Connection Platform (CoPa) at Generali - Microservices and Kafka on OpenShift

Insights into an Event-Driven Data Streaming Project

23 September 2019

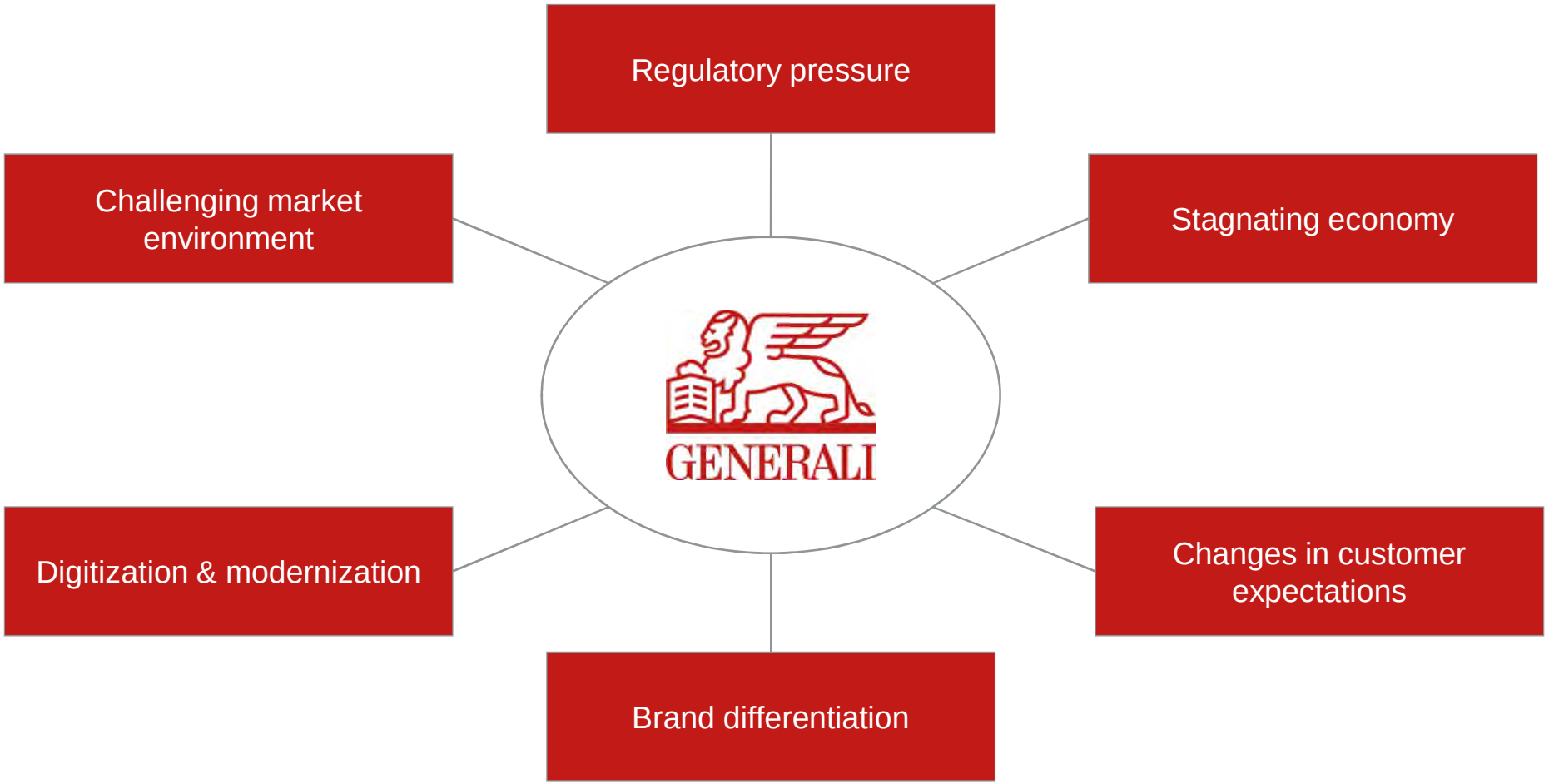
Thomas Peter (Generali)
Abhishek Pande (Generali)
Yves Brise (ipt)

Agenda

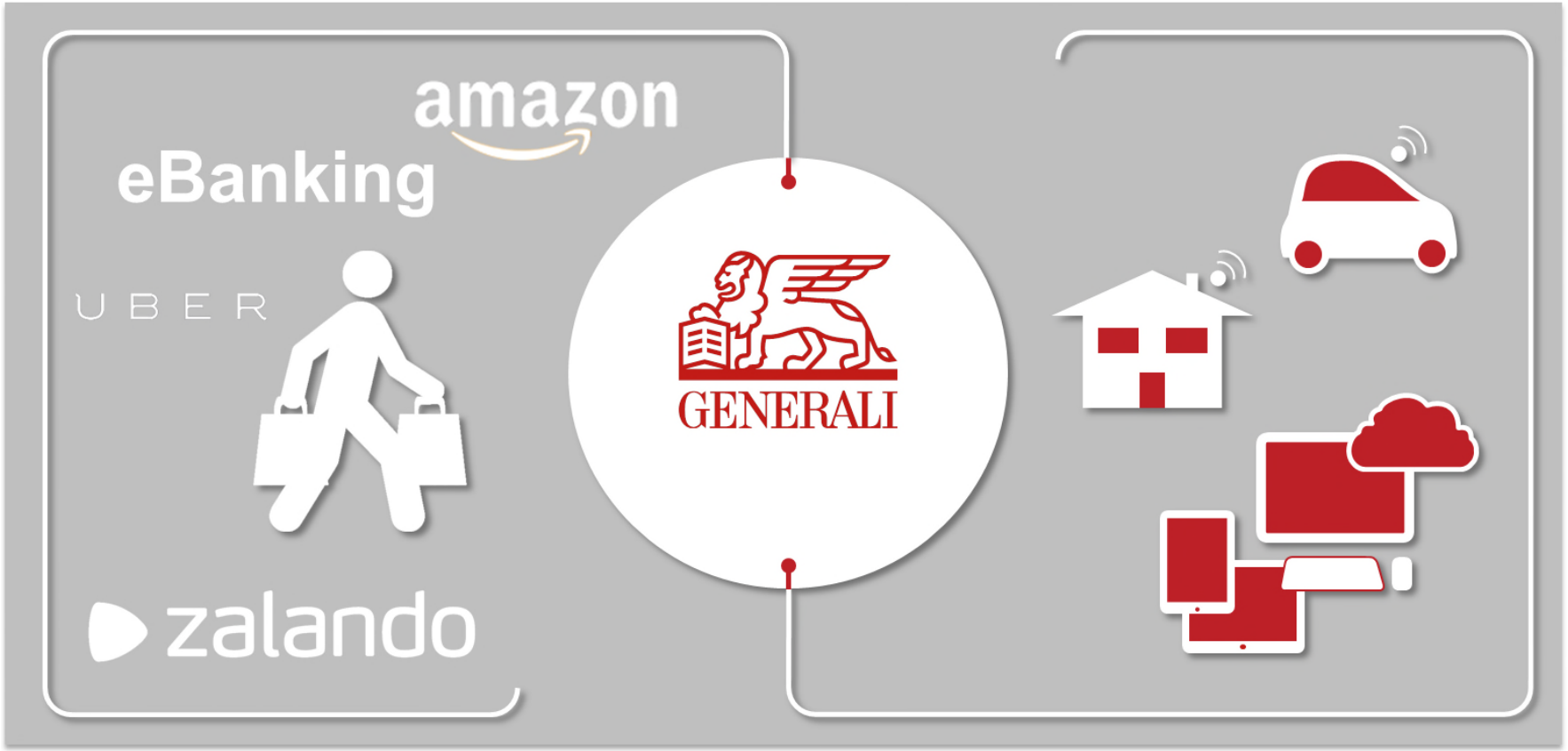
1. Vision & Principles
2. Technical Architecture
3. Design
4. Challenges
5. Next steps
6. Key Take-Aways

