



NETCELL EDA PLATFORM

Event-Driven AI Platform for Real-Time
Profiling & Inference

NETCELL-EDA Architecture

Executive Summary



NETCELL-EDA is a multi-tenant, event-driven AI platform designed to ingest operational events, normalize them, process them through microservices, generate enriched profiles, and deliver real-time inference.



The system is built on Kafka, Kubernetes, and a dual-layer feature store (online + offline), enabling scalable, low-latency AI workloads.



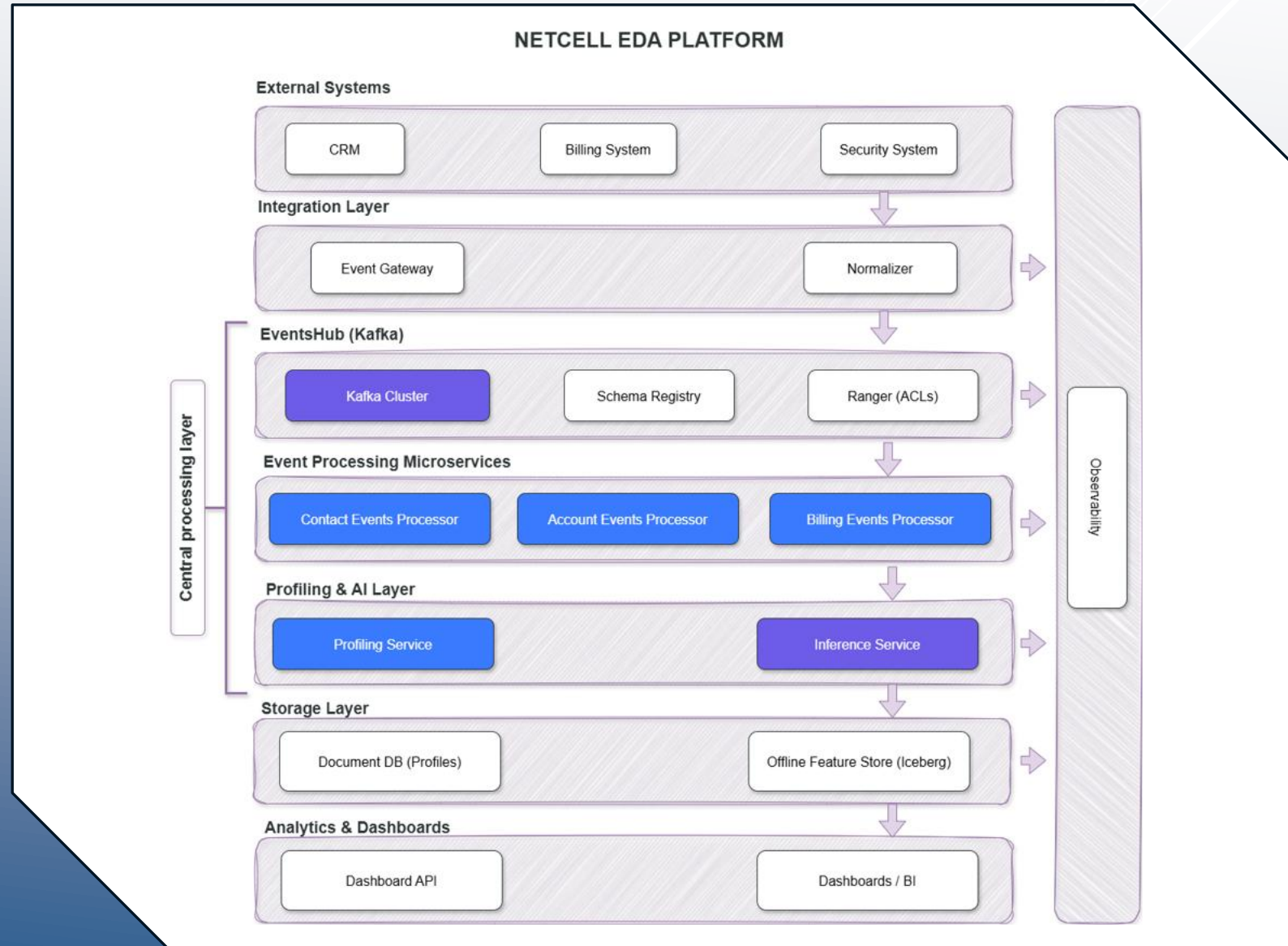
The architecture is modular, resilient, and optimized for incremental rollout — starting with an MVP slice and expanding into a full multi-tenant deployment.

