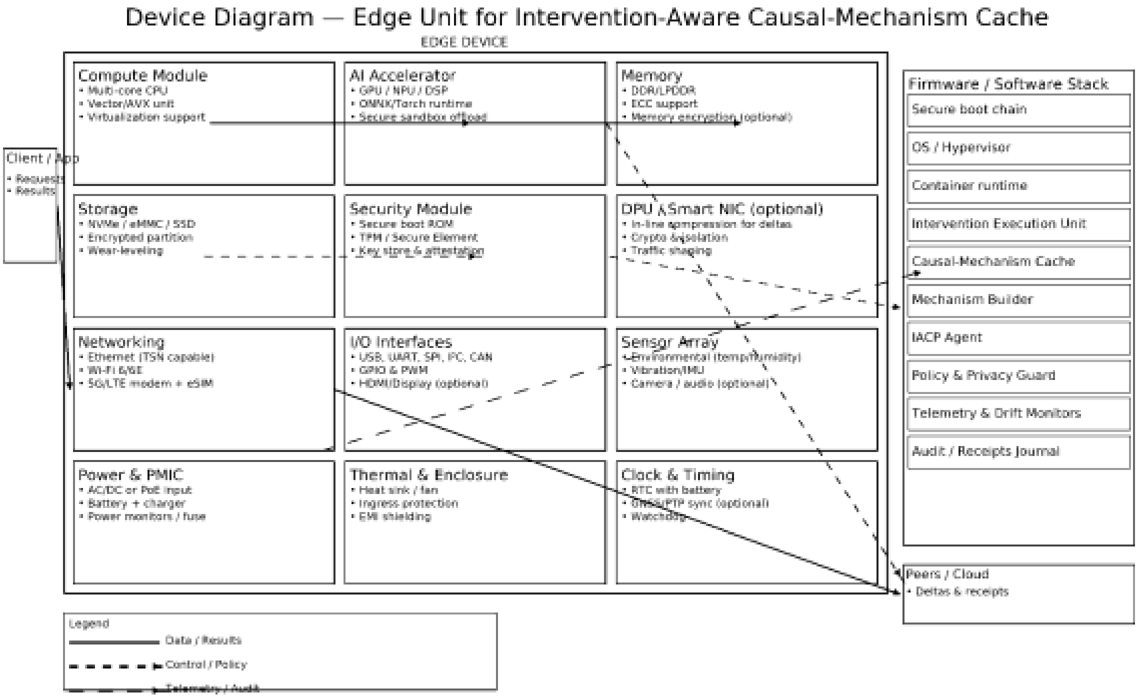
Sudarshana Bandyopadhyay
Regn No.: IN/PA 2802
Agent for the Applicants

Applicant Name: SRJX RESEARCH AND INNOVATION LAB LLP

Total No. of Sheets: 3

Appl No. -

Figure 3 of 3



Device Diagram

Figure 3

Sudarshana

Sudarshana Bandyopadhyay
Regn No.: IN/PA 2802
Agent for the Applicants

FORM 5
THE PATENTS ACT, 1970
(39 of 1970)
&
THE PATENTS RULES, 2003

Declaration as to Inventorship
[*See* section 10(6) and rule 13(6)]

1. **NAME OF APPLICANT**: SRJX RESEARCH AND INNOVATION LAB LLP,

hereby declare that the true and first inventor(s) of the invention disclosed in the complete specification filed in pursuance of our application numbered ____________________ dated 30 September 2025 are:

2. **INVENTORS:**

I. a) Name: **DR SOUMYA RANJAN JENA**
b) Nationality: An Indian National
c) Address: Plot No - 3E/474, Sector-9, CDA, Post- Markat Nagar,
Cuttack-753014, Odisha, India

II. a) Name: **MR SANJOY SAHA**
b) Nationality: An Indian National
c) Address: Flat No - 63/1, Thakur Para Road, P.O.- Naihati, North 24 Parganas, West Bengal-743165, India

III. a) Name: **DR. SOHIT AGARWAL**
b) Nationality: Indian
c) Address: D 388, Sarvanand Marg, Malviya Nagar, Jaipur-302017, Rajasthan, India