Vision & Principles

Starting Position



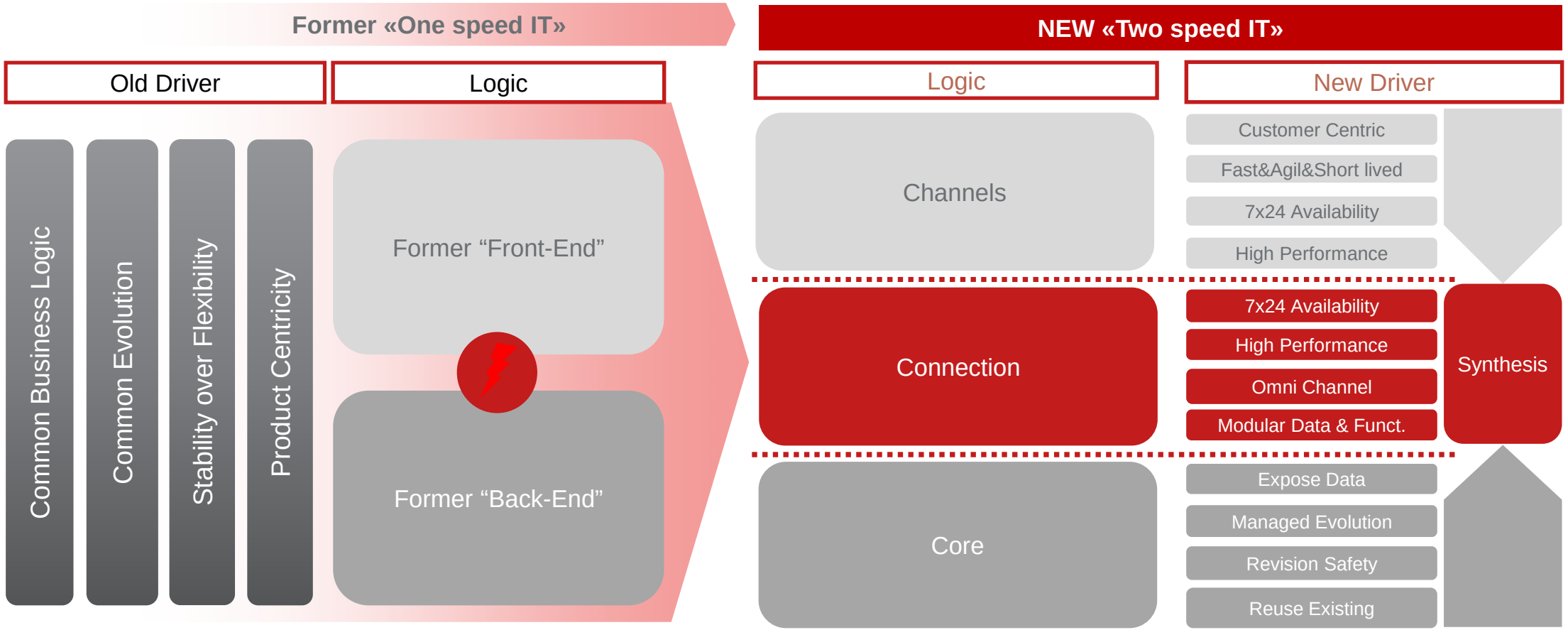
Changes in Customer Expectations



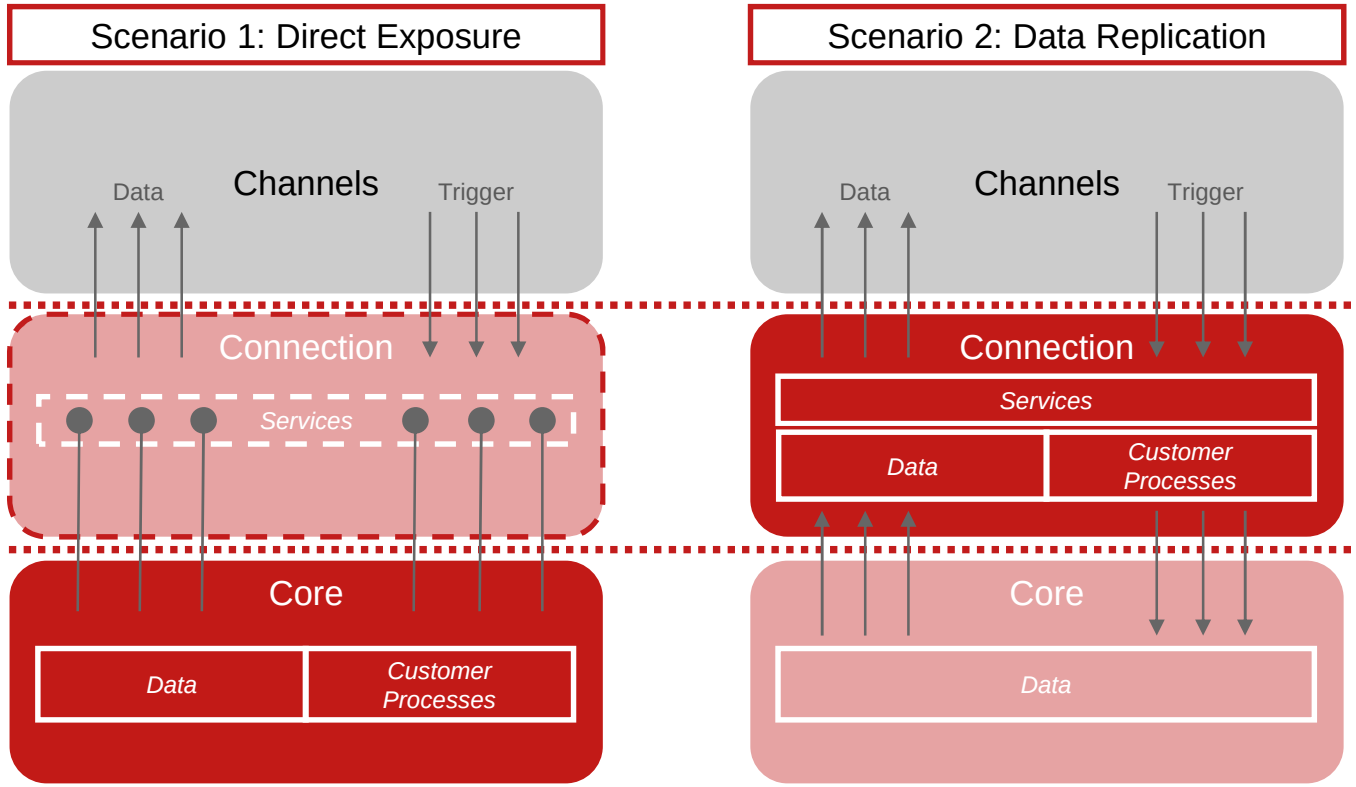
Sleeping Beauty

https://vignette.wikia.nocookie.net/grimmbilder/images/b/b5/Dornroeschen_Paul_Hey_Jahr_unbekannt.jpg/revision/latest?cb=20130516132113&path-prefix=de

