Cloud-Native Application Delivery and Operations

Shain Singh, Principal Security Architect, F5

Halim Fadhli, Service Provider Solutions Lead [ASEAN], F5

Our Speakers

Shain Singh

F5

Principal Security Architect [APCJ]

@shainsingh

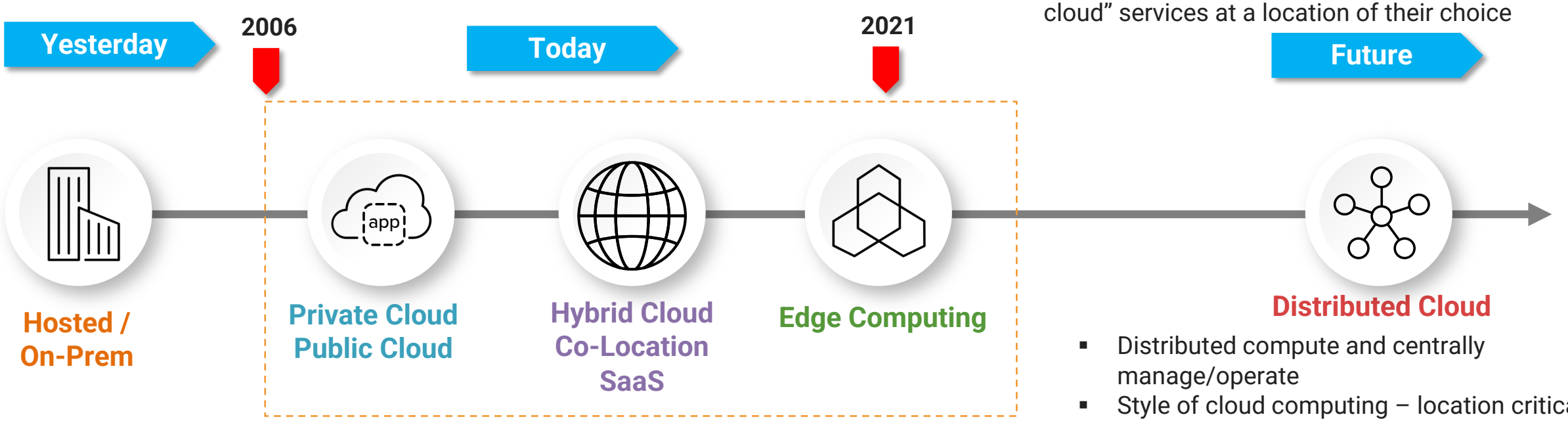
Halim Fadhli

F5

Service Provider Solution Lead [ASEAN]

Computing Evolution

Gartner
By 2025, 50% of large enterprise will enable transformation business models using “distributed cloud” services at a location of their choice



2006 – AWS Launched
2008 – Google Cloud
2009 – Alibaba Cloud
2010 – Azure Cloud
2011 – IBM SmartCloud
2012 – Oracle Cloud
…
…

Public Cloud
- Amazon
- Azure
- Google
- Ali
- Oracle

Private Cloud
- OpenStack
- VMWare
- IBM
- HPE

Co-Lo
- Equinix
- NextDC
- RackSpace

- Distributed compute and centrally manage/operate
- Style of cloud computing – location critical component
- Low latency closer to who need capabilities
- Data Sovereignty
- Data egress cost
- Reduce risk global outage

Catalyst
- Multi-Cloud
- IOT Platform
- Edge AI
- SASE
- Service Mesh
- Etd.

Edge Evolution

Content-Centric Solution

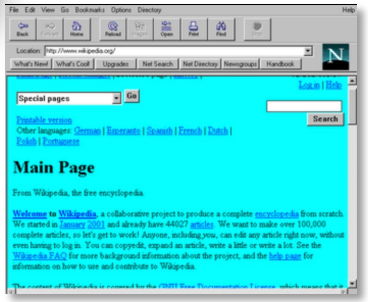
App-Centric Solution

Edge 1.0

Edge 1.5

Edge 2.0

Hosted / On-Prem



1998

- Content Delivery Network (CDN)
- Focus on static content
- Solve slow internet link and traffic congestion by content closer to user
- Netscape Navigator. End device “dumb”. Passive participant
- Physical PoP

Edge Computing

- The rise of applications – digital economy
- The rise of cyber threat and computing power.
- Security add-on staple to CDN provider (mitigate closer to the source)
- Proprietary env. Service non portable to another.
- Challenges
  - Endpoint passive entities
  - Rise of container-based apps and intelligent end-user computing

Distributed Cloud/Edge

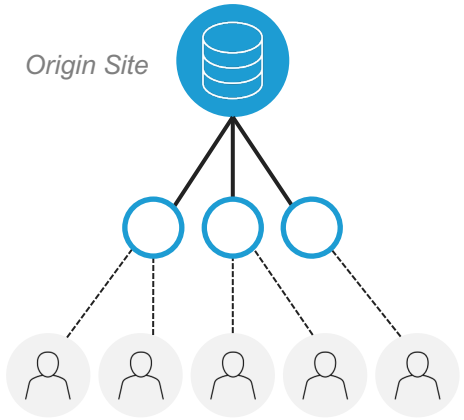
F5 Research

76% of enterprises planning to use Edge for various use cases

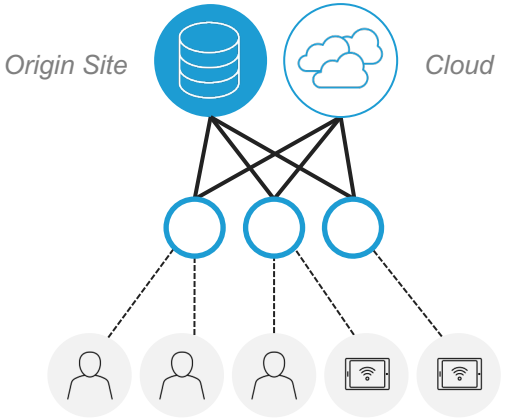
- Improve performance
- Speeding data collection and analytics
- Supporting IoT
- Real-Time or Near-Real-Time processing