Dated this 30th day of September 2025

Sudarshana

Name of the signatory:

Signature Not Verified

Digitally Signed.
Name: Sudarshana
Bandyopadhyay
Date: 30-Sep-2025 20:05:31
Reason: Patent Efiling

Dated this 30th day of September 2025

Sudarshana Bandyopadhyay
Regn No.: IN/PA 2802
Agent for the Applicants
Email: bandyopadhyay.sudarshana@gmail.com
Phn No: 9748818235

To,
The Controller of Patents,
The Patent Office
At Kolkata

UDYAM REGISTRATION CERTIFICATE

UDYAM REGISTRATION NUMBER **UDYAM-OD-07-0095836**

NAME OF ENTERPRISE **SRJX RESEARCH AND INNOVATION LAB LLP**

TYPE OF ENTERPRISE *

SNo.	Classification Year	Enterprise Type	Classification Date
1	2025-26	Micro	16/08/2025

MAJOR ACTIVITY **SERVICES**

SOCIAL CATEGORY OF ENTREPRENEUR **GENERAL**

NAME OF UNIT(S)

S.No.	Name of Unit(s)
1	SRJX RESEARCH AND INNOVATION LAB LLP

OFFICAL ADDRESS OF ENTERPRISE

Flat/Door/Block No.	PLOT NO-3E/474	Name of Premises/ Building	SECTOR-9
Village/Town	CDA CUTTACK	Block	NA
Road/Street/Lane	Avinab Bidanasi	City	Cuttack Sadar
State	ODISHA	District	CUTTACK , Pin 753014
Mobile	9090255155	Email:	soumyajena1989@gmail.com

DATE OF INCORPORATION / REGISTRATION OF ENTERPRISE **05/05/2025**

DATE OF COMMENCEMENT OF PRODUCTION/BUSINESS **05/05/2025**

NATIONAL INDUSTRY CLASSIFICATION CODE(S)

SNo.	NIC 2 Digit	NIC 4 Digit	NIC 5 Digit	Activity
1	72 - Scientific research and development	7210 - Research and experimental development on natural sciences and engineering	72100 - Research and experimental development on natural sciences and engineering	Services

DATE OF UDYAM REGISTRATION **16/08/2025**

* **In case of graduation (upward/reverse) of status of an enterprise, the benefit of the Government Schemes will be availed as per the provisions of Notification No. S.O. 2119(E) dated 26.06.2020 issued by the M/o MSME.**

Disclaimer: This is computer generated statement, no signature required. Printed from https://udyamregistration.gov.in & Date of printing - 16/08/2025

Signature Not Verified
Digitally Signed.
Name: Sudarshana Bandyopadhyay
Date: 30-Sep-2025 22:20:14
Reason: Patent Efiling

For any assistance, you may contact:

1. District Industries Centre: CUTTACK (ODISHA)

2. MSME-DFO: CUTTACK (ODISHA)

Visit : www.msme.gov.in ; www.dcmsme.gov.in ; www. n

Follow us @minmsme & @ms

Udyam Registration Number : UDYAM-OD-07-0095836

Type of Enterprise	MICRO	**Major Activity**	Services
Type of Organisation	Limited Liability Partnership	**Name of Enterprise**	SRJX RESEARCH AND INNOVATION LAB LLP
Owner Name	SRJX RESEARCH AND INNOVATION LAB LLP	**PAN**	AFPFS4480L
Do you have GSTIN	No	**Mobile No.**	9090255155
Email Id	soumyajena1989@gmail.com	**Social Category**	General
Gender	Male	**Specially Abled(DIVYANG)**	No
Date of Incorporation	05/05/2025	**Date of Commencement of Production/Business**	05/05/2025

Bank Details

Bank Name	IFS Code	Bank Account Number
Punjab national bank	PUNB0787800	7878002100002490

Employment Details

Male	Female	Other	Total
3	2	0	5

Investment in Plant and Machinery OR Equipment (in Rs.)

