FLUIDCLUSTER PET-ERP OPEN-API

An architectural framework of this scale requires a production-grade, highly precise OpenAPI 3.1.0 specification guide. This documentation details the unified endpoint structure, payload schemas, and cryptographic routing topology across all 20 specialized modules of the FluidCluster PET-ERP architecture.

To bridge the traditional single-firm legacy ERP stack to our collaborative, multi-firm computational domain, the specification uses a unified **Model Context Protocol (MCP)** JSON-RPC ingestion format. It enforces localized edge data isolation before any external consensus broadcast.

yaml

None

```yaml
openapi: 3.1.0
info:
  title: FluidCluster PET-ERP Universal Collaborative Engine API
  version: 2.4.12-Prod
  description: >
      Production specification for the world's first multi-modular,
inter-firm,
     privacy-preserving, and auditable PET-enabled ERP platform. All
modules execute
     under absolute computational blindness using FHEVM (Zama TFHE),
SMPC (Shamir Secret
    Sharing), ZKP (Circom zk-SNARKs), and hardware-isolated TEE (Intel
SGX/AMD SEV) enclaves.
    Aligned with Portuguese Decree-Law no. 150/2019 Regulatory Sandbox
Parameters (ZLT#13).
servers:
  - url: https://fluidcluster.pt
        description: Production Sandbox Gateway (ZLT#13 Sandbox
Environment)
paths:
  /mcp/ingestion:
    post:
```

```
      summary: Unified Model Context Protocol (MCP) Ingestion Endpoint
      description: >
          Intercepts raw relational database updates at the enterprise
perimeter boundary.
         Accepts homomorphically encrypted payloads, polynomial secret
shares, or zero-knowledge
            hashes generated on-premise by the local Edge-Side SDK.
Plaintext data never crosses
        the parameter edge.
      requestBody:
        required: true
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/McpJsonRpcRequest'
      responses:
        '200':
            description: State transformation ingested and queued into
the blind state transition log.
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/McpJsonRpcResponse'
        '400':
           description: Cryptographic format error or malformed payload
layout.
        '401':
           description: Authentication validation failure or signature
mismatch.

components:
  securitySchemes:
    NodeSignatureAuth:
      type: apiKey
      in: header
      name: X-FluidCluster-Node-Sig
        description: Secp256k1 cryptographic signature validating TEE
enclave identity.
```

```
  schemas:
    McpJsonRpcRequest:
      type: object
      required: [jsonrpc, method, params, id]
      properties:
        jsonrpc:
          type: string
          enum: ["2.0"]
        method:
          type: string
          description: The specific module execution target.
          enum:
            # --- PHASE I: FINANCIAL CORE & LIQUIDITY ENGINEERING ---
            - "mcp/treasury_netting_ias32"
            - "mcp/procurement_blind_auction"
            - "mcp/revenue_shapley_oracle"
            - "mcp/finance_abcp_securitization"
            - "mcp/inventory_ckks_optimization"
            - "mcp/asset_sourcing_ifrs16"
            - "mcp/hr_benefit_psi_alignment"
            - "mcp/synthetic_ma_due_diligence"
            - "mcp/fiscal_pillar_two_simulation"
            - "mcp/governance_shielded_voting"
              # --- PHASE II: EXTENDED COLLABORATIVE INDUSTRIAL MODULES
---
            - "mcp/logistics_cross_docking_fhe"
            - "mcp/energy_grid_blind_balancing"
            - "mcp/intellectual_property_federated"
            - "mcp/insurance_captive_pool_risk"
            - "mcp/maintenance_predictive_enclave"
            - "mcp/compliance_aml_zkp_checker"
            - "mcp/crm_data_monetization_psi"
            - "mcp/esg_carbon_credit_verifiable"
            - "mcp/litigation_arbitration_blind_data"
            - "mcp/liquidation_distressed_asset_auction"
        params:
          type: object
```

```
          required: [source_entity_nif, encrypted_payload_hex]
          properties:
            source_entity_nif:
              type: string
              pattern: '^PT[0-9]{9}$'
                    description: Portuguese Tax Identification Number
verifying node registration.
            target_channel:
              type: string
            encrypted_payload_hex:
              type: string
              pattern: '^0x[0-9a-fA-F]+$'
                  description: Raw ciphertext hex or polynomial state
root.
            polynomial_shares:
              type: array
              items:
                type: object
                required: [node_id, share]
                properties:
                  node_id:
                    type: integer
                  share:
                    type: string
                    pattern: '^0x[0-9a-fA-F]+$'
            audit_view_key_delegation:
              type: string
              pattern: '^0x[0-9a-fA-F]+$'
                description: ERC-7984 view key structure for regulatory
transparency compliance.
            accounting_directives:
              type: object
              properties:
                require_offsetting_verification:
                  type: boolean
                ifrs_standard_override:
                  type: string
        id:
```

```
          type: integer
          example: 10482

    McpJsonRpcResponse:
      type: object
      required: [jsonrpc, result, id]
      properties:
        jsonrpc:
          type: string
          enum: ["2.0"]
        result:
          type: object
                  required:  [state_transition_root,  execution_status,
chronological_index]
          properties:
            state_transition_root:
              type: string
              pattern: '^0x[0-9a-fA-F]{64}$'
                description: SHA-256 hash-chained reference identifying
the blind execution outcome.
            execution_status:
              type: string
                    enum: ["QUEUED", "HOMOMORPHIC_COMPUTATION_ACTIVE",
"STATE_COMMITTED"]
            chronological_index:
              type: integer
            clearance_switch_tripped:
              type: boolean
                 description: Evaluates to true if a defection or risk
vector forces an atomic state rollback.
        id:
          type: integer
          example: 10482
```