Application delivery is also changing

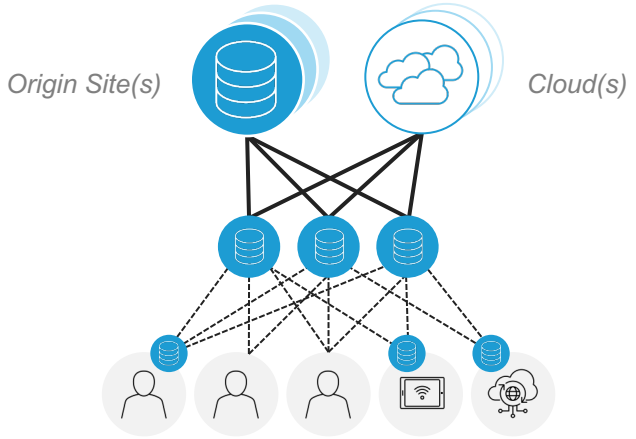
CDNs
Scale out static object serving

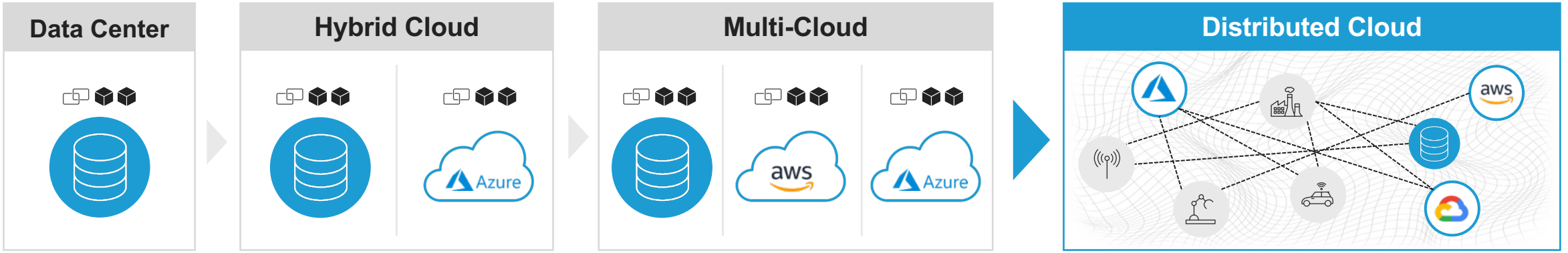


Cloud
Scale out app servers



Distributed Cloud
Scale and connect everything





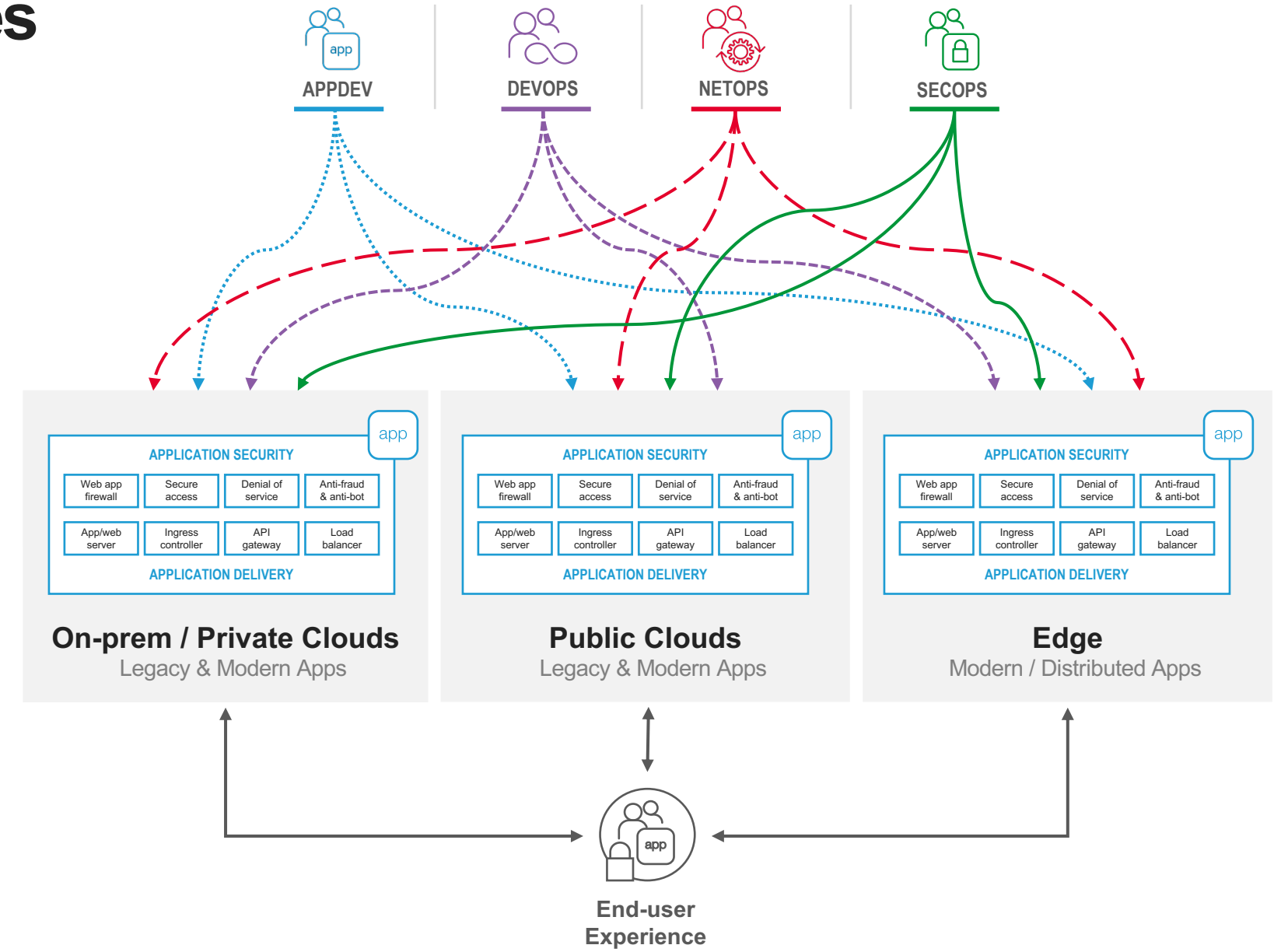
Technical challenges of delivering apps

#1 Complex coordination because of technology inconsistencies between teams and across environments

#2 Automation challenge ”stitching” multiple environments, layering net, security, and apps, at scale

#3 Security difficulties due to multiple different attack surfaces and sophistication of bad actors

#4 Limited observability of siloed telemetry trapped in disjointed systems & environments

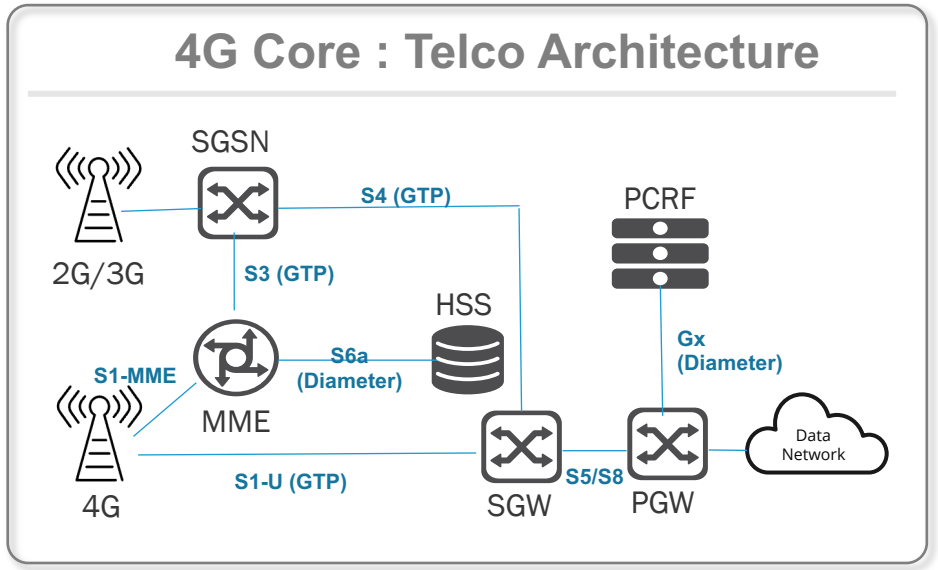


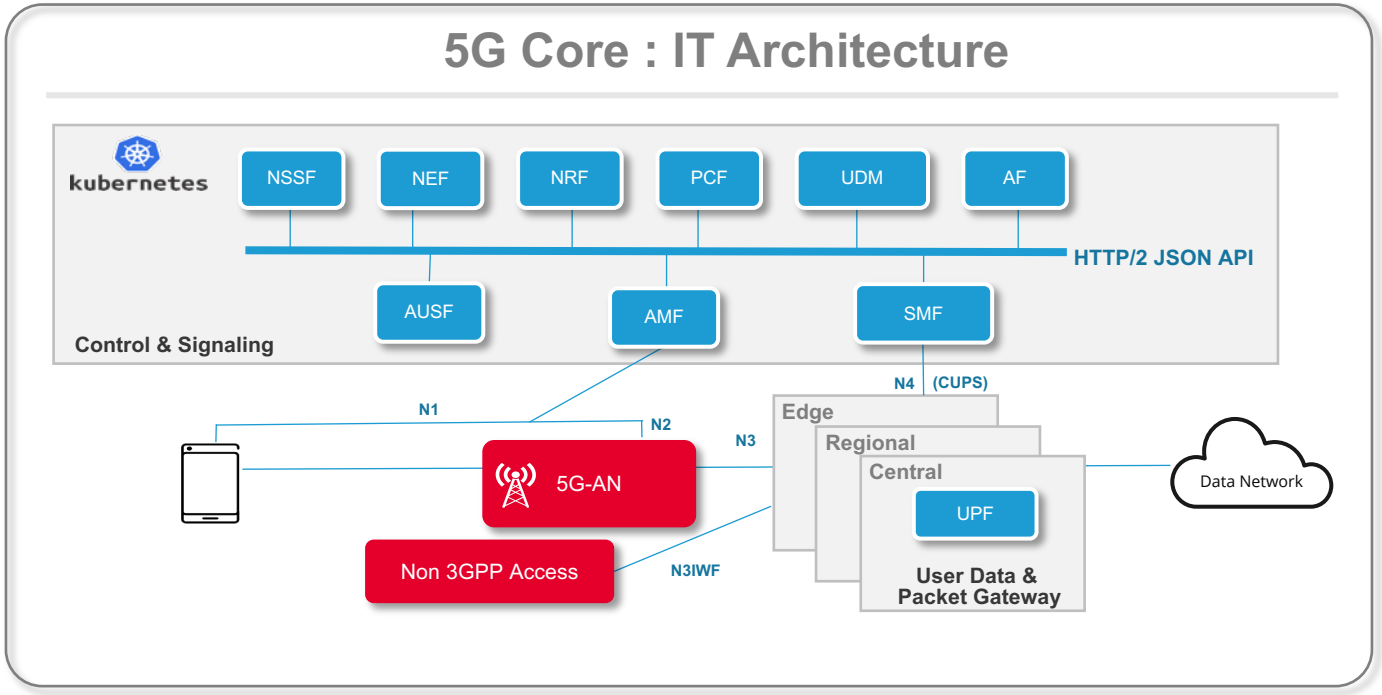
Industry Case Study

Carrier Service Providers

From 4G to 5G : Functional & Architectural Transformation

SERVICE BASED ARCHITECTURE (SBA)