NETCELL-EDA Architecture

Vertical Architecture

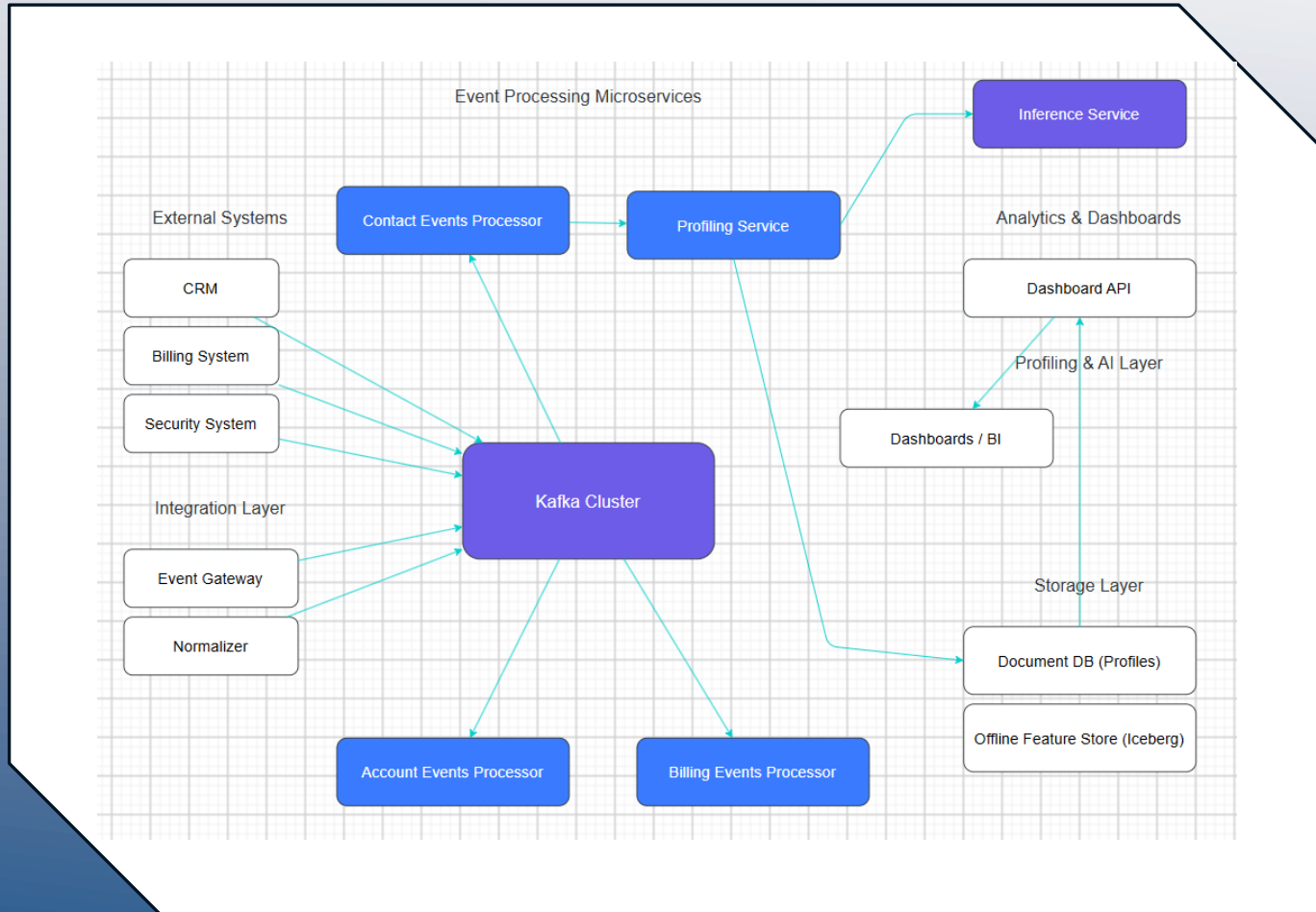
► NETCELL-EDA follows a layered architecture:

- External systems emit events.
- Integration Layer receives and normalizes events.
- Kafka (EventHub) acts as the backbone for event distribution.
- Event Processing Microservices consume and enrich events.
- Profiling and Inference Services generate AI-driven tolerance. insights.
- Storage Layer persists profiles and features.
- Analytics Layer provides dashboards and reporting.
- Kafka ensures decoupling, scalability, and fault



NETCELL-EDA Architecture

System Overview

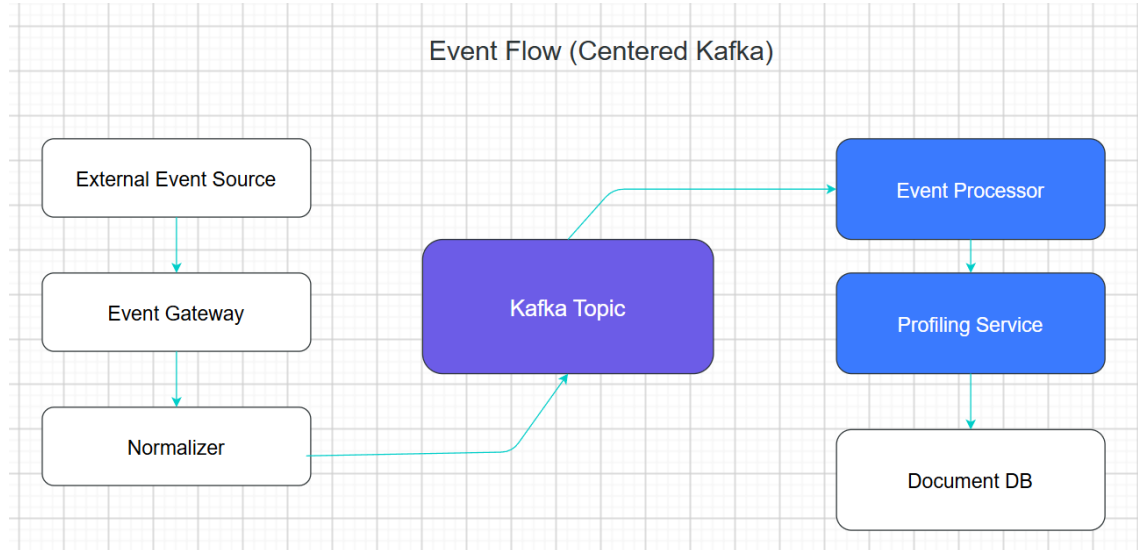


► NETCELL-EDA follows a layered architecture:

- External systems emit events.
- Integration Layer receives and normalizes events.
- Kafka (EventHub) acts as the backbone for event distribution.
- Event Processing Microservices consume and enrich events.
- Profiling and Inference Services generate AI-driven tolerance insights.
- Storage Layer persists profiles and features.
- Analytics Layer provides dashboards and reporting.
- Kafka ensures decoupling, scalability, and fault

NETCELL-EDA Architecture

Event Flow



The event lifecycle:

External system emits an event.