S.No.	Financial Year	Enterprise Type	Written Down Value (WDV)	Exclusion of cost of Pollution Control, Research & Development and Industrial Safety Devices	Net Investment in Plant and Machinery OR Equipment[(A)-(B)]	Total Turnover (A)	Export Turnover (B)	Net Turnover [(A)-(B)]	Is ITR Filled?	ITR Type
1	2023-24	Micro	0.00	0.00	0.00	0.00	0.00	0.00	No	NA

Unit(s) Details

SN	Unit Name	Flat	Building	Village/Town	Block	Road	City	Pin	State	District
1	SRJX RESEARCH AND INNOVATION LAB LLP	PLOT NO-3E/474	SECTOR-9	CDA CUTTACK	NA	Avinab Bidanasi	Cuttack Sadar	753014	ODISHA	CUTTACK

Official address of Enterprise

Flat/Door/Block No.	PLOT NO-3E/474	**Name of Premises/ Building**	SECTOR-9
Village/Town	CDA CUTTACK	**Block**	NA
Road/Street/Lane	Avinab Bidanasi	**City**	Cuttack Sadar
State	ODISHA	**District**	CUTTACK , **Pin :** 753014
Mobile	9090255155	**Email:**	soumyajena1989@gmail.com
Latitude	20.5021859203546	**Longitude:**	85.88860428847029

National Industry Classification Code(S)

SNo.	Nic 2 Digit	Nic 4 Digit	Nic 5 Digit	Activity
1	72 - Scientific research and development	7210 - Research and experimental development on natural sciences and engineering	72100 - Research and experimental development on natural sciences and engineering	Services

Are you interested to get registered on Government e-Market (GeM) Portal	No
Are you interested to get registered on TReDS Portals(one or more)	No
Are you interested to get registered on National Career Service(NCS) Portal	No
Are you interested to get registered on NSIC B2B Portal	No
Are you interested in availing Free .IN Domain and a business email ID	N/A
Are you interested in getting registered on Skill India Digital Portal	No
District Industries Centre	CUTTACK (ODISHA)
MSME-DFO	CUTTACK (ODISHA)
Date of Udyam Registration	16/08/2025
Date of Printing	16/08/2025

IEC Details	
IEC Number	
IEC Status	Inactive
IEC Registration Date	
IEC Modification Date	

“**FORM 1** THE PATENTS ACT 1970 (39 of 1970) and THE PATENTS RULES, 2003 **APPLICATION FOR GRANT OF PATENT** (See section 7, 54 and 135 and sub-rule (1) of rule 20)	(FOR OFFICE USE ONLY)	
	Application No.	
	Filing date:	
	Amount of Fee paid:	
	CBR No:	
	Signature:	
1. APPLICANT’S REFERENCE / IDENTIFICATION NO. (AS ALLOTTED BY OFFICE)		

2. TYPE OF APPLICATION [Please tick (✓) at the appropriate category]					
Ordinary (✓)		Convention ( )		PCT-NP ( )	
Divisional ( )	Patent of Addition ( )	Divisional ( )	Patent of Addition ( )	Divisional ( )	Patent of Addition ( )

3A. APPLICANT(S)			
Name in Full	Nationality	Country of Residence	Address of the Applicant
SRJX RESEARCH AND INNOVATION LAB LLP	Indian	India	SRJX RESEARCH AND INNOVATION LAB LLP, Plot No - 3E/474, Sector-9, CDA, Post- Markat Nagar, Cuttack-753014, Odisha, India

3B. CATEGORY OF APPLICANT [Please tick (✓) at the appropriate category]			
Natural Person ( )	Other than Natural Person		
	Small Entity (✓)	Startup ( )	Others ()

4. INVENTOR(S) [Please tick (✓) at the appropriate category]		
Are all the inventor(s) same as the applicant(s) named above?	Yes ( )	No (✓)

Signature Not Verified
Digitally Signed.
Name: Sudarshana
Bandyopadhyay
Date: 30-Sep-2025 20:03:59
Reason: Patent Efiling

If "No", furnish the details of the inventor(s)