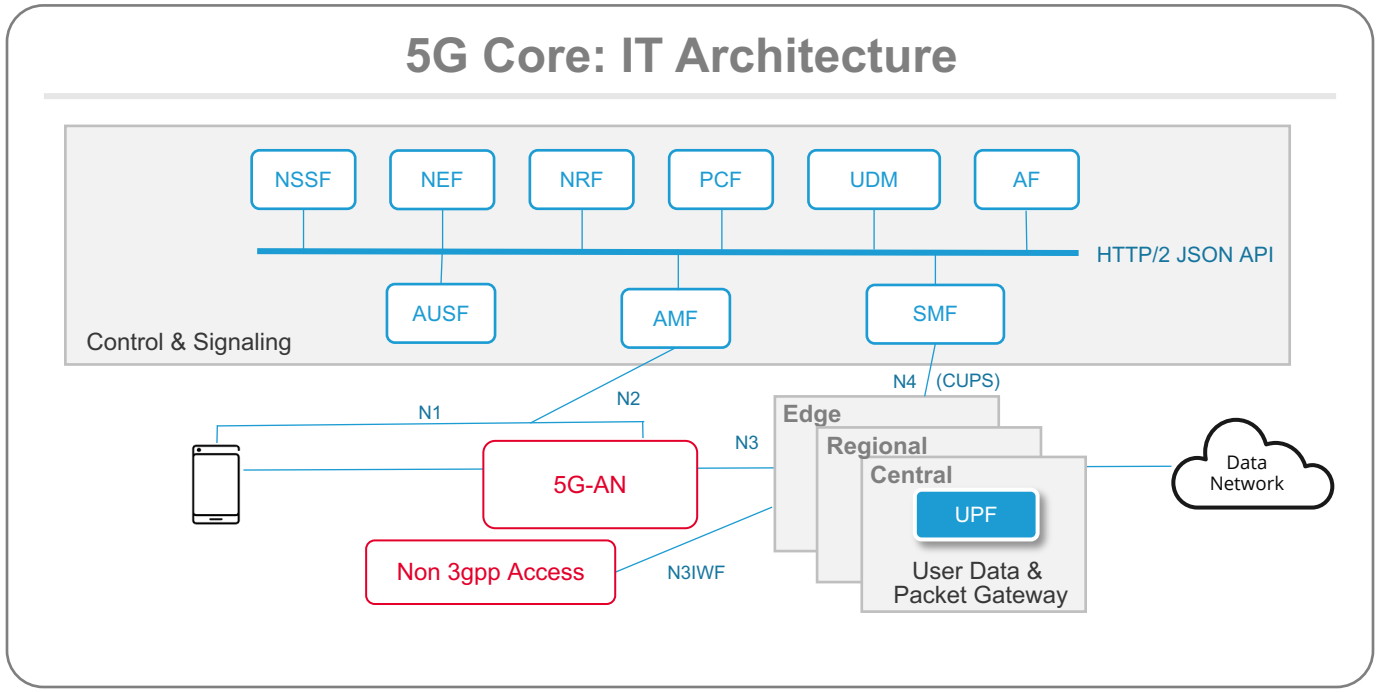


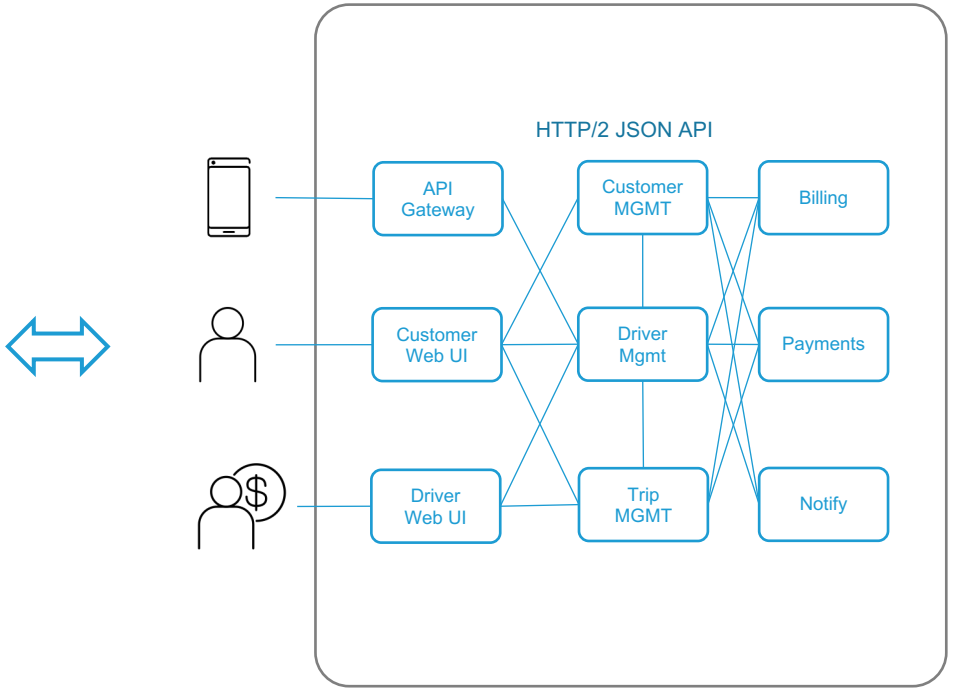


5G SBA Technology Principles (derived from IT industry)

- Micro-Services
- API centric
- HTTP/2 — Web protocol
- Telco Cloud
- CUPS

5G SBA is an example of a Modern Application Architecture



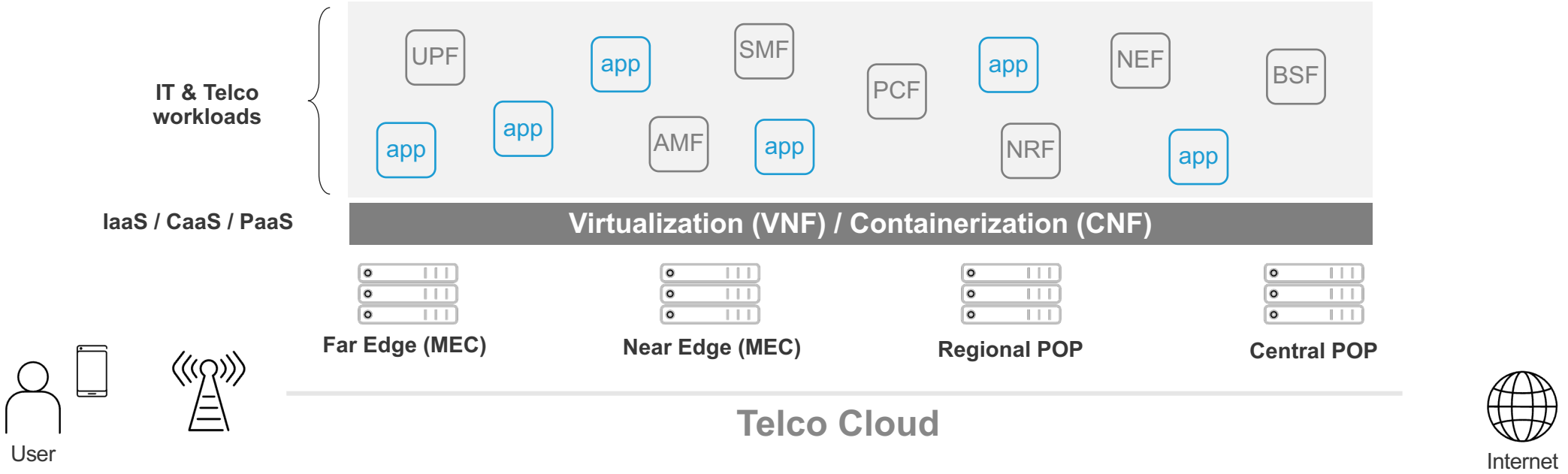


What are the equivalent parts?

- Micro-Services
- Cloud native
- HTTP/2 Web protocol
- API centric

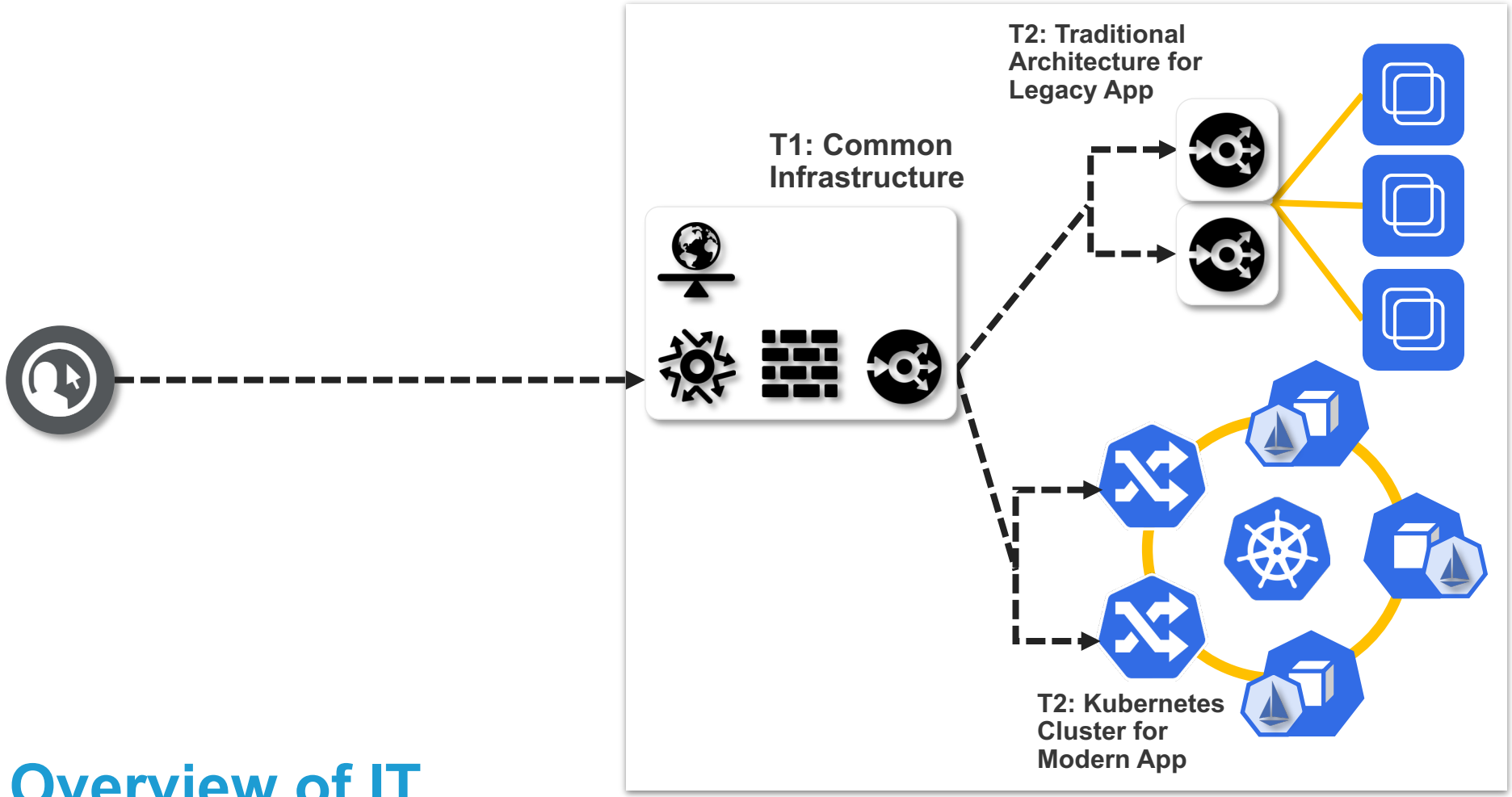
Service Providers on the Road to become IT Companies