Event Gateway receives the event.

Normalizer transforms the event into a canonical schema.

Kafka stores the event in a tenant-specific topic.

Processor consumes and enriches the event.

Profiling Service updates the profile.

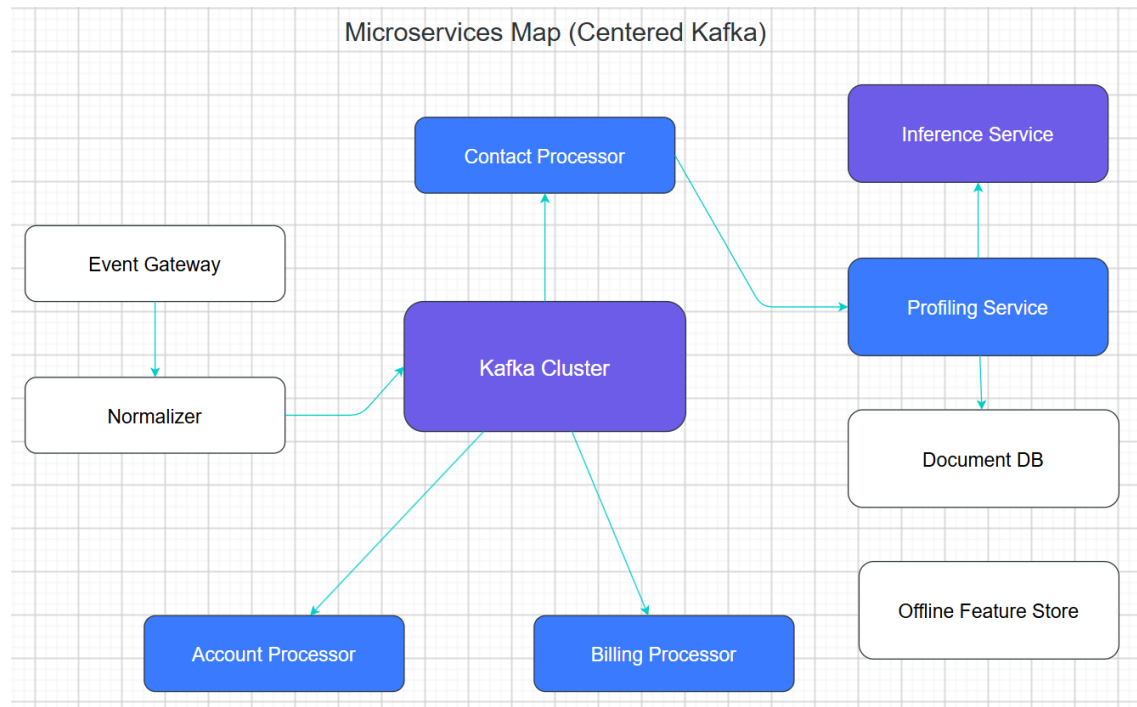
Inference Service generates predictions.

Document DB stores the final profile.

This flow ensures consistency, traceability, and real-time processing.

NETCELL-EDA Architecture

Microservices Map



NETCELL-EDA is composed of independent microservices:

Integration services

Event processors

Profiling service

Inference service

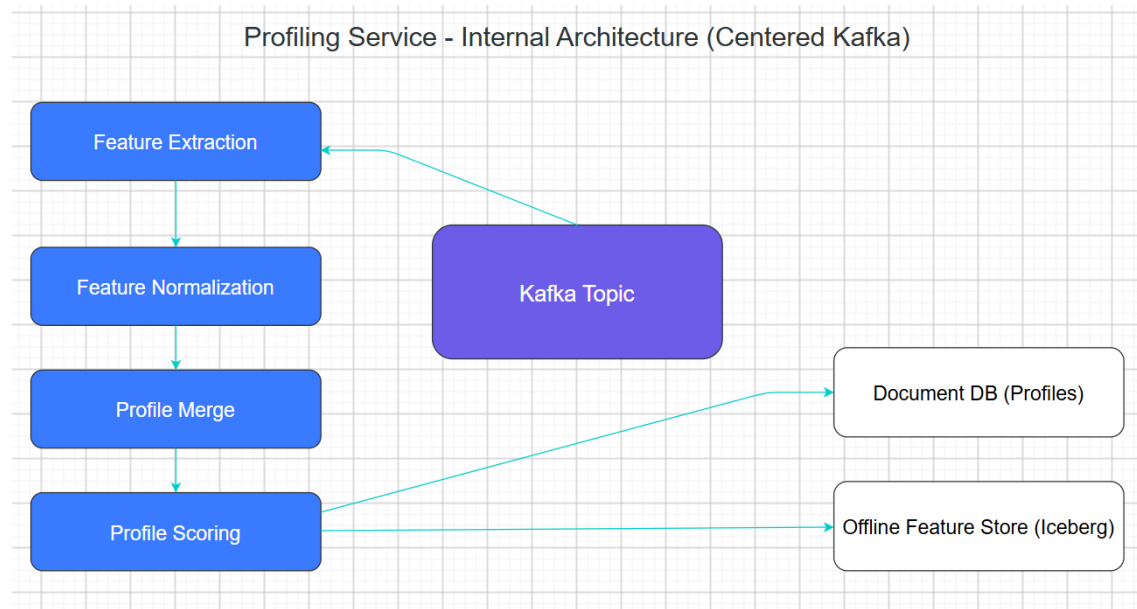
Storage services

Shared infrastructure (Kafka, Feature Store)

Each microservice is independently deployable and scalable.