Name in Full	Nationality	Country of Residence	Address of the Inventor
DR SOUMYA RANJAN JENA	Indian	India	Plot No - 3E/474, Sector-9, CDA, Post-Markat Nagar, Cuttack-753014, Odisha, India
MR SANJOY SAHA	Indian	India	63/1, Thakur Para Road, P.O.- Naihati, North 24 Parganas, West Bengal-743165, India
DR SOHIT AGARWAL	Indian	India	D 388, Sarvanand Marg, Malviya Nagar, Jaipur-302017, Rajasthan, India

5. TITLE OF THE INVENTION

CAUSAL MECHANISM-BASED DISTRIBUTED COMPUTING ARCHITECTURE FOR EDGE AND FEDERATED ENVIRONMENTS

6. AUTHORISED REGISTERED PATENT AGENT(S)	IN/PA No.	2802
	Name	Sudarshana Bandyopadhyay
	Mobile No.	9748818235
7. ADDRESS FOR SERVICE OF APPLICANT IN INDIA	Name	**SUDARSHANA BANDYOPADHYAY**
	Postal Address	Ground Floor, S-456, LGF, Greater Kailash – II, New Delhi – 110048, India
	Telephone No.	NA
	Mobile No.	97488 18235
	Fax No.	NA
	E-mail ID	bandyopadhyay.sudarshana@gmail.com

8. IN CASE OF APPLICATION CLAIMING PRIORITY OF APPLICATION FILED IN CONVENTION COUNTRY, PARTICULARS OF CONVENTION APPLICATION

Country	Application Number	Filing date	Name of the applicant	Title of the invention	IPC (as classified in the convention country)

N.A.					

9. IN CASE OF PCT NATIONAL PHASE APPLICATION, PARTICULARS OF INTERNATIONAL APPLICATION FILED UNDER PATENT CO-OPERATION TREATY (PCT)

International application number	International filing date

10. IN CASE OF DIVISIONAL APPLICATION FILED UNDER SECTION 16, PARTICULARS OF ORIGINAL (FIRST) APPLICATION

Original (first) application No.	Date of filing of original (first) application
N.A.	

11. IN CASE OF PATENT OF ADDITION FILED UNDER SECTION 54, PARTICULARS OF MAIN

Main application/patent No.	Date of filing of main application
N.A.	N.A.

12. DECLARATIONS

(i) Declaration by the inventor(s)

(**In case the applicant is an assignee**: the inventor(s) may sign herein below or the applicant may upload the assignment or enclose the assignment with this application for patent or send the assignment by post/electronic transmission duly authenticated within the prescribed period).

We, the above-named inventor(s) is/are the true & first inventor(s) for this Invention and declare that the applicant(s) herein is/are my/our assignee or legal representative.

(a) Date:
(b) Signature:
(c) Name: **Dr Soumya Ranjan Jena**

(a) Date
(b) Signature(s):
(c) Name: **Mr Sanjoy Saha**

(a) Date:
(b) Signature:
(c) Name: **Dr Sohit Agarwal**

(ii) Declaration by the applicant(s) in the convention country

(In case the applicant in India is different than the applicant in the convention country: the applicant in the convention country may sign herein below or applicant in India may upload the assignment from the applicant in the convention country or enclose the said assignment with this application for patent or send the assignment by post/electronic transmission duly authenticated within the prescribed period)

I/We, the applicant(s) in the convention country declare that the applicant(s) herein is/are my/our assignee or legal representative. – **N.A.**

(a) Date
(b) Signature(s)
(c) Name(s) of the signatory

(iii) Declaration by the applicant

We the applicant hereby declare that: -

[✓] We are in possession of the above-mentioned invention.

[✓] The complete specification relating to the invention is filed with this application.

[**x**] The invention as disclosed in the specification uses the biological material from India and the necessary permission from the competent authority shall be submitted by me/us before the grant of patent to me/us.

[✓] There is no lawful ground of objection(s) to the grant of the Patent to us.

[**x**] We are the true & first inventor(s).

[✓] We are the assignee or legal representative of true & first inventor(s).

[**x**] The application or each of the applications, particulars of which are given in Paragraph-8, was the first application in convention country in respect of my invention(s).