A Two Speed IT with a Three Tier Approach



A Platform for Data, Services, and Processes



Key Design Principles of the Platform

Agility & Scalability

- Strive for **Fast Delivery (TTM)**
- **Capability Model** Drives **Governance**
- Foster **Enterprise Assets**
- **Flexible** Structuring and Deployment

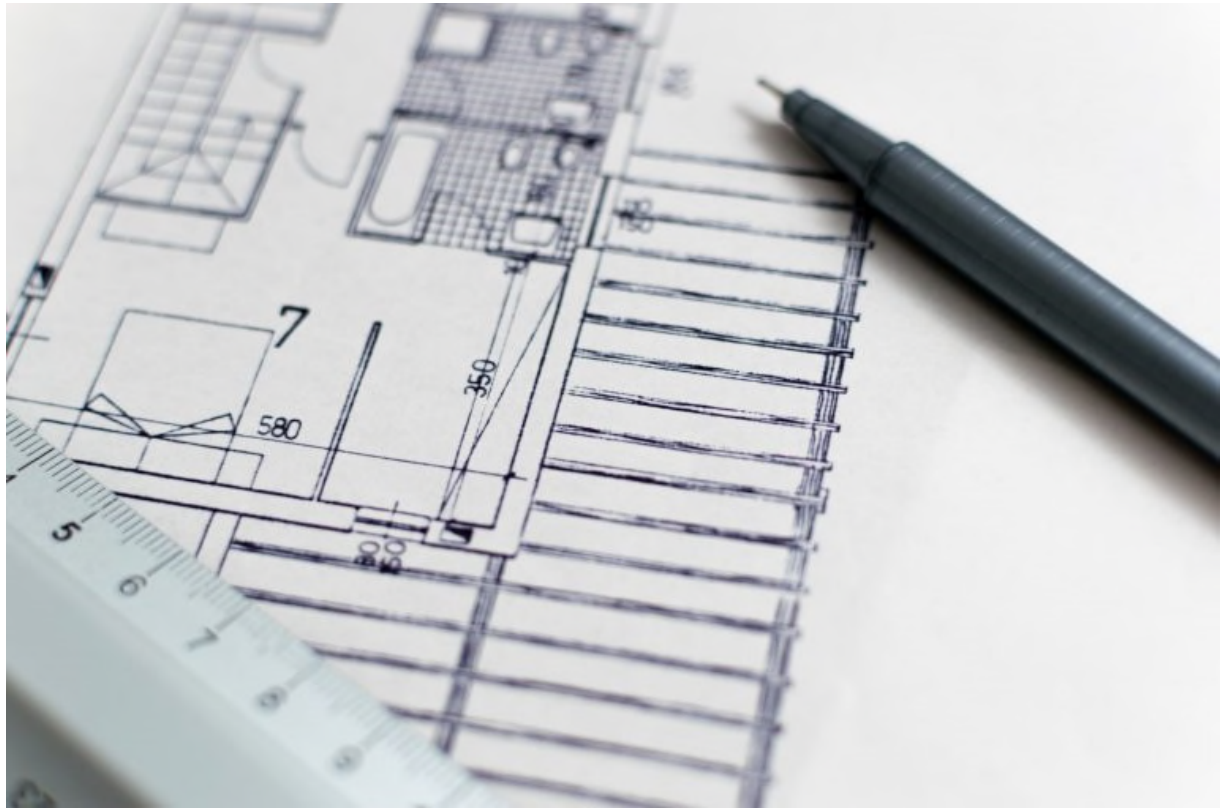
Cloud & Governance

- Design for **Public Cloud,** run on-prem
- **Declare** and **Automate** everything
- Built-in **Multi-Tenancy, Security** and **Compliance**
- Foundation for **Cloud-Native Solutions**

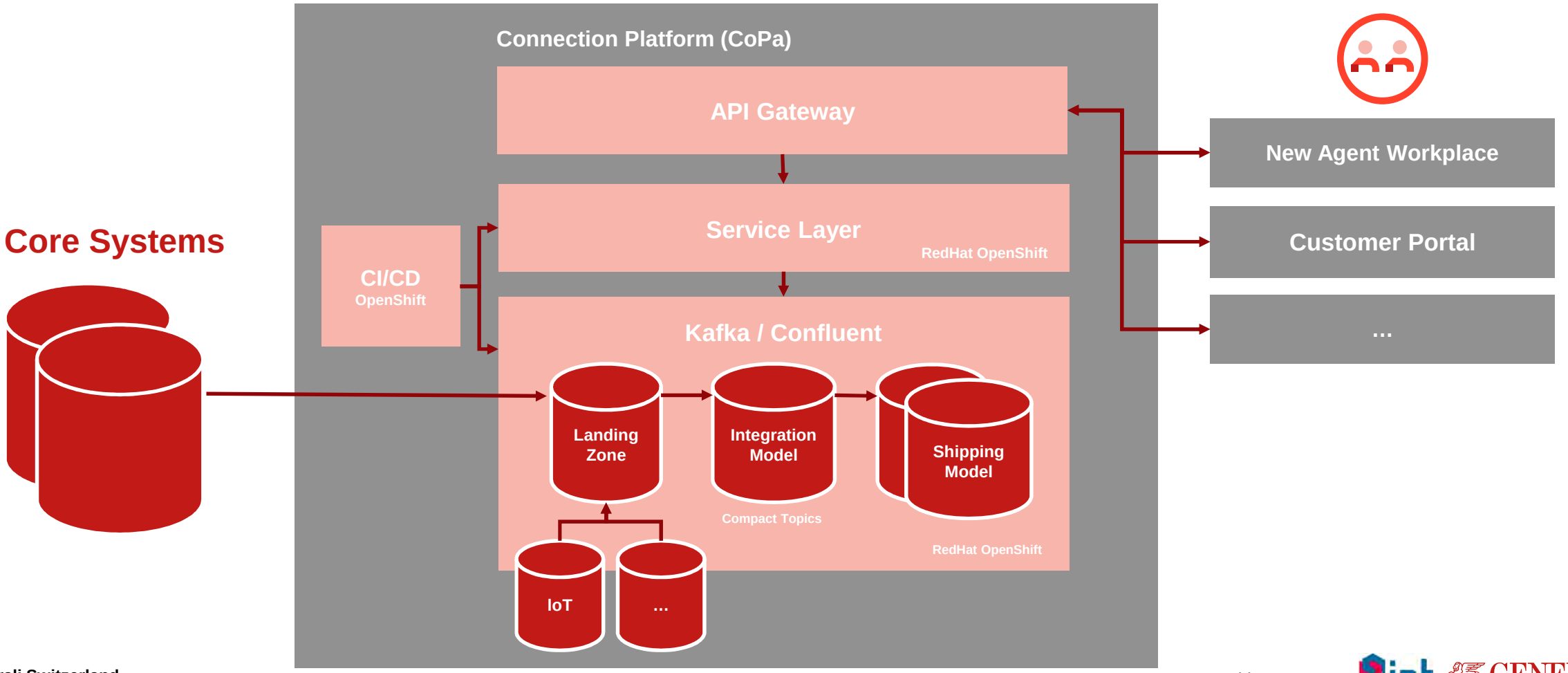
Data & Events

- Enterprise **Data & Event Backbone**
- Provide **CDC Streaming**
- Democratized **Stream Processing**
- **Low Footprint** on **Legacy**