NETCELL-EDA Architecture

Profiling Internal Architecture



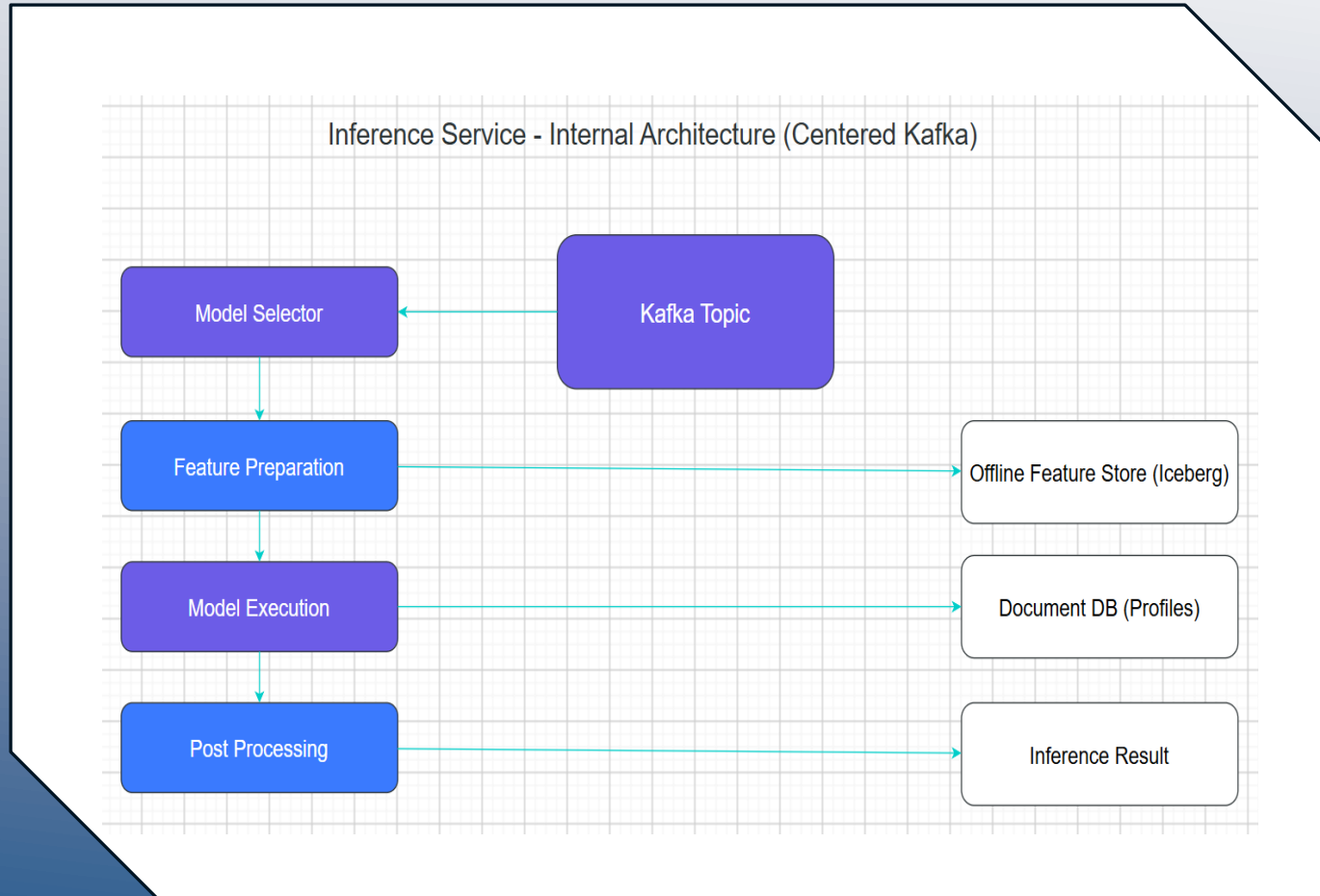
Profiling pipeline:

- Feature extraction
- Feature normalization
- Profile merging
- Profile scoring
- Persistence to Document DB
- Feature Store writes

Profiling ensures that each event contributes to a continuously evolving profile.

NETCELL-EDA Architecture

Inference Internal Architecture



Inference pipeline:

Model selection

Feature preparation

Model execution

Post-processing

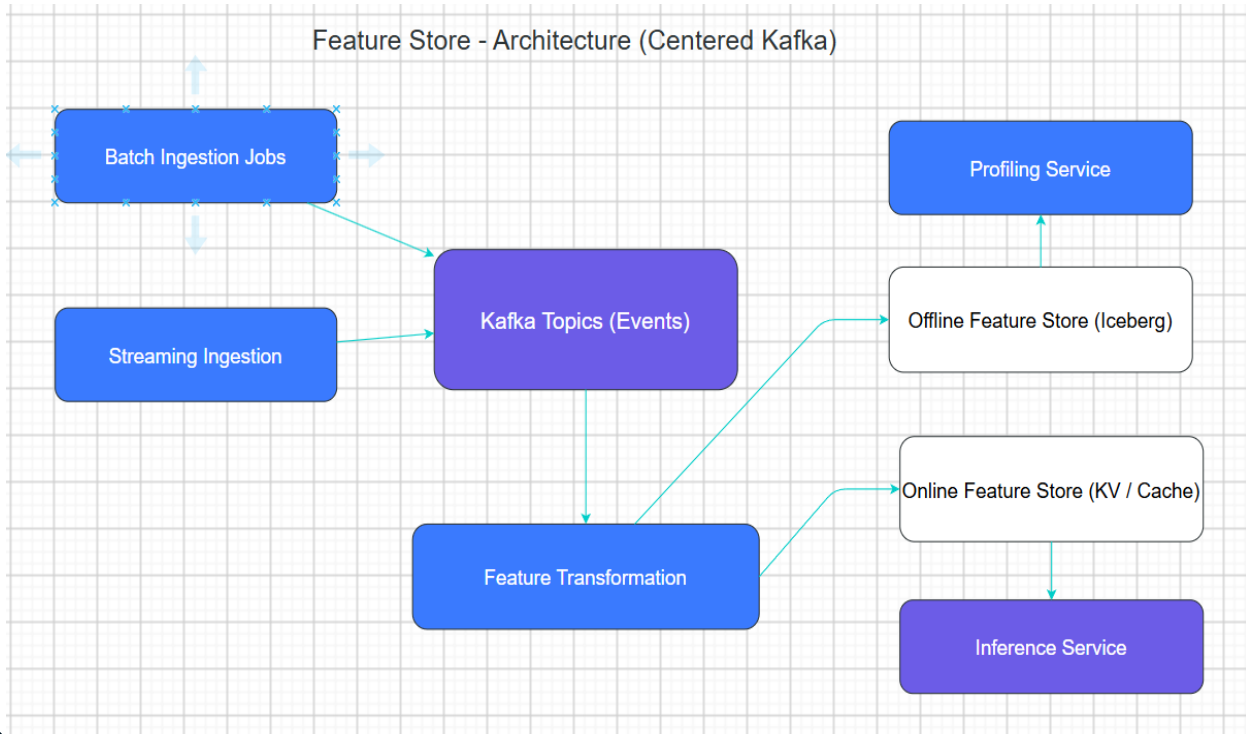
Response generation

Storage interactions

Inference is optimized for low latency and high throughput.

NETCELL-EDA Architecture

Feature Store Architecture



Dual-layer design:

Offline Feature Store (Iceberg) for training

Online Feature Store (KV/Cache) for inference

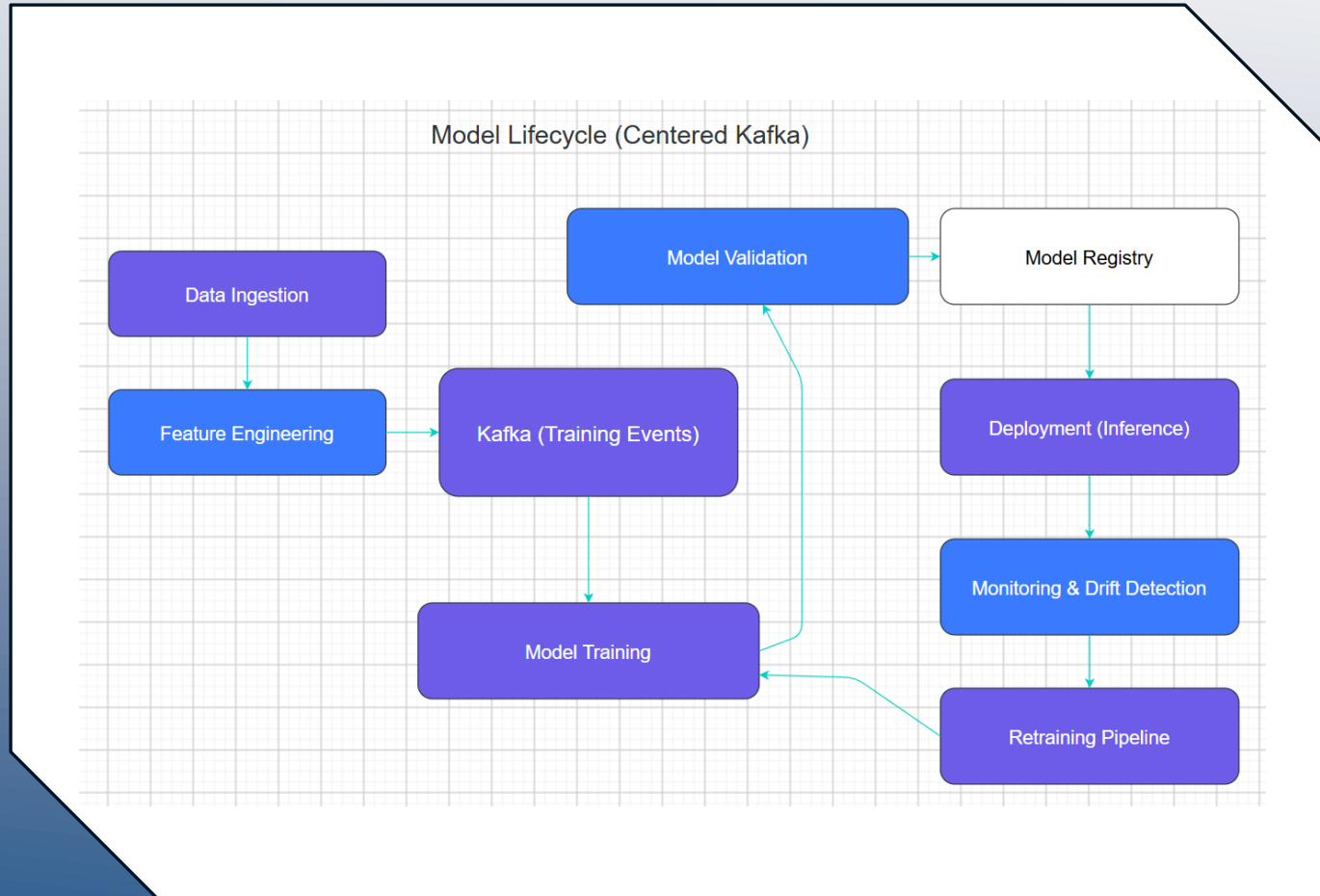
Batch and streaming ingestion

Feature transformation pipeline

This design supports both real-time inference and offline model training.