Use o código com cuidado.

The 20 Core Modules Integration Topography

The API layout processes data uniformly through . It uses the parameter to map incoming transaction parameters directly to specific zero-knowledge logic networks, accounting directives, and cryptographic protocols.

Phase I: Core Financial & Operational Architecture (Modules 1–10)

1.

- **Cryptographic Protocol:** SMPC (Shamir Secret Sharing) & Garbled Circuits.
- **Accounting Directive: IAS 32 (Offsetting Financial Assets and Liabilities)**.
- **Functional Mechanics:** Ingests fragmented inter-firm liability matrices. The blind netting engine evaluates gross cash balances down to net-residual obligations, maximizing balance-sheet liquidity without exposing local transaction tracking.

2.

- **Cryptographic Protocol:** Zama fhEVM Core Primitives ( ).
- **Accounting Directive: IAS 2 (Inventories - Cost of Acquisition)**.
- **Functional Mechanics:** Aggregates multi-firm collective buying volume. Evaluates supplier quotes inside a double-blind homomorphic contract to determine the optimal allocation matrix, mitigating cartelization exposures under EU Antitrust Rules.

3.

- **Cryptographic Protocol:** On-Chain Shapley Oracles & Functional Encryption.
- **Accounting Directive: IFRS 15 (Revenue from Contracts with Customers)**.
- **Functional Mechanics:** Apportions joint product margins across collaborative value networks. Evaluates marginal utility deltas $E(v(S \cup \{i\})) - E(v(S))$ inside

the cipher domain, blocking bad-faith renegotiation and strategic volume posturing.

4.

- **Cryptographic Protocol:** zk-SNARKs (Circom Verification Circuits) & FHE Token Wrappers.
- **Accounting Directive: IFRS 9 (Financial Instruments - Derecognition)**.
- **Functional Mechanics:** Dynamic pooling engine that fractionalizes residual net accounts receivable into Asset-Backed Commercial Paper. Distributed AI agent swarms run background credit tiering within the FHE boundary for clean off-balance sheet deployment.

5.