5G & EDGE COMPUTING : MANAGING TELCO AND IT WORKLOADS ON A TELCO CLOUD PLATFORM



Horizontal Telco Cloud for Network Functions and Applications

Overview of IT Architecture Evolution



Traditional App Approach

- Processes are typically server-side heavy
- Single, monolithic and stateful DB
- Single App – presentation and business logic in single module
- 3-Tier – Web presentation and Business logics in layered modules

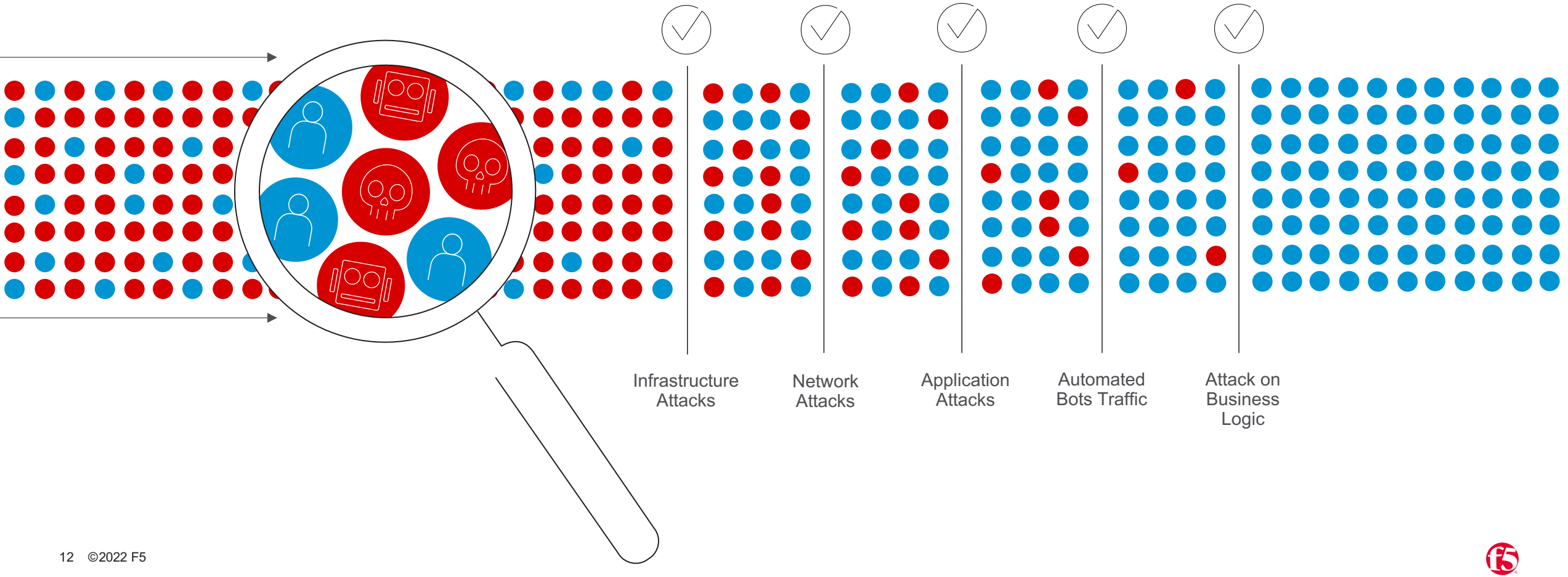
Modern App Approach

- Presentation layer processing are mostly offloaded to client/browser side.
- Server side are largely dealing with business logics and data aggregations
- Business logic are broken down into individual services or modules.
- Services components communicate with each other over APIs.
- Services can be stateful or stateless with DB
- Fully automated

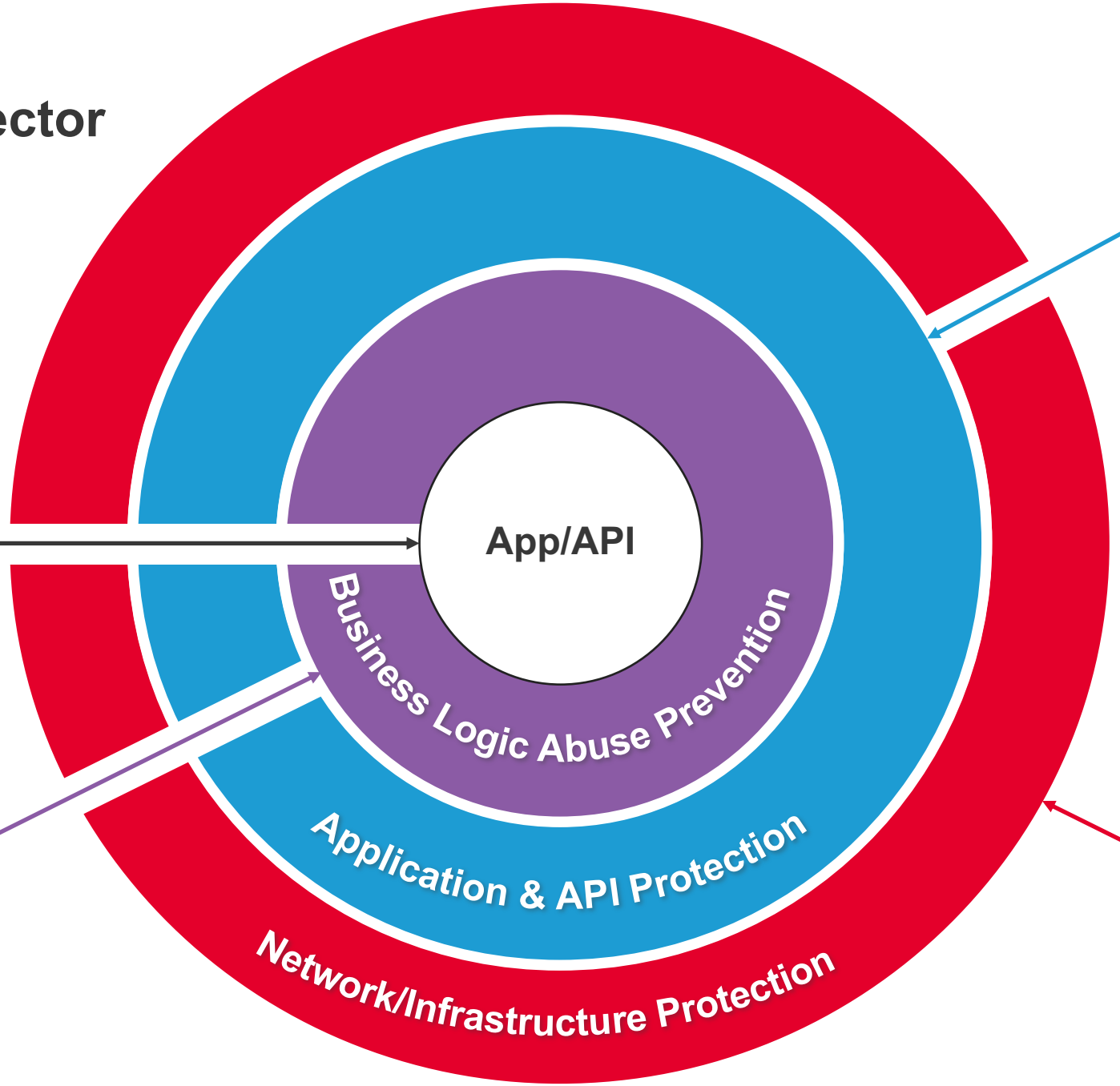
Cloud Microservices PaaS

- On-prem private cloud
- Public cloud (e.g: AWS, Azure, GCP)

Modern attacks can affect Business Intelligence



Modern Apps Require Multi-Vector Protection



Back door channel
Exploiting vulnerabilities
Overwhelming application

Web Bot

Script Kiddies

State Sponsored Attacks

User

Hacker

Fraudster

Front door channel
Exploiting business logics
Stealing financial asset

Malware

Known Bad IP

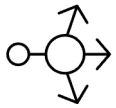
Infected Bots

Attacker

Back door channel
Exploiting network loophole
Overwhelming network

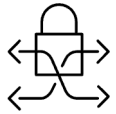
F5 Security Solutions for Modern Telco Architecture

Securing Cloud Native Infrastructure

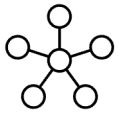


Ingress Control
- KIC with Application Security (for IT/Edge workload scenarios)
- SPK (KIC for service provider protocols into 5GC)

API Gateway
- API Protection of services (such as IT App or 5G NEF)

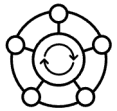


Service Mesh
- NSM (for IT App Service Mesh)
- CGAM (for internetworking and security between CNFs)



Infrastructure
- Securing Kubernetes/cloud native infrastructure as well as providing networking services

Securing 5G Core Network



N6 Services
- Gi/N6-LAN Services as CNF
  - CGNAT
  - N6-FW
  - DDoS Protection
  - Application Detection
  - DNS
  - etc.