NETCELL-EDA Architecture

Model Lifecycle



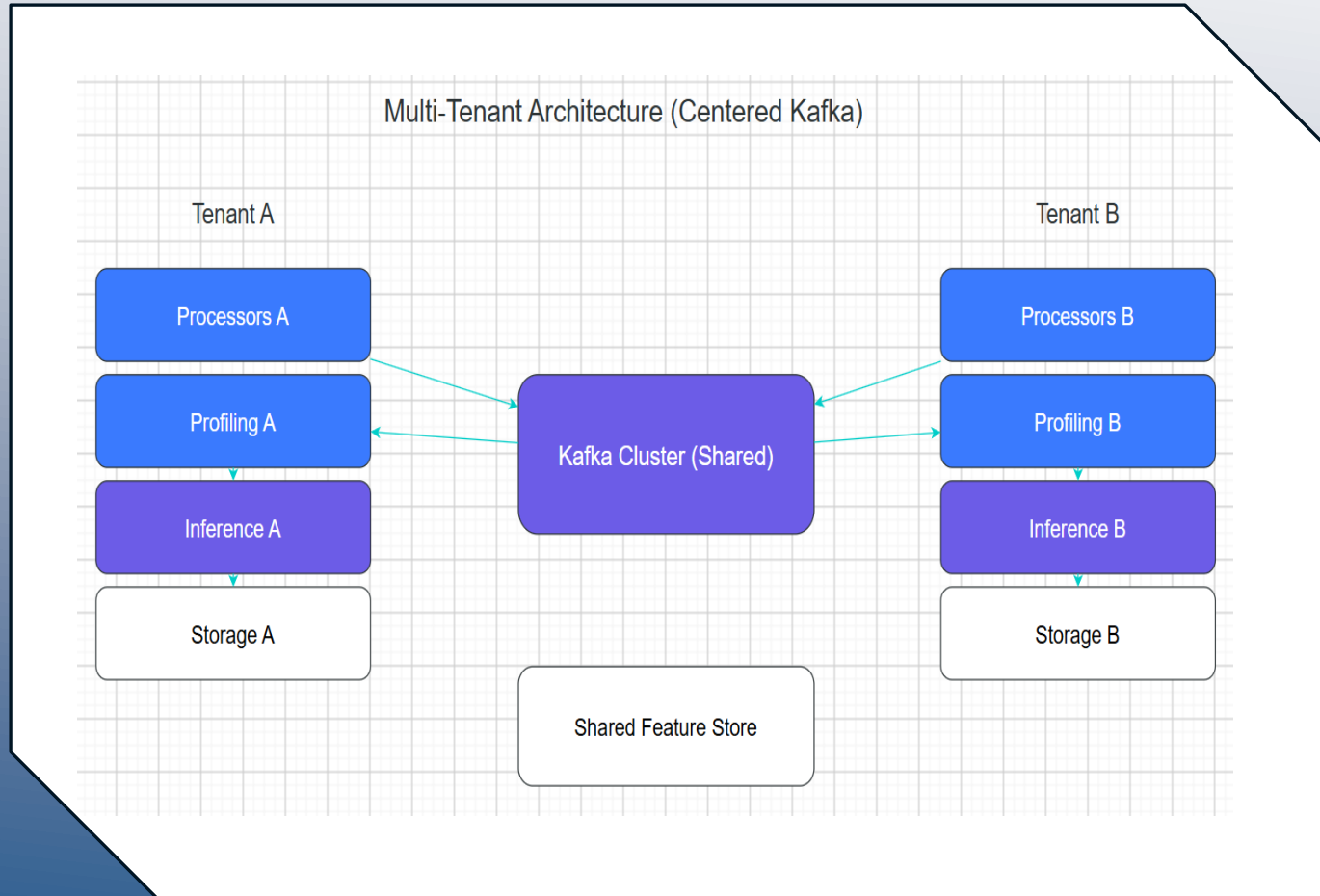
Full ML lifecycle:

Data ingestion
Feature engineering
Model training
Model validation
Model registry
Deployment
Monitoring
Drift detection
Retraining

This ensures continuous model improvement and governance.