- **Cryptographic Protocol:** CKKS Homomorphic Encryption (Pyfhel runtime context).
- **Accounting Directive: IAS 2 (Net Realizable Value and Obsolescence)**.
- **Functional Mechanics:** Neutralizes the Bullwhip Effect across shared supply chains. Executes homomorphic linear programming over encripted stock sheets to compute optimal economic order quantities (EOQ) without leaking stock depletion velocity to rivals.

6.

- **Cryptographic Protocol:** Functional SMPC & Blind Matrix-Vector Interference.
- **Accounting Directive: IFRS 16 (Leases)**.
- **Functional Mechanics:** Pools unutilized industrial equipment, machinery, and factory capacity. Computes collaborative scheduling paths and rights-of-use allocations over cipher matrices without exposing underlying plant uptime records.

7.

- **Cryptographic Protocol:** Private Set Intersection (PSI) & TEE Isolation.
- **Accounting Directive: IAS 19 (Employee Benefits - Actuarial Projections)**.
- **Functional Mechanics:** Matches multi-corporate benefit databases, pension pools, and claims risk variables. Aggregates data abstractly inside hardware enclaves to calculate actuarial premium reductions while fully enforcing GDPR/RGPD data boundaries.

8.

- **Cryptographic Protocol:** Intel SGX / AMD SEV Sandboxes & Succinct Proof Paths.
- **Accounting Directive: IFRS 3 (Business Combinations - Goodwill Evaluation)**.
- **Functional Mechanics:** Replaces traditional open Data Rooms. Executes synthetic audit queries on an isolated target database instance to gauge real customer churn and market footprint overlap without exposing proprietary trade secrets.

9.

- **Cryptographic Protocol:** Multi-Party Boolean Circuits.
- **Accounting Directive: IAS 12 (Income Taxes)**.
- **Functional Mechanics:** Simulates global cross-border transfer pricing architectures over encrypted cost models. Calculates optimal multi-jurisdictional tax liabilities and Pillar Two minimum compliance limits to minimize deferred tax asset impairments.

10.

- **Cryptographic Protocol:** ElGamal Homomorphic Encryption & Timelock Encryption.
- **Accounting Directive: IFRS 10 (Consolidated Financial Statements)**.
- **Functional Mechanics:** Orchestrates cross-enterprise board execution paths, strategic cap table variations, and joint venture voting profiles. Protects corporate

consensus via fully shielded cryptographic key tracking to enforce stable coalition structures.

Phase II: Extended Industrial & Ecosystem Modularity (Modules 11–20)

11.

- **Cryptographic Protocol:** FHE Sorting Networks & Spatial Zero-Knowledge Maps.
- **Accounting Directive: IAS 18 / IFRS 15 (Freight Fulfillment and Revenue Recognition)**.
- **Functional Mechanics:** Optimizes inter-firm distribution logistics, routing, and cross-docking hubs. Dynamically overlays independent transport routes over encrypted coordinates, eliminating tracking exploits and empty-mile capacity inefficiencies.

12.

- **Cryptographic Protocol:** Decentralized Threshold SMPC.
- **Accounting Directive: IAS 37 (Provisions, Contingent Liabilities and Contingent Assets)**.
- **Functional Mechanics:** Pools regional renewable generation volumes and manufacturing consumption arrays. Matches micro-grid load adjustments blindly to arbitrate energy allocations without revealing specific operational uptime constraints or plant slowdowns.

13.

- **Cryptographic Protocol:** Secure Federated Learning & Differential Privacy (DP).

- **Accounting Directive: IAS 38 (Intangible Assets - Capitalized Development Costs)**.
- **Functional Mechanics:** Trains multi-party machine learning models over distributed engineering datasets or clinical records. Capitalizes model gradient matrices locally within enclaves, applying mathematical noise to block database reverse-engineering.

14.