Service Bus Interface (5G SBI)
- Security Edge Protection Proxy (SEPP) [roadmap]
- Service Communication Proxy (SCP)

Roaming Interconnect
- N9 (GTP-u) interface firewalling and protection
- Security Edge Protection Proxy (SEPP) [roadmap]

Securing Distributed Cloud/Edge

Hybrid and Multi Cloud Networking
- Connecting multi-cloud environment
- Connecting MEC edge sites

Application Delivery Network
- Run microservice-based apps wherever required globally, in the cloud, data center, or the edge

Hybrid and Multi Cloud Security
- Distributed security in hybrid or multi cloud world

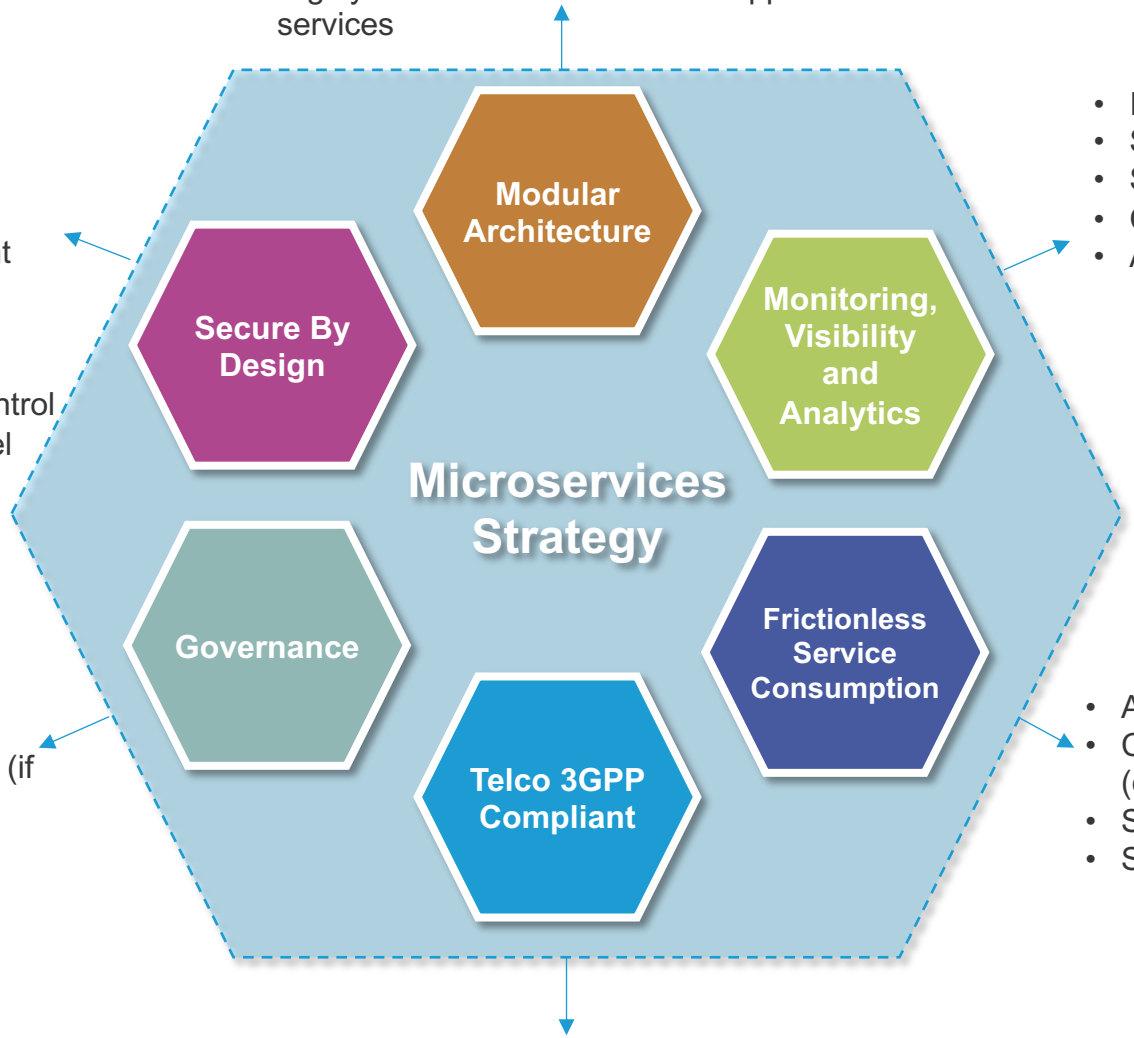
End-to-End Security and Operations

Comprehensive strategy for modern architectures

Key Components of a Microservices Strategy

- Portable and platform vendor independent
- Agile and scalable architecture – Works everywhere, infrastructure independent
- Environment elasticity
- API driven ecosystem
- Run in consistent and predictable manner
- Highly resilient architecture and application services

- Embed Security control in the beginning
- Secure image registries
  - Trusted signed images
  - Container Vulnerability Scan
  - Scan for unintentional private embedded data.
- Application Services Protection
  - Web Application Firewall
  - Bot Protection
  - L7 DoS Protection
- Identity and access management
  - mTLS
  - Authentication Proxy
  - OpenID Connect
- RBAC – Role-based Access Control
  - Least privilege access model



- Insight on apps utilization & performance
- SLA and resiliency and availability monitoring
- Service oriented monitoring
- Cost consumption monitoring
- Application performance monitoring (APM)

- Standardization
- Tooling and language framework
- Deployment pipeline
- Alignment with standards bodies (if possible – e.g. OCI, CNCF)
- Automation
- Consistent deployment patterns
- API definition and strategy
- Monitoring (e.g. metrics and dashboard)

- API driven ecosystem
- Consumable full application services (e.g. traffic management and security)
- Support Blue/Green deployment model
- Seamless integration into CI/CD pipeline

- Organization structure alignment

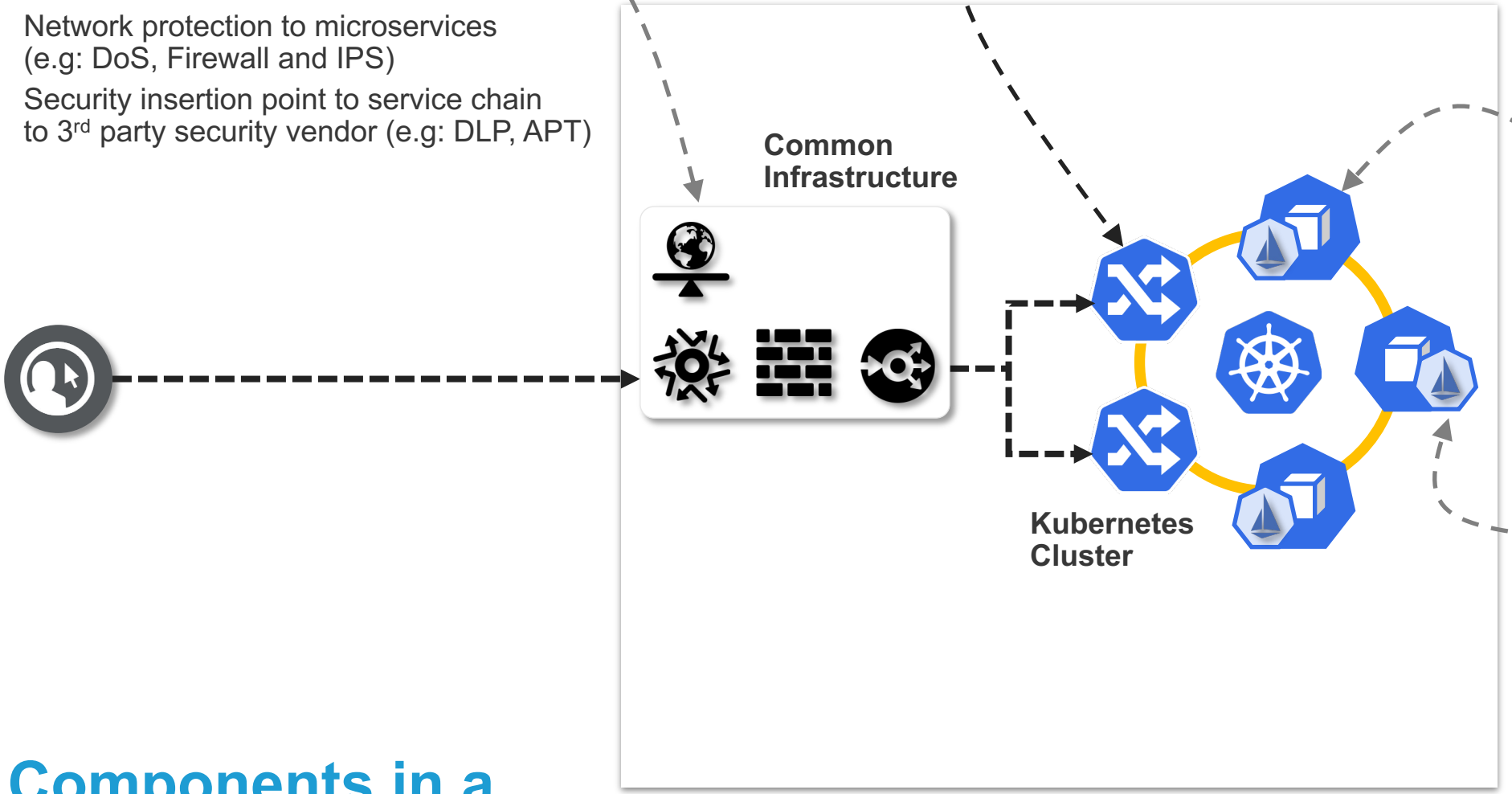
Components in a Modern Architecture

Infra Services

- L4 LB to Kubernetes nodes
- Application resiliency across multi-cloud, multi-cluster (e.g: DNS and GSLB)
- Network protection to microservices (e.g: DoS, Firewall and IPS)
- Security insertion point to service chain to 3rd party security vendor (e.g: DLP, APT)

Ingress (with API Gateway)

- Layer 7 routing for traffic entry point coming into Kubernetes
- Built for HTTP traffic. TCP/UDP for non-HTTP traffic
- May include API Gateway implementation