NETCELL-EDA Architecture

Multi Tenant Architecture



Tenant isolation strategy:

Name spaced Kafka topics

Per-tenant processors

Per-tenant profiling and inference

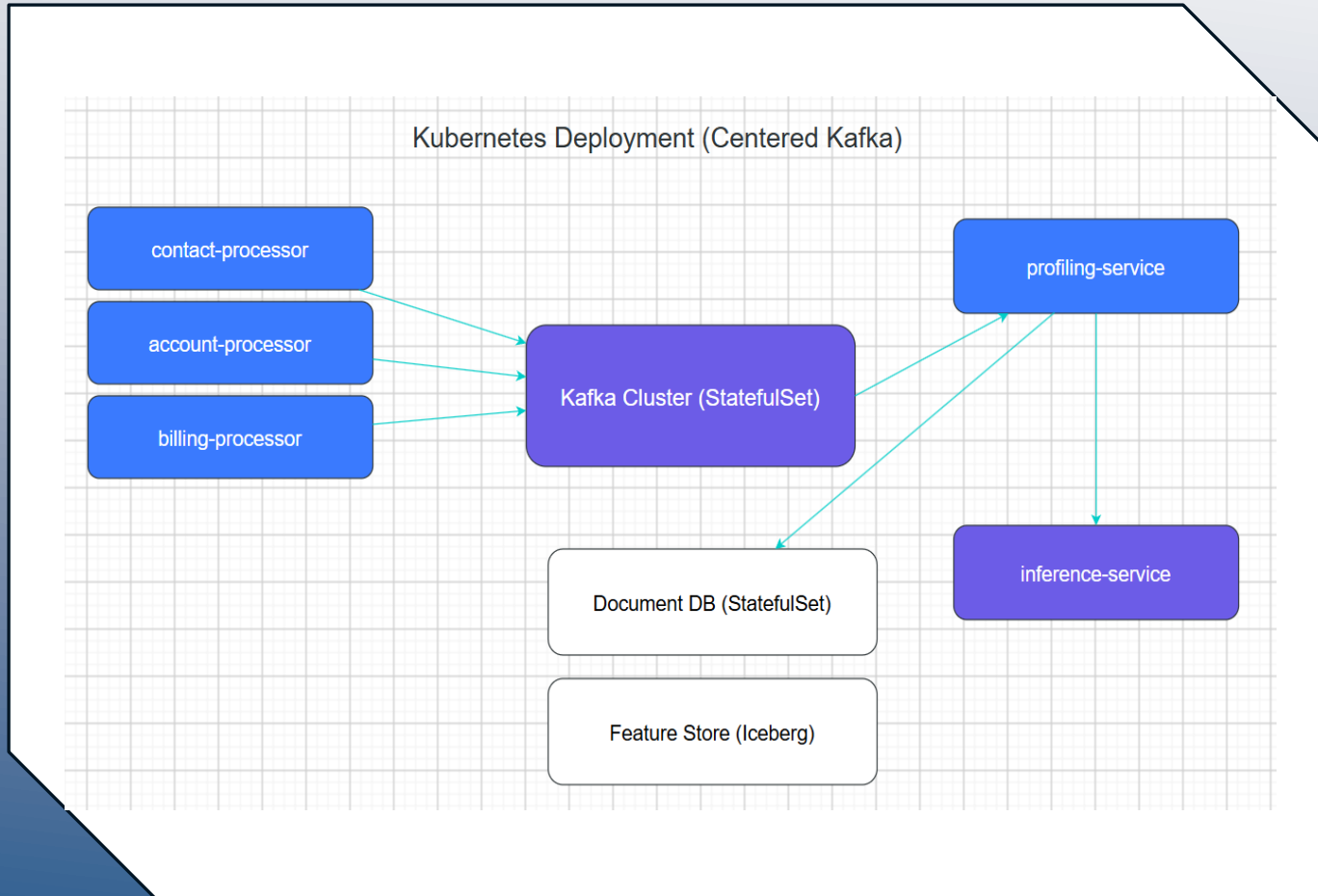
Shared infrastructure (Kafka, Feature Store)

Per-tenant storage

This enables secure and scalable multi-tenant operation.

NETCELL-EDA Architecture

Kubernetes Deployment



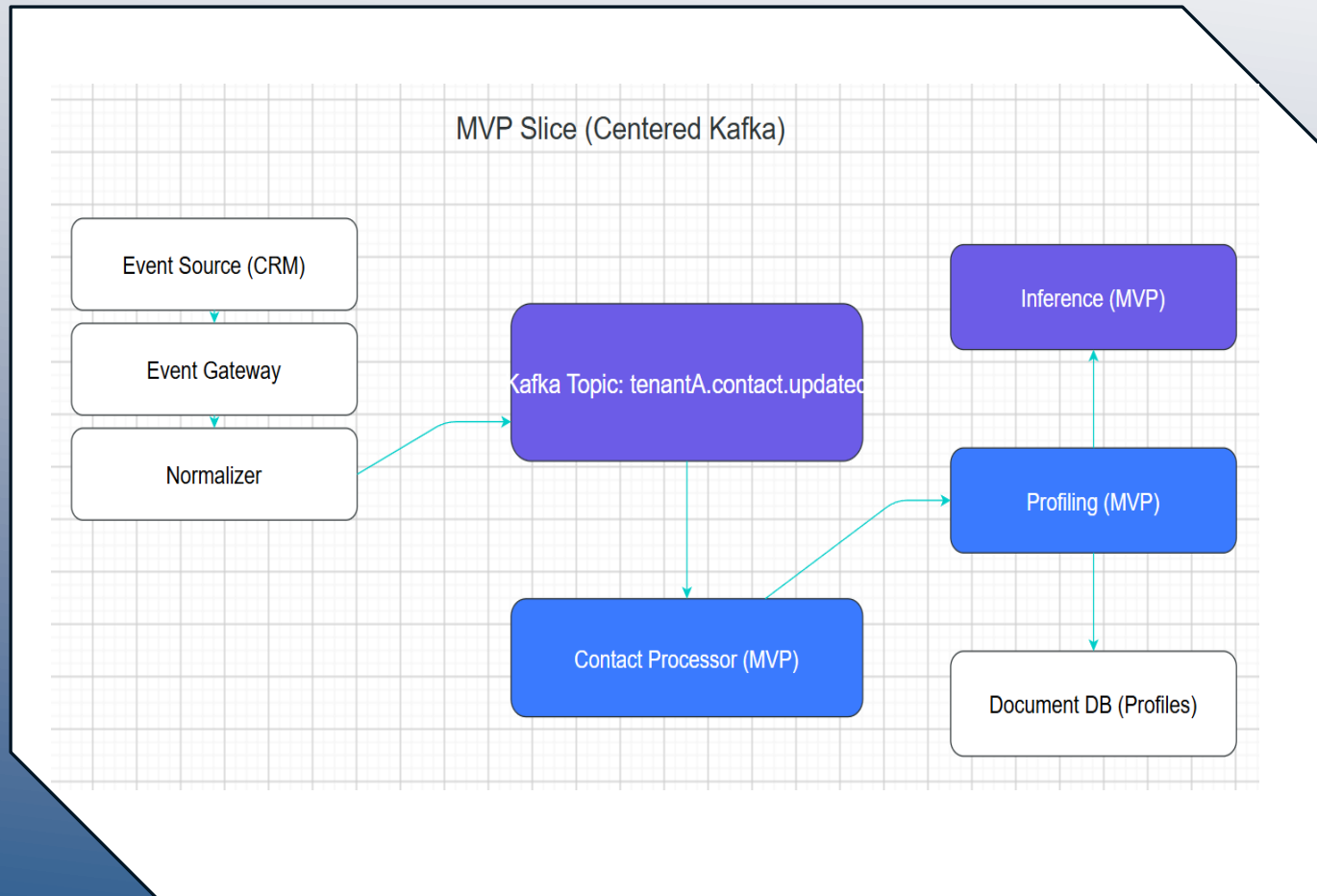
Deployment components:

Namespaces
Deployments
Stateful Sets
Services
Autoscaling
Storage classes

Kubernetes provides resilience, scalability, and operational consistency

NETCELL-EDA Architecture

MVP Slice



Minimal viable deployment:

One event type

One processor

Profiling

Inference

Document DB

Minimal Kafka topics

Minimal Feature Store

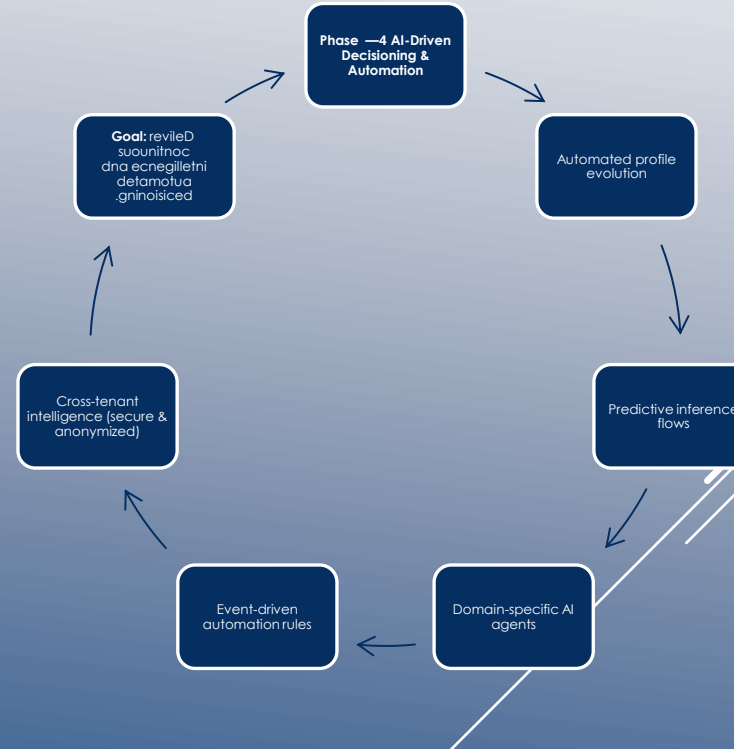
This is the fastest path to production.

NETCELL-EDA Architecture Roadmap

Core Platform —1 Phase Foundation

Event backbone (Kafka)
Microservices skeleton
Profiling engine (baseline version)
Inference engine (baseline version)
Feature store (online + offline)
Multi-tenant isolation (initial)
Kubernetes deployment
Monitoring & logging

Goal: Establish a stable, scalable, cloud-first, event-driven AI platform



Advanced Profiling & —2 Phase Intelligence

Behavioral profiling
Domain-specific profile templates
Real-time feature extraction
Model governance framework
DR strategy
Security hardening
Tenant onboarding automation

Goal: Enable real-time intelligence and safe model operations

NETCELL-EDA Architecture Roadmap

Multi-Tenant SaaS —3 Phase Expansion

Tenant management portal
Billing & usage tracking
Role-based access control
Tenant-specific pipelines
Custom inference models per tenant
High-availability deployment
Goal: Transform Netcell-EDA into a scalable SaaS platform

AI-Driven Decisioning —4 Phase & Automation

Automated profile evolution
Predictive inference flows
Domain-specific AI agents
Event-driven automation rules
Cross-tenant intelligence (secure & anonymized)
Goal: Deliver continuous intelligence and automated decisioning

Ecosystem & —5 Phase Integrations

External API gateway
Partner integrations
Marketplace for models & pipelines
Plug-in architecture for new domains
Goal: Build an ecosystem around .Netcell-EDA

NETCELL-EDA Architecture

Q&A

Thank you

The background of the slide features a blue gradient. On the right side, there are several sets of parallel white lines that are slanted upwards from left to right, creating a sense of motion or a modern design element.