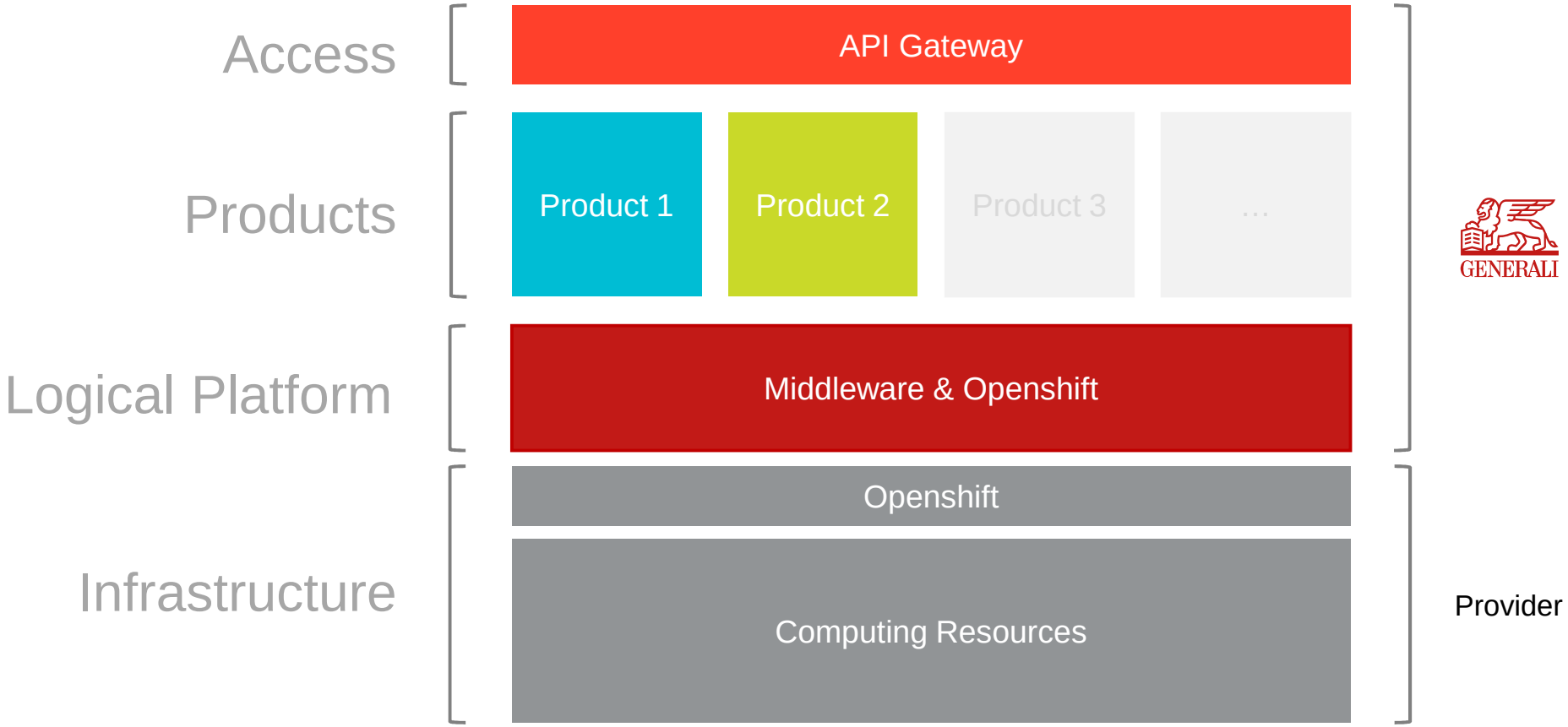
Technical Architecture



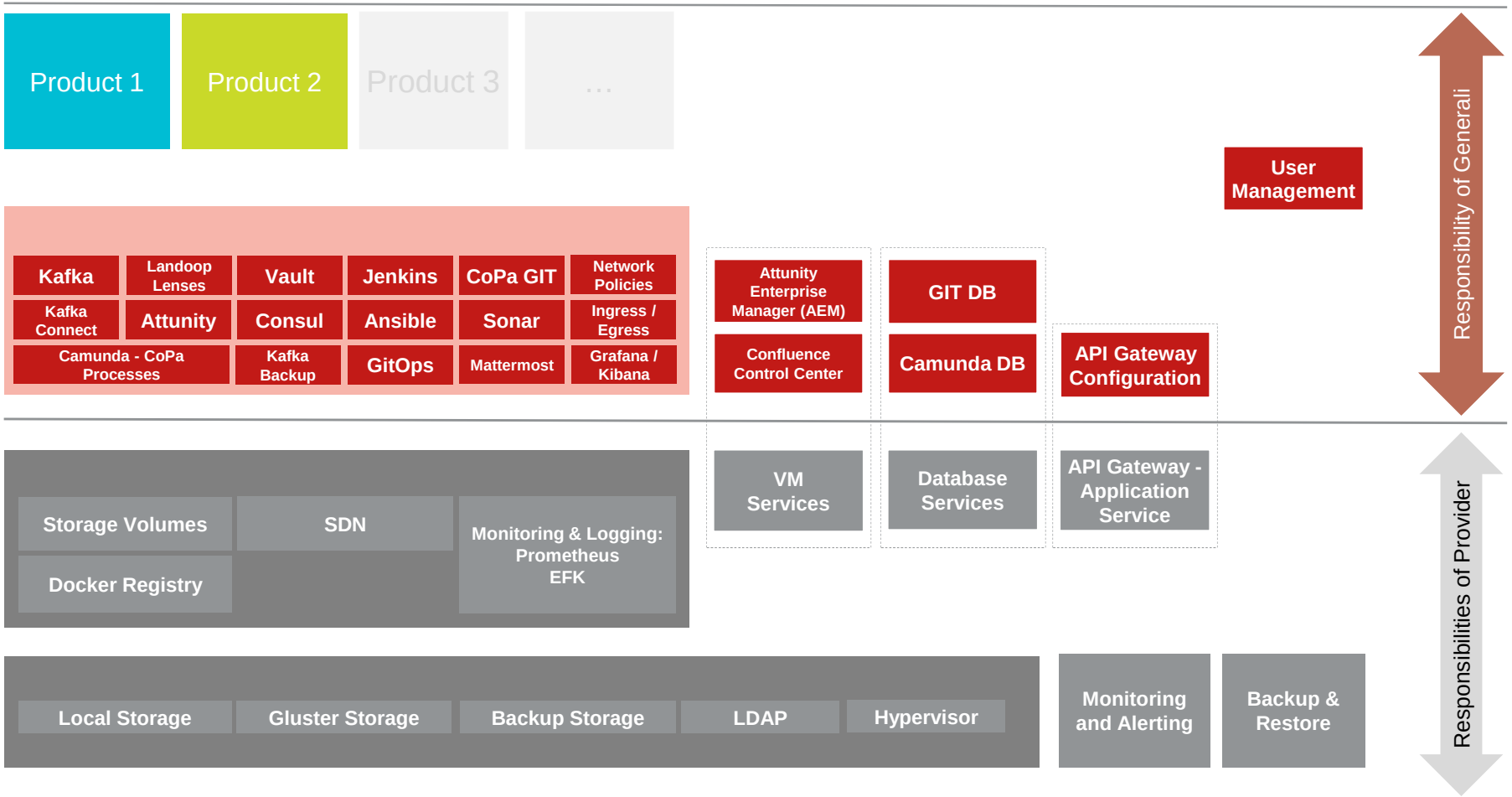
The high-level solution sketch is driven by four key technology decisions



