Apache APISIX API Gateway Architecture – Data Plane, Control Plane, and API Types

---

1. Planes in APISIX

Data Plane (DP)

- **Location in Diagram:** Left side
- **Purpose:** Handles all actual API request and response processing.
- **Responsibilities:**
    - Receives requests from clients/services
    - Applies plugins (rate limiting, authentication, logging, etc.)
    - Proxies and routes traffic to backend services
    - Returns responses to clients

Control Plane (CP)

- **Location in Diagram:** Right side
- **Purpose:** Manages configuration, orchestration, and observability of the API gateway.
- **Responsibilities:**
    - Stores and manages configuration (routes, plugins, credentials)
    - Provides admin UI (dashboard/Manager API)
    - Synchronizes configuration to Data Plane nodes
    - Exports data to monitoring/observability tools

---

2. Data Plane Components

- **Client/Service:**
  End-users or client applications making API requests.
- **Apache APISIX Gateway:**
  The main API gateway engine (orange box), built on top of NGINX, handling incoming traffic, enforcing policies, and proxying requests.
- **Plugin Layer:**
- **Rate Limit:** Controls request rates.
- **Auth:** JWT, OAuth2, Key-auth, etc.
- **Security:** Access controls, IP restrictions.
- **Logging:** Captures API access logs.
- **Custom Plugins:** Extend APISIX functionality as needed.
- **Public / Private / Partner:**
  Categories of APIs/services exposed via APISIX, each with different access and security policies.

---

3. Control Plane Components

- **etcd Cluster:**
  Distributed, highly available storage for all APISIX configuration and metadata.
- **Dashboard / Manager API:**
  Admin interfaces to configure and manage APISIX, which send updates to etcd.
- **Observability & Monitoring Stack:**
- **Apache Skywalking, Prometheus, Grafana:**
  For tracing, monitoring, and visualizing logs/metrics collected from APISIX.

---

4. Flow Overview

API Request Flow: 1. Clients send API requests to APISIX. 2. APISIX processes the request via enabled plugins. 3. Requests are routed to the appropriate backend service (public, private, or partner APIs).

Configuration Flow: 1. Admins manage configs via Dashboard/Manager API. 2. Configurations are stored in etcd. 3. APISIX dynamically pulls configs from etcd—no restart needed.

Monitoring Flow: 1. APISIX exports monitoring data to Prometheus, Grafana, and Skywalking for real-time operational visibility.

---

5. Public, Private, and Partner APIs in APISIX

API Type	Exposed To	Example Security	Example Use
Public	Anyone/external users	Basic Auth, Rate Limit	Open data, public search
Private	Internal teams/apps	mTLS, JWT, LDAP	Microservices, internal ops
Partner	Trusted third parties	API Key, OAuth, Quotas	B2B integration, supply chain

How APISIX Handles These: - **Routing:** Defines which APIs are public, private, or partner. - **Policy Enforcement:** Applies appropriate plugins for auth, rate limiting, logging, etc., for each API category. - **Access Control:** Ensures only authorized users or partners can access non-public APIs.

---

6. Summary Table

Plane	Primary Role	Typical Components
Data Plane	Handles API requests & policies	APISIX Gateway, Plugins
Control Plane	Config & monitoring management	etcd, Dashboard/API, Monitoring tools

---

7. Summary Points

- **Data Plane:** Real API traffic processing and enforcement of policies/plugins.
- **Control Plane:** Configuration, orchestration, and monitoring of gateway nodes.
- **etcd:** Central configuration store.
- **Observability:** Real-time monitoring and visualization via Skywalking, Prometheus, and Grafana.

One-Line Summary

> **Apache APISIX** is a next-gen cloud-native API gateway, built on NGINX and etcd, supporting dynamic routing, hot plugin loading, and deep integration with monitoring stacks.

(Use as documentation, speaker notes, or slide content as needed.)