Pods

- Runs app container / CNF

Service Mesh

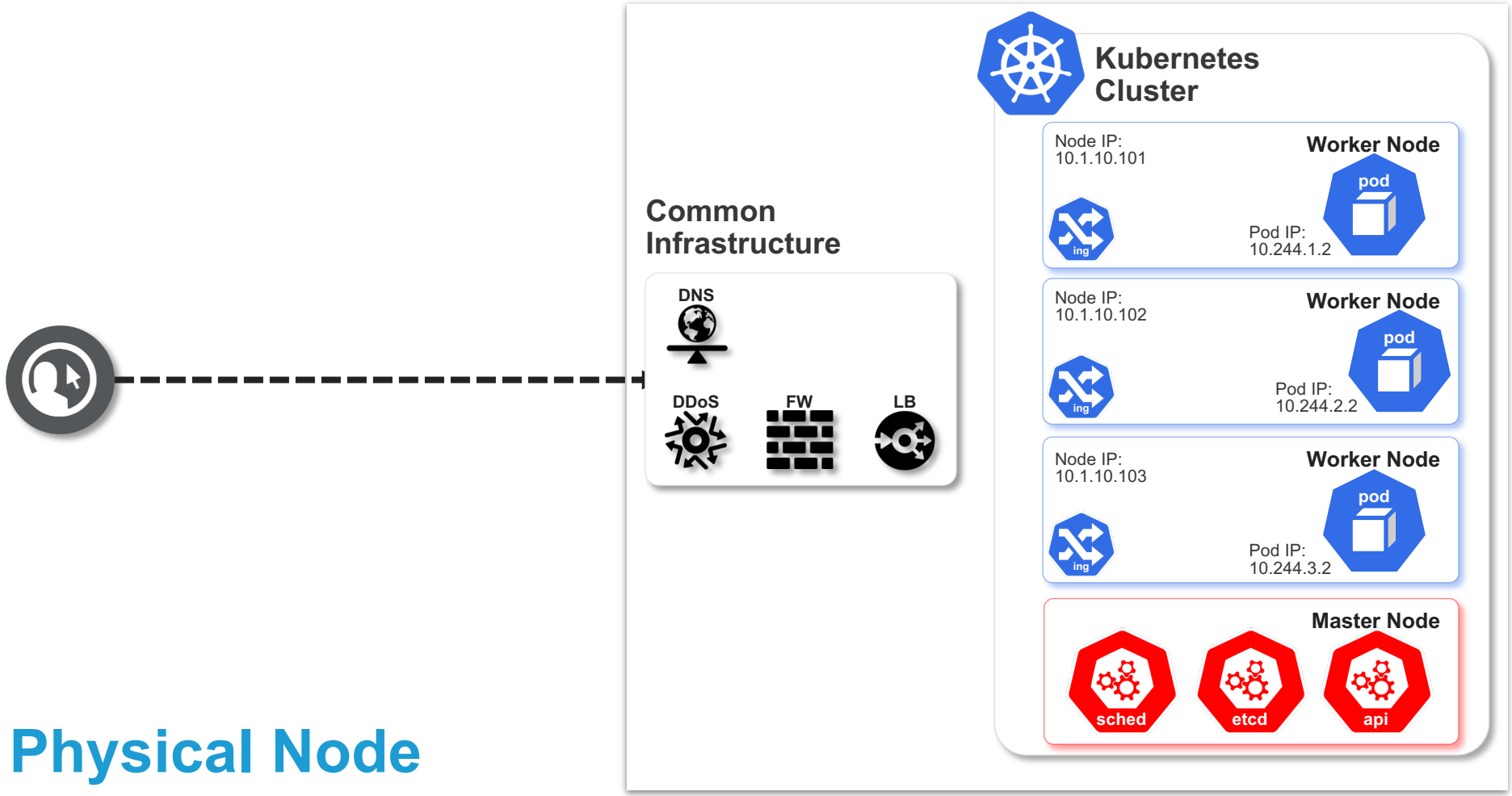
- Open Source Service Mesh implementation (Istio)
- Injects Sidecar to every pod
- Enforces routing, security with mTLS, etc.
- Provides traceability of pod communication

Cloud Microservices PaaS

- On-prem private cloud
- Public cloud (e.g: AWS, Azure, GCP)

Physical Node Diagram

K8s is designed to be highly customizable



Cloud Microservices PaaS

- On-prem private cloud
- Public cloud (e.g: AWS, Azure, GCP)

Underneath it all, there is more component that stitch K8s together such as…

Container Runtime Interface (CRI)

- Container runtime that allows K8s to run containers in pod

Container Network Interface (CNI)

- Provides networking within K8s cluster so containers can communicate to each other as well as isolating as per policy applied

Ingress Controllers

- Manage external access to the services in a cluster and provides L7 routing, load balancing, SSL termination and name-based virtual hosting

External Load Balancer

- Externally-accessible IP address that sends traffic to the correct port on K8s cluster nodes

External DNS

- makes K8s resources discoverable via public DNS servers

Image Registry Management

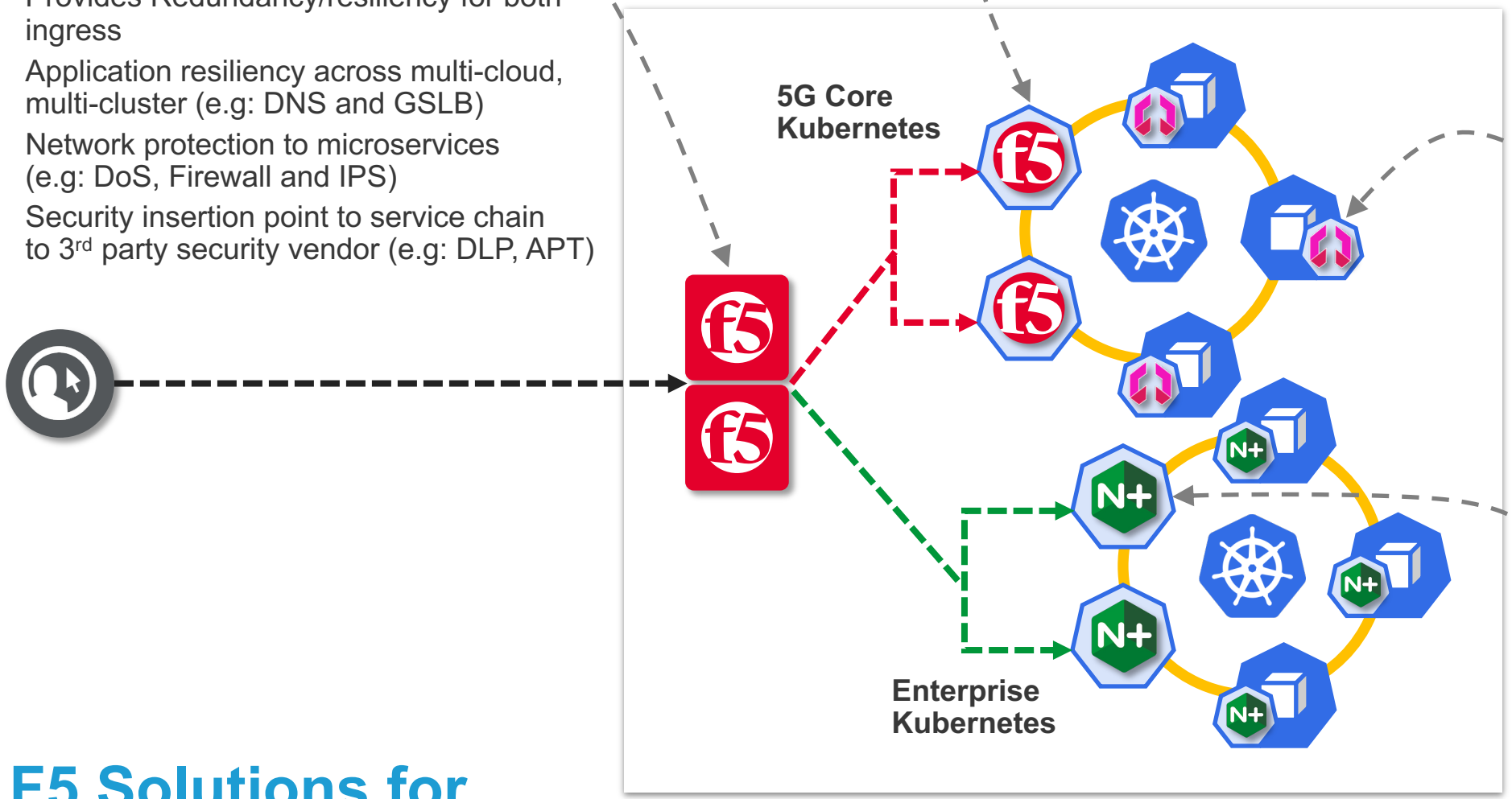
- Internal, integrated container image registry to build images from source code, deploy, and manage its lifecycle

F5 Service Gateway

- L4 LB to F5 SPK for SP Protocol
- L4 LB to NGINX+ KIC
- Provides Redundancy/resiliency for both ingress
- Application resiliency across multi-cloud, multi-cluster (e.g: DNS and GSLB)
- Network protection to microservices (e.g: DoS, Firewall and IPS)
- Security insertion point to service chain to 3rd party security vendor (e.g: DLP, APT)

F5 SPK

- Carrier-grade Kubernetes Ingress Controller
- Multi-protocol aware (e.g: Diameter, SIP, MQTT, etc.)