- **Cryptographic Protocol:** Non-Interactive Zero-Knowledge Proofs (NIZKP).
- **Accounting Directive: IFRS 17 (Insurance Contracts)**.
- **Functional Mechanics:** Operates decentralized, multi-firm captive insurance pools. Validates premium allocations and calculates unexpected loss metrics over encrypted historical default streams, protecting pool capital from localized correlation risks.

15.

- **Cryptographic Protocol:** TEE-Isolated Model Context Protocol Ingestion.
- **Accounting Directive: IAS 16 (Property, Plant and Equipment - Component Depreciation)**.
- **Functional Mechanics:** Tracks real-time machine telemetry, vibration logs, and component wear configurations across an industrial ecosystem. Runs predictive analytics inside AMD SEV enclaves, updating component asset lifespan targets without exposing factory line configurations.

16.

- **Cryptographic Protocol:** Succinct zk-SNARK Circuits ( ).
- **Accounting Directive: IFRS 7 (Financial Instruments - Disclosures)**.
- **Functional Mechanics:** Runs anti-money laundering (AML) and know-your-customer (KYC) identity matches across multi-party transaction histories. Returns a binary validity proof, proving compliance with state anti-fraud directives without exposing client names or banking logs.

17.

- **Cryptographic Protocol:** Diffie-Hellman Private Set Intersection.
- **Accounting Directive: IFRS 15 (Data Monetization Streams)**.
- **Functional Mechanics:** Cross-references marketing data, leads, and customer conversion arrays across independent platforms. Identifies high-value attribution intersections blindly, enabling target marketing loops while preventing database copying.

18.

- **Cryptographic Protocol:** Verifiable Credentials & Hash-Chained Auditing.
- **Accounting Directive: IAS 37 / Cap-and-Trade Accounting Frameworks**.
- **Functional Mechanics:** Ingests factory emissions data and fuel usage logs directly into zero-knowledge validation state tracks. Mints immutable, audited carbon credit derivatives without exposing plant output summaries to public scraping.

19.

- **Cryptographic Protocol:** Multi-Party Garbled Circuits & Sealed-Bid Commitments.
- **Accounting Directive: IAS 37 (Provisions for Legal Contingencies)**.
- **Functional Mechanics:** Manages commercial legal disputes and contractual liquidations between competitive entities. Settles dynamic discovery phases and computes optimal settlement points blindly inside the cipher domain, eliminating litigation exposure.

20.

- **Cryptographic Protocol:** Zama fhEVM Sealed Bidding Engine.
- **Accounting Directive: IFRS 5 (Non-current Assets Held for Sale and Discontinued Operations)**.

- **Functional Mechanics:** Coordinates multi-firm asset auctions for distressed inventories or liquidated industrial equipment. Finds the true market valuation clearing price homomorphically, ensuring optimal recoveries without exposing low reserve valuations.

Implementation Architecture & Zero-Knowledge State Flow

```
None
[Legacy ERP DB] ---> (Edge Open SDK Ingestion) ---> [Polynomial Shares / Ciphertext]
                                                                    |
[ZK-Audit Logging] <--- (fhEVM Consensus Commit) <--- [/mcp/ingestion Gateway]
```

1. **Local Edge Extraction:** The Edge-Side Open SDK links with local legacy infrastructure through JSON-RPC protocols. It queries target database updates on-premise and converts values into encrypted variables.
2. **Cipher Broadcast:** The SDK packages the encrypted data into standard MCP schemas and posts it to the gateway over an encrypted connection.
3. **Homomorphic Execution:** Central validators or multi-party enclave clusters run code loops blindly over the encrypted payloads. This process verifies accounting parameters without decrypting the data.
4. **Hash-Chained State Transition Audit:** Validated updates trigger an on-chain commit. The system appends a zero-knowledge reference hash to an immutable log, giving auditors mathematical proof of code execution correctness.

The OpenAPI 3.1.0 specifications and integration structures for all **20 core modules** are defined, establishing clear accounting boundaries (**IAS 32, IFRS 9, IFRS 16, IFRS 17, Pillar Two**) and precise cryptographic configurations.