[**x**] We claim the priority from the above mentioned application(s) filed in convention country/countries and state that no application for protection in respect of the invention had been made in a convention country before that date by us or by any person from which I derive the title.

[**x**] Our application in India is based on international application under Patent Cooperation Treaty (PCT) as mentioned in Paragraph-9.

[**x**] The application is divided out of my /our application particulars of which is given in Paragraph-10 and pray that this application may be treated as deemed to have been filed on DD/MM/YYYY under section 16 of the Act.

[**x**] The said invention is an improvement in or modification of the invention particulars of which are given in Paragraph-11.

13. FOLLOWING ARE THE ATTACHMENTS WITH THE APPLICATION

(a) Form 2

Item	*Details*	*Fee*	*Remarks*
Complete/ provisional specification	No. of pages: 28	1600 + 7 x 160	Including Form 2, description, claim pages + abstract + drawings sheets
No. of Claim(s)	No. of Claims = 22 No. of Pages = 5	12 x 320	Claim pages
Abstract	1		Abstract page
No. of Drawing(s)	No. of drawings = 3 and No. of pages = 3		Drawing sheets

# In case of a complete specification, if the applicant desires to adopt the drawings filed with his provisional specification as the drawings or part of the drawings for the complete specification under rule 13(4), the number of such pages filed with the provisional specification are required to be mentioned here.

b. Form 3: Statement and Undertaking
c. Form 5: Declaration as to inventorship
d. Power of Attorney
e. Form 28
f. Form 9

Total fee ₹ 9060/- is being paid online through electronic portal

We hereby declare that to the best of our knowledge, information and belief the fact and matters stated herein are correct and we request that a patent may be granted to us for the said invention.

Dated this 30th day of September 2025.

Signature:

Sudarshana

Name: Sudarshana Bandyopadhyay
(Regn No: IN/PA 2802)
Agent for the Applicant
Phn no.: 97488 18235
email: bandyopadhyay.sudarshana@gmail.com

To,
The Controller of Patents
The Patent Office,
at Kolkata

FORM 28

THE PATENTS ACT,

1970 (39 of 1970)

AND

THE PATENTS RULES, 2003

TO BE SUBMITTED BY A SMALL ENTITY /STARTUP/EDUCATIONAL INSTITUTION

[See rules 2 (fa), 2(fb), 2(ca) and 7]

1	Name, address and nationality.	We, SRJX RESEARCH AND INNOVATION LAB LLP, of the address Plot No - 3E/474, Sector-9, CDA, Post- Markat Nagar, Cuttack-753014, Odisha, India, applicant in respect of the patent application no. __________________ dated 30 September 2025 hereby declare that we are a micro entity in accordance with rule 2(fa) and submit the following document as a proof :
2	Documents to be submitted	
	i. For claiming the status of a **micro** entity:	
	A. For an Indian applicant: Evidence of registration under the Micro, Small and Medium Enterprises Act, 2006 (27 of 2006).	
3	To be signed by the applicant(s) / patentee (s) / authorised registered patent agent.	The information provided herein is correct to the best of my/our knowledge and belief. Dated this 30th day of September 2025
4	Name of the natural person who has signed.	Signature: Sudarshana

Signature Not Verified

Digitally Signed.
Name: Sudarshana Bandyopadhyay
Date: 30-Sep-2025 22:20:14
Reason: Patent Efiling

	Designation and official seal, if any, of the person who has signed.	Sudarshana Bandyopadhyay Regn. No.: IN/PA 2802 Agent for the applicant Phn No. 9748818235 Email: bandyopadhyay.sudarshana@gmail.com To The Controller of Patents, The Patent Office, At Kolkata

FORM 9

THE PATENTS ACT, 1970

(39 of 1970)

&

THE PATENTS RULES, 2003

REQUEST FOR PUBLICATION

[See Section 11A(2); Rule 24A]

We, SRJX RESEARCH AND INNOVATION LAB LLP, of the address Plot No - 3E/474, Sector-9, CDA, Post- Markat Nagar, Cuttack-753014, Odisha, India, hereby request for an early publication of our Patent Application No. ____________________ filed on 30 September 2025 under Section 11A(2) of the Act.