Aspen Mesh

- Carrier-grade distribution of ISTIO service mesh
- Enhanced RBAC
- Visibility, Security and Observability
- 3GPP compliant sidecar

NGINX+ & Service Mesh

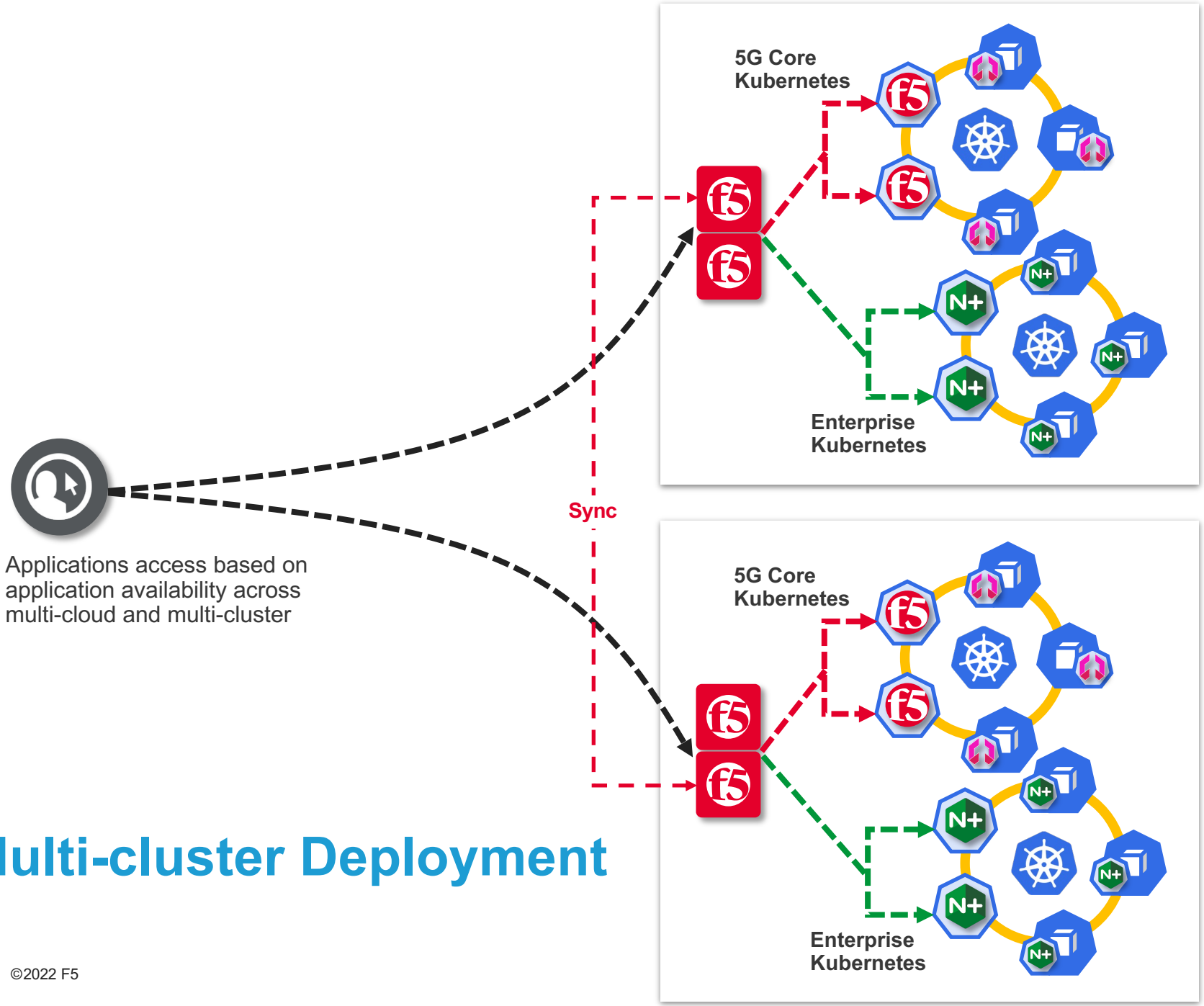
- Kubernetes Ingress Controller
- Enterprise-grade WAF
- L7 DoS Protection
- Lightweight service discovery
- Cloud-agnostic Lightweight App Services NGINX+ App Protect (WAF)
- Egress traffic control

Cloud Microservices PaaS

- On-prem private cloud
- Public cloud (e.g: AWS, Azure, GCP)

F5 Solutions for Modern Architectures

Multi-cluster Deployment



Cloud Cluster 1

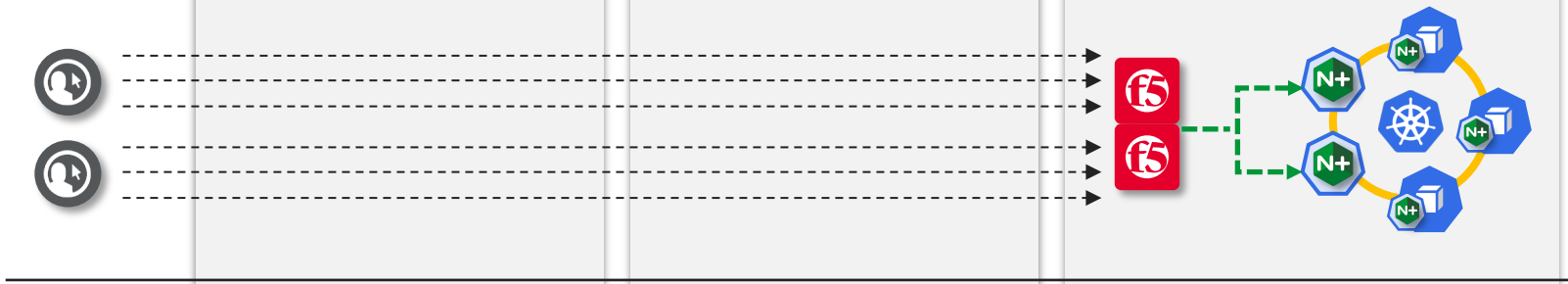
Cloud Microservices PaaS

- On-prem private cloud
- Public cloud (e.g: AWS, Azure, GCP)

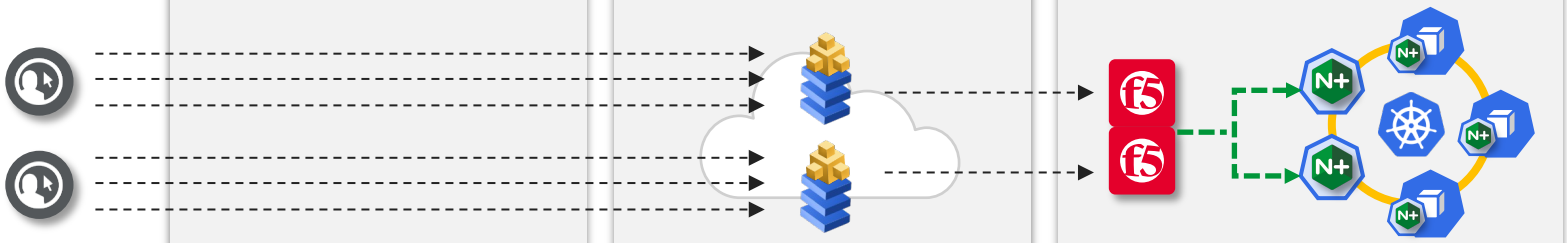
Cloud Cluster 2

Cloud Microservices PaaS

- On-prem private cloud
- Public cloud (e.g: AWS, Azure, GCP)



- Highly secured K8s cluster in central sites
- Latency from end user to central sites



- Highly secured K8s cluster in central sites
- Secure K8s infrastructure in cloud edge
- Latency from end user to cloud edge
- Central sites allow traffic originating from cloud edge IP only **OR** secure tunnel from cloud edge to central sites for end to end private connection

- Highly secured K8s cluster in central sites
- Secure, lightweight K8s platform on far-edge / local network
- Latency from end user to far-edge / local network
- Secure tunnel from far-edge to central sites for end to end private connection

Far Edge Sites (1000s)
- Branch offices, factories, shops, home, etc.
- Lightweight, easy to deploy and managed as fleet K8s
- Lightweight, real time processing application workload

Edge Sites (100s)
- Cloud provider edge or point of presence (PoPs)
- Managed K8s, ready to run workload in the cloud on pre-built PoPs
- Lightweight, real time processing application workload

Central Sites (10s)
- AWS, Azure, GCP, On-prem DC, etc.
- Highly governed, deep policy K8s cluster
- Critical/core application workload, database

Distributed Modern Architecture

Single Datacenter Deployment

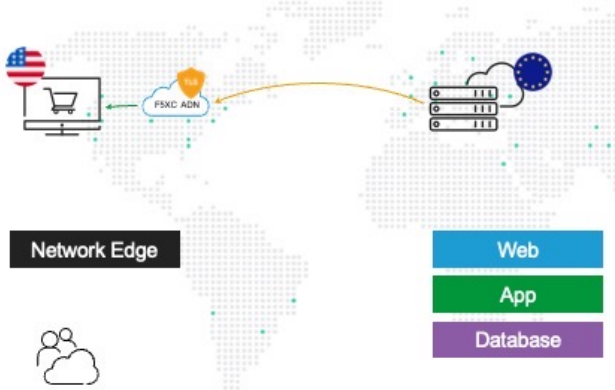
ALL 3 TIERS IN CUSTOMER'S EU DATACENTER

Baseline Scenario

- Single datacenter hosting all 3 tiers of the application
- TLS & cookie/API processing is handled by the app tier in the DC
- Sub-optimal end user experience for most geo's outside of the EU