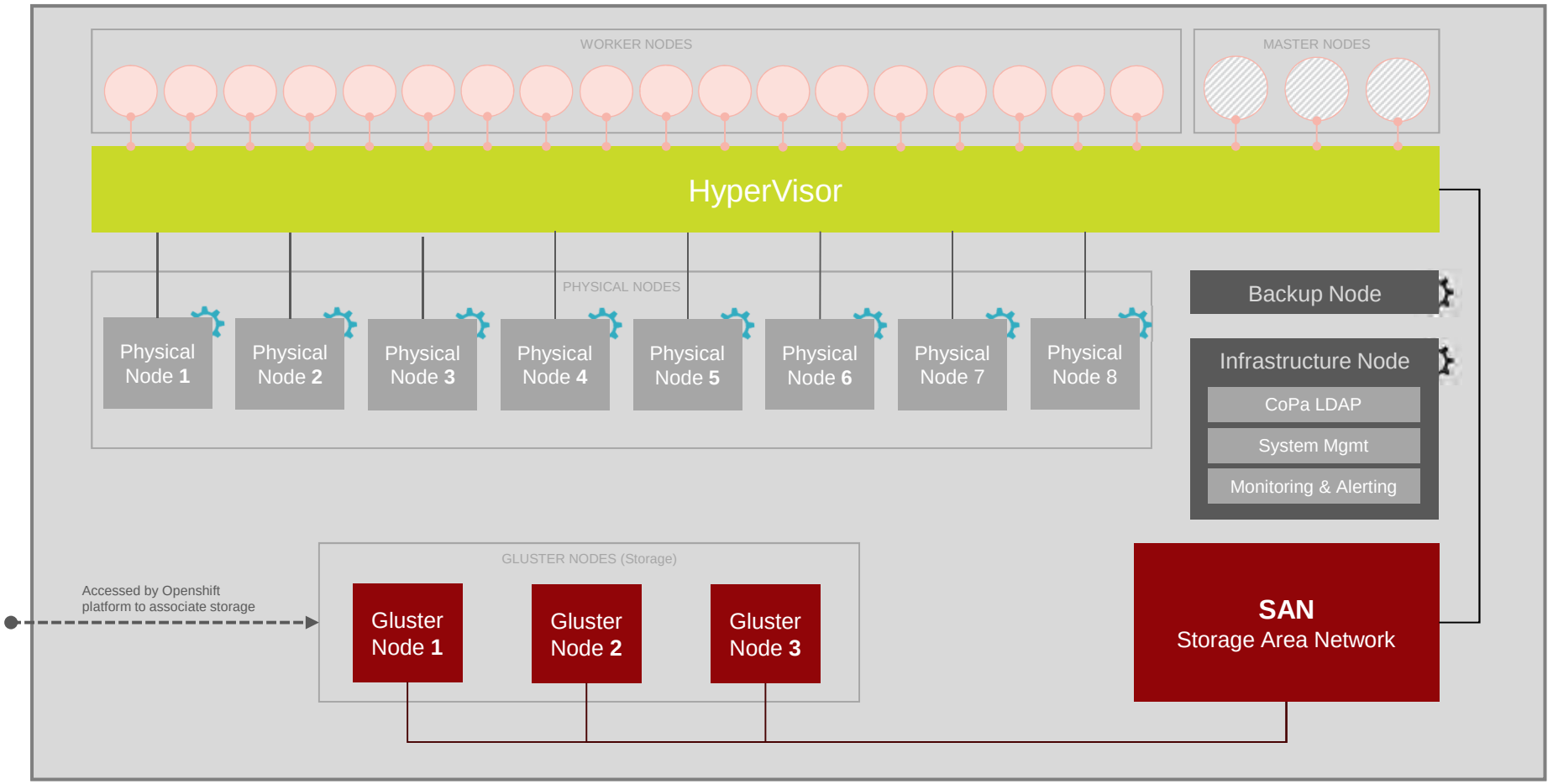
Infrastructure, Platform and Products



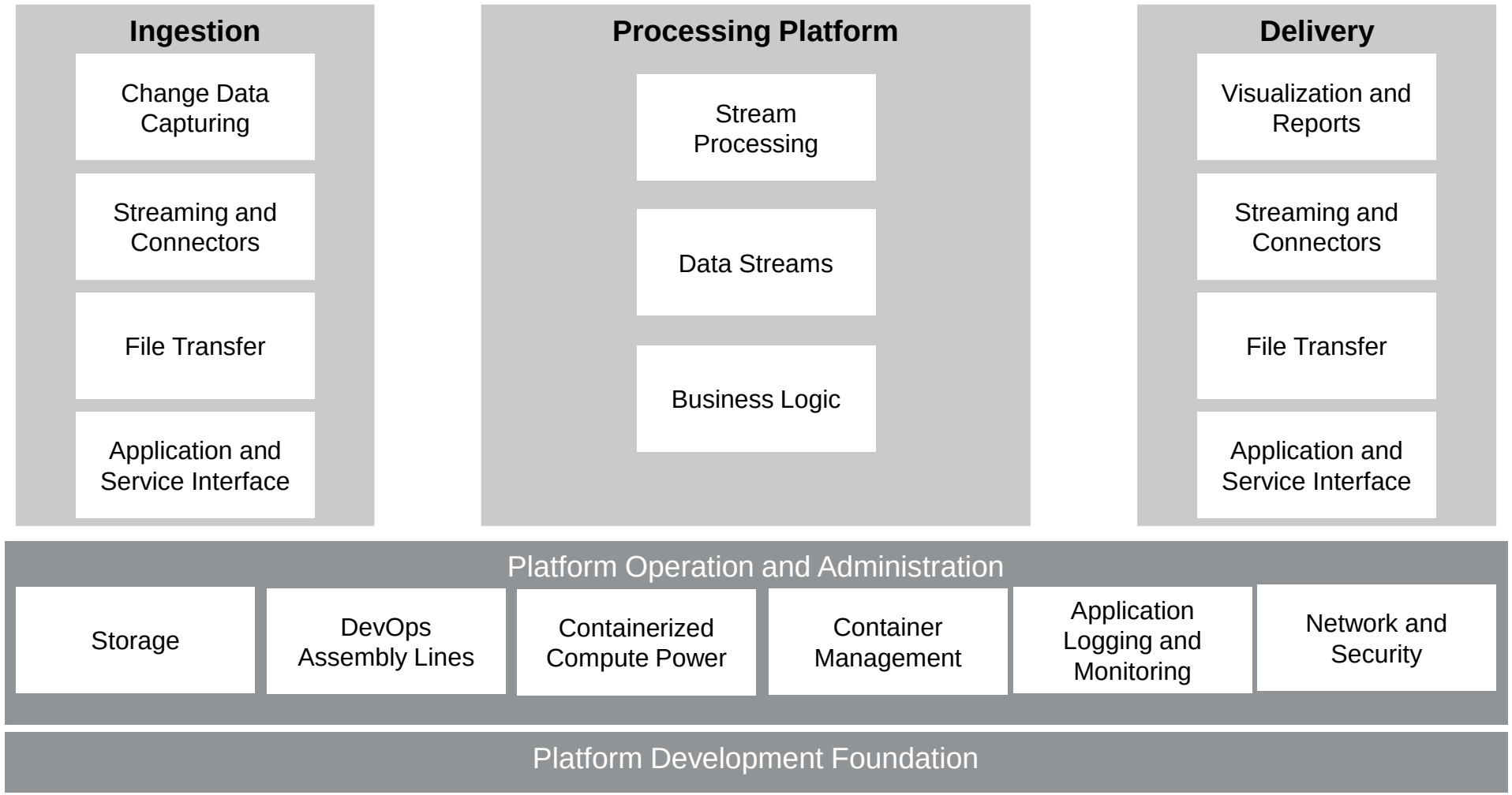
Components


Product 1
Product 2
Product 3
...
User Management
Responsibility of Generali
Kafka
Landoop Lenses
Vault
Jenkins
CoPa GIT
Network Policies
Kafka Connect
Attunity
Consul
Ansible
Sonar
Ingress / Egress
Camunda - CoPa Processes
Kafka Backup
GitOps
Mattermost
Grafana / Kibana
Attunity Enterprise Manager (AEM)
Confluence Control Center
GIT DB
Camunda DB
API Gateway Configuration
Storage Volumes
SDN
Docker Registry
Monitoring & Logging: Prometheus EFK
VM Services
Database Services
API Gateway - Application Service
Local Storage
Gluster Storage
Backup Storage
LDAP
Hypervisor
Monitoring and Alerting
Backup & Restore
Responsibilities of Provider