Dated this 30th day of September 2025

Sudarshana

Sudarshana Bandyopadhyay
Regn No.: IN/PA 2802
Agent for the Applicants
Email: bandyopadhyay.sudarshana@gmail.com
Phn No: 9748818235

Signature Not Verified
Digitally Signed.
Name: Sudarshana
Bandyopadhyay
Date: 30-Sep-2025 22:15:42
Reason: Patent Efiling

सत्यमेव जयते

INDIA NON JUDICIAL

Government of National Capital Territory of Delhi

₹100

e-Stamp

₹100₹100₹100₹100₹100

Certificate No.	:	IN-DL35961746213944X
Certificate Issued Date	:	16-Aug-2025 11:10 AM
Account Reference	:	IMPACC (IV)/ dl962703/ DELHI/ DL-ESD
Unique Doc. Reference	:	SUBIN-DLDL96270305293890128756X
Purchased by	:	SRJX RESEARCH AND INNOVATION LAB LLP
Description of Document	:	Article 48(c) Power of attorney - GPA
Property Description	:	Not Applicable
Consideration Price (Rs.)	:	0 (Zero)
First Party	:	SRJX RESEARCH AND INNOVATION LAB LLP
Second Party	:	ZAINAB SYED AND ASSOCIATES
Stamp Duty Paid By	:	SRJX RESEARCH AND INNOVATION LAB LLP
Stamp Duty Amount(Rs.)	:	100 (One Hundred only)

NOTARY
R.V. Singh
Advocate
Reg. No.2282
Commission
Expire On
24-10-2027
Area-Delhi
GOVT. OF INDIA

Please write or type below this line IN-DL35961746213944X

NOTARY
R.V. Singh
Advocate
Reg. No.2282
Commission
Expire On
24-10-2027
Area-Delhi
GOVT. OF INDIA

Statutory Alert:

1. The authenticity of this Stamp certificate should be verified at 'www.shcilestamp.com' or using e-Stamp Mobile App of Stock Holding. Any discrepancy in the details on this Certificate and as available on the website / Mobile App renders it invalid.
2. The onus of checking the legitimacy is on the users of the certificate.
3. In case of any discrepancy please inform the Competent Authority.

Signature Not Verified

Digitally Signed.
Name: Sudarshana Bandyopadhyay
Date: 30-Sep-2025 21:50:52
Reason: Patent Efiling

FORM-26
The Patents Act, 1970
(39 of 1970)
FORM FOR AUTHORIZATION OF A PATENT AGENT/OR ANY PERSON IN A MATTER OR PROCEEDING UNDER THE ACT
[See Sections 127 and 132; Rule 135]

I, **SRJX RESEARCH AND INNOVATION LAB LLP**, Indian, of the address SRJX RESEARCH AND INNOVATION LAB LLP, Plot No - 3E/474, Sector-9, CDA, Post- Markat Nagar, Cuttack-753014, Odisha, India, hereby authorize **Zainab Syed & Associates** having address 3E, Nawab Bhagwanpora, Lal Bazar, Srinagar, Jammu & Kashmir, 190023, India **(Mobile No.: +91 9748818235, Email: bandyopadhyay.sudarshana@gmail.com)** through **Ms. Sudarshana Bandyopadhyay (IN/PA 2802)** and **Ms. Meenu Sharma (IN/PA-2856),** registered Indian Patent Agents, to act on our behalf and to further appoint attorney(s)/agent(s) in connection with the filing and prosecution of our patent applications for grant of Letters Patent, filing of request for examination, filing request for amendment, recordal of change of name and address, ownership, change of address of service in India, renewal of patent, recordal of assignments, filing and defending oppositions and infringement actions, restoration of patents, registration of documents and such other actions and all proceedings under the Patents Act, 1970 and the Patent Rules, 2003 and all such proceedings before the Patent Office or the Government of India or any Court in India and all acts and things as the said attorney may deem necessary or expedient in connection therewith or incidental thereto.