Scenario 1: SSL Termination on ADN

MOVE TLS SETUP CLOSER TO THE END-USER

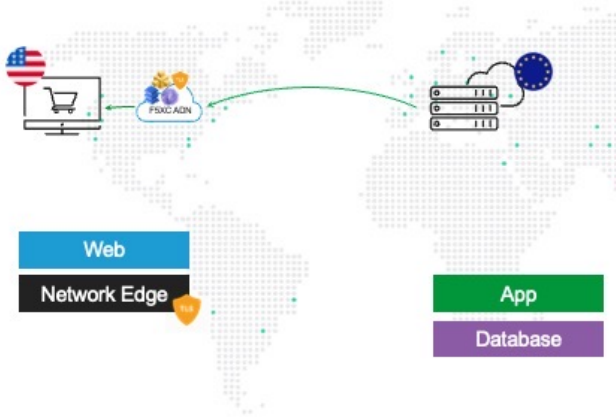


Baseline Scenario

- Leverage ADN load balancer to handle HTTPS
- High-performance network connectivity to App Tier
- Offload TLS/HTTPS processing to high performance ADN

Scenario 2: Front-End on ADN

MOVE LATENCY SENSITIVE SERVICES TO CLOSER TO USER

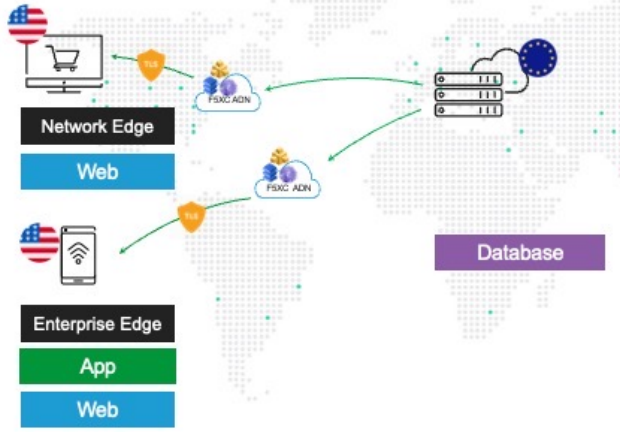


Move container to ADN

- Easily move container-based apps to full-featured containers on vK8s
- Native Kubernetes environment w/ familiar tooling for DevOps
- Ease of deployment, management of workloads across multiple K8s & clouds

Scenario 3: Front-End to In-Store

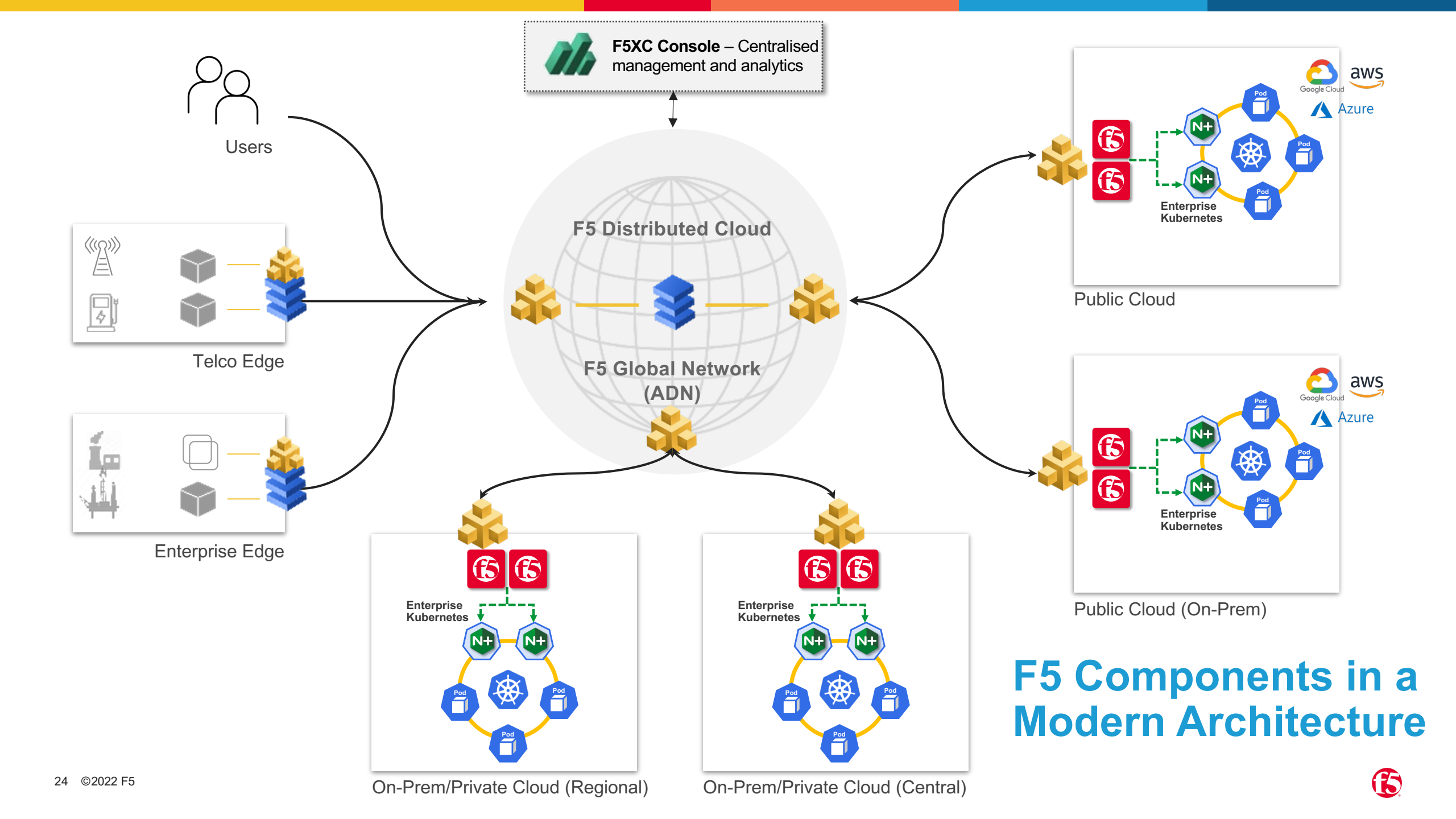
MOVE LATENCY SENSITIVE SERVICES CLOSER TO USER



Move container to in-store

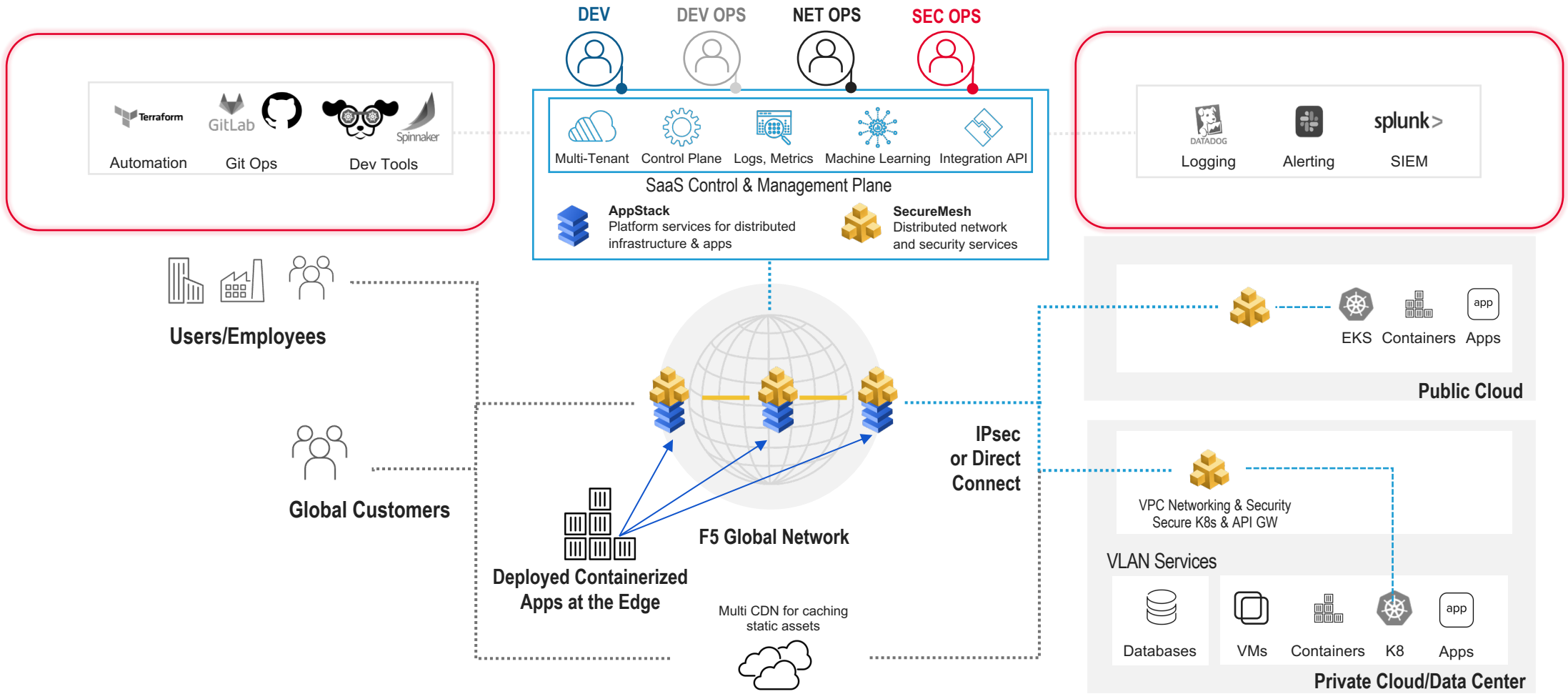
- Bring the Front-End and Latency-sensitive services to In-store
- Leverage K8s on VMware or Bare Metal
- Optimize performance where it's needed; secure connection back to database, etc.
- Single pane of glass manageability

Summary



F5 Components in a Modern Architecture

Modern Application Delivery and Operations



Thanks for listening!