Computing Resources

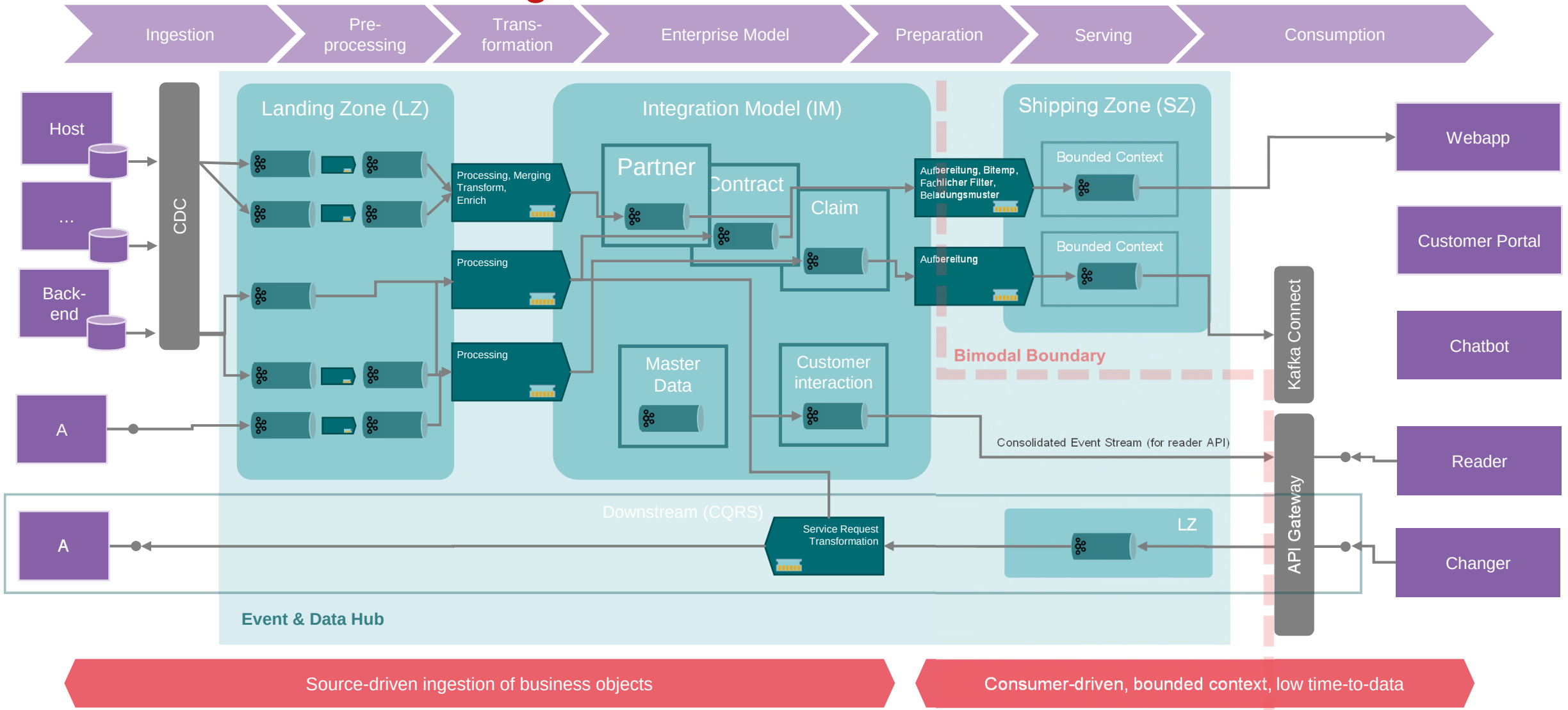


Technical Capabilities for Ingestion, Processing and Delivery



Design

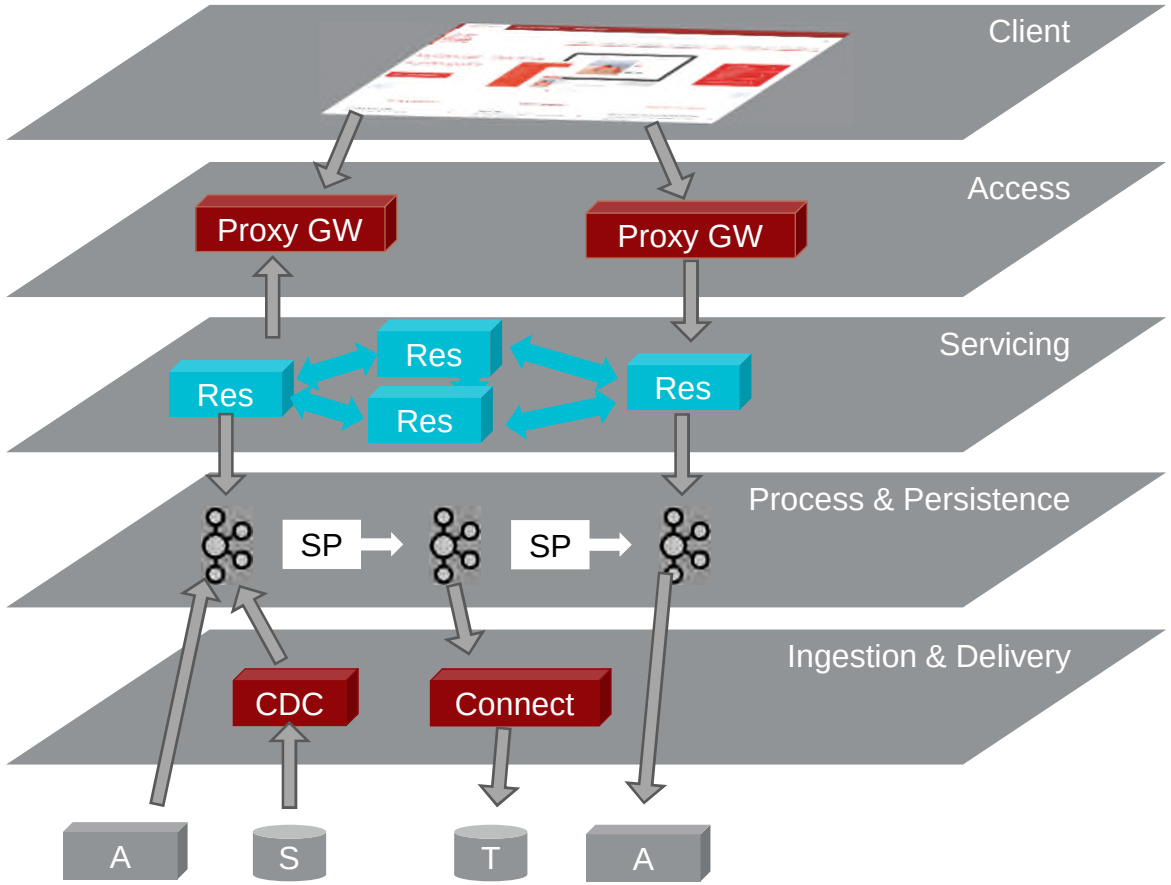
The Meta Model of all Integrations



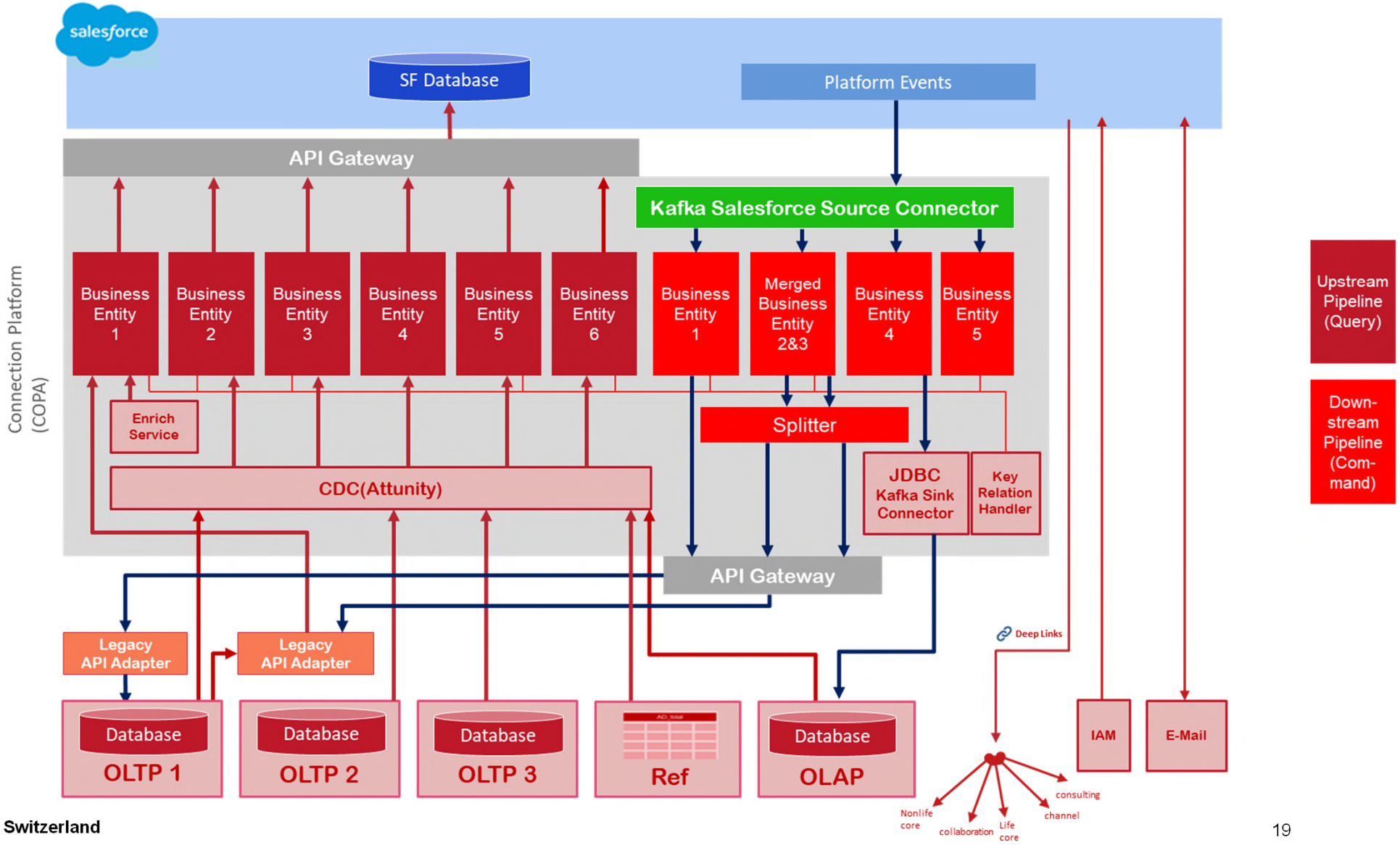
CoPa Logical Layers

- Sources and targets are served through ingestion & delivery layer
- ‘External’ clients are served through servicing layer implementations (Resources)
- Servicing layer translates from external to internal technical identities