We further request that all notices, requisitions and communication relating thereto may be sent to such person/s at the corresponding address mentioned below:

Ground Floor, S-456, LGF, Greater Kailash – II, New Delhi – 110048, India,

(Contact No.: +91 9748818235; Email: bandyopadhyay.sudarshana@gmail.com)

We, hereby, revoke all previous authorizations, if any, in respect of the proceedings.

NOTARY
R.V. Singh
Advocate
Reg. No.2282
Commission
Expire On
24-10-2027
Area-Delhi
GOVT. OF INDIA

We, hereby, assent to the action already taken by the said person/s in the above matter.

Dated this 14th **day of August, 2025**

SRJX RESEARCH AND INNOVATION LAB LLP
Through:

Signature: Soumya Ranjan Jena

Name: Dr. Soumya Ranjan Jena

Company Seal: SRJX Research and Innovation Lab LLP
LLPIN: ACO-1435

To,
The Controller of Patents,
The Patent Office,
Kolkata

NOTARY
R.V. Singh
Advocate
Reg. No.2282
Commission
Expire On
24-10-2027
Area-Delhi
GOVT. OF INDIA

ATTESTED
Notary Public Delhi

16 AUG 2025

FORM 9

THE PATENTS ACT, 1970

(39 of 1970)

&

THE PATENTS RULES, 2003

REQUEST FOR PUBLICATION

[See Section 11A(2); Rule 24A]

We, SRJX RESEARCH AND INNOVATION LAB LLP, of the address Plot No - 3E/474, Sector-9, CDA, Post- Markat Nagar, Cuttack-753014, Odisha, India, hereby request for an early publication of our Patent Application No. ____________________ filed on 30 September 2025 under Section 11A(2) of the Act.

Dated this 30th day of September 2025

Sudarshana

Sudarshana Bandyopadhyay
Regn No.: IN/PA 2802
Agent for the Applicants
Email: bandyopadhyay.sudarshana@gmail.com
Phn No: 9748818235

Signature Not Verified
Digitally Signed.
Name: Sudarshana
Bandyopadhyay
Date: 30-Sep-2025 20:03:59
Reason: Patent Efiling

FORM 3

THE PATENTS ACT,
1970 (39 of 1970)
and
THE PATENTS RULES,
2003

STATEMENT AND UNDERTAKING UNDER SECTION 8

(See section 8; Rule 12)

1. Name of the applicant(s).	We, SRJX RESEARCH AND INNOVATION LAB LLP, Plot No - 3E/474, Sector-9, CDA, Post-Markat Nagar, Cuttack-753014, Odisha, India hereby declare:
2. Name, address and nationality of the joint applicant.	(i) that we have not made any application for the same/substantially the same invention outside India Or (ii) that we who have made this application No date 30th September 2025 alone/~~jointly with~~ ………, made for the same/ substantially same invention, application(s) for patent in the other countries, the particulars of which are given below:

Name of the country	Date of application	Application No.	Status of the application	Date of publication	Date of grant
N.A.					

3. Name and address of the assignee	(iii) that the rights in the application(s) have been assigned to SRJX RESEARCH AND INNOVATION LAB LLP, Plot No - 3E/474, Sector-9, CDA, Post-Markat Nagar, Cuttack-753014, Odisha, India

Signature Not Verified
Digitally Signed.
Name: Sudarshana Bandyopadhyay
Date: 30-Sep-2025 20:03:59
Reason: Patent Efiling

	that we undertake that upto the date of grant of the patent by the Controller, we would keep him informed in writing the details regarding corresponding applications for patents filed outside India within six months from the date of filing of such application. Dated this 30th day of September 2025
4. To be signed by the applicant or his authorized registered patent agent.	Sudarshana Signature. …………………
5. Name of the natural person who has signed.	Sudarshana Bandyopadhyay Regn. No.: IN/PA 2802 Agent for the applicant Phn No. 9748818235 Email: bandyopadhyay.sudarshana@gmail.com
	To The Controller of Patents, The Patent Office, at Kolkata
Note.- Strike out whichever is not applicable;	