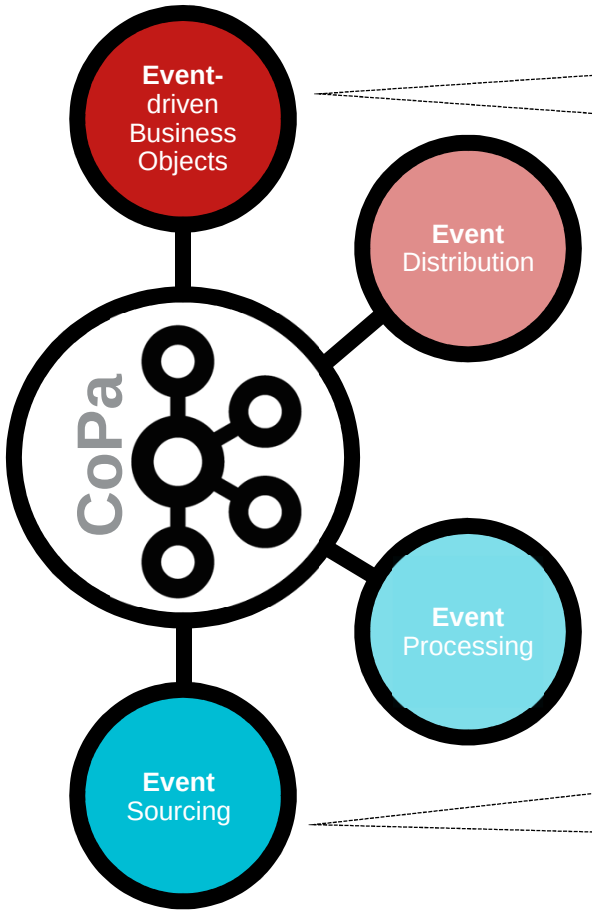
SP = Stream Processing
Res = Resource
GW = Gateway
CDC = Change Data Capture



A Concrete Integration Solution for a Cloud CRM



Patterns for the Kafka Cluster

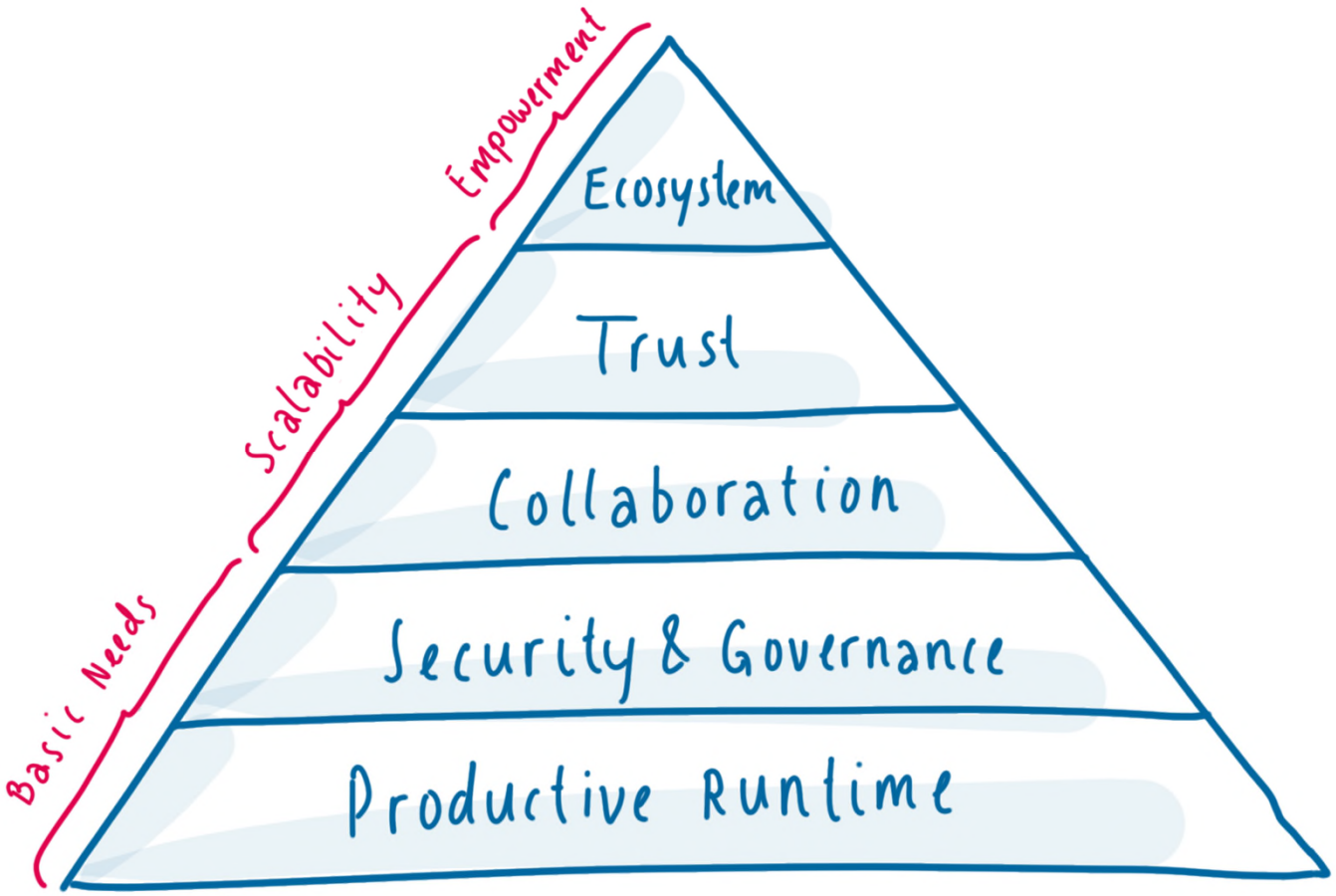


	Strong, centralized governance	Cleanup Policy	Complexity
Business Objects for data-driven system-integration & decoupling aka near real-time business object replication	✓	**compact**	**high**
Distribution Use Kafka as plain old messaging system**:** publish/subscribe, queues, data ingestion, IoT events, data lake ingestion, etc…	✓	**delete** (eventually)	**low**
Processing event streams in near real-time, e.g. click stream analysis for webapp, GPS locations, sensor data, etc…	⊘	**delete** (eventually)	**medium**
Event Sourcing as application pattern. Incrementally store all modifications to objects. Any state in time is reproducible. Great for auditability	⊘	**none**	**high**

Challenges

(adapted from Patrick Bönzli, spoud.io)

Maslow's Pyramid of Enterprise Kafka.



Central nervous system,
bridge to other departments or external organizations, consolidated view of the world

Reliable content, added value, traceability, lineage, data quality

Data Hub, single point of truth, ownership, schema, data lifecycle

Non-functional requirements, access control, encryption, auditability

Functional requirements, monitoring, elasticity, product lifecycle

Some Challenges of

Productive Runtime

- **Storage**
  - Gluster filesystem without block storage
- **Custom Deployment in Pre-Operator Era**
  - Anti-affinity groups
  - Health check
  - Kubernetes StatefulSet
- **Complex Parameter Tuning**
  - Replication Factor
  - Offset Retention
  - Default Number of Partitions & Filehandles
- **Compute and memory provisioning**
  - Stick to the suggested values

Some Challenges of

Security and Governance

ACL & SSL client auth
(Confluent >=5.3 now offers SASL/Kerberos)

- Certificate management is cumbersome
- Self-signed certs are frowned upon
- Zookeeper does not allow mutual SSL -> SASL
- KIP-500 aims to abandon zookeeper

Rest proxy, aka the man in the middle

Needs copy of private key of client

Multi DC and disaster recovery

For RPO=0 you need stretched cluster
(Confluent >5.2 has synchronous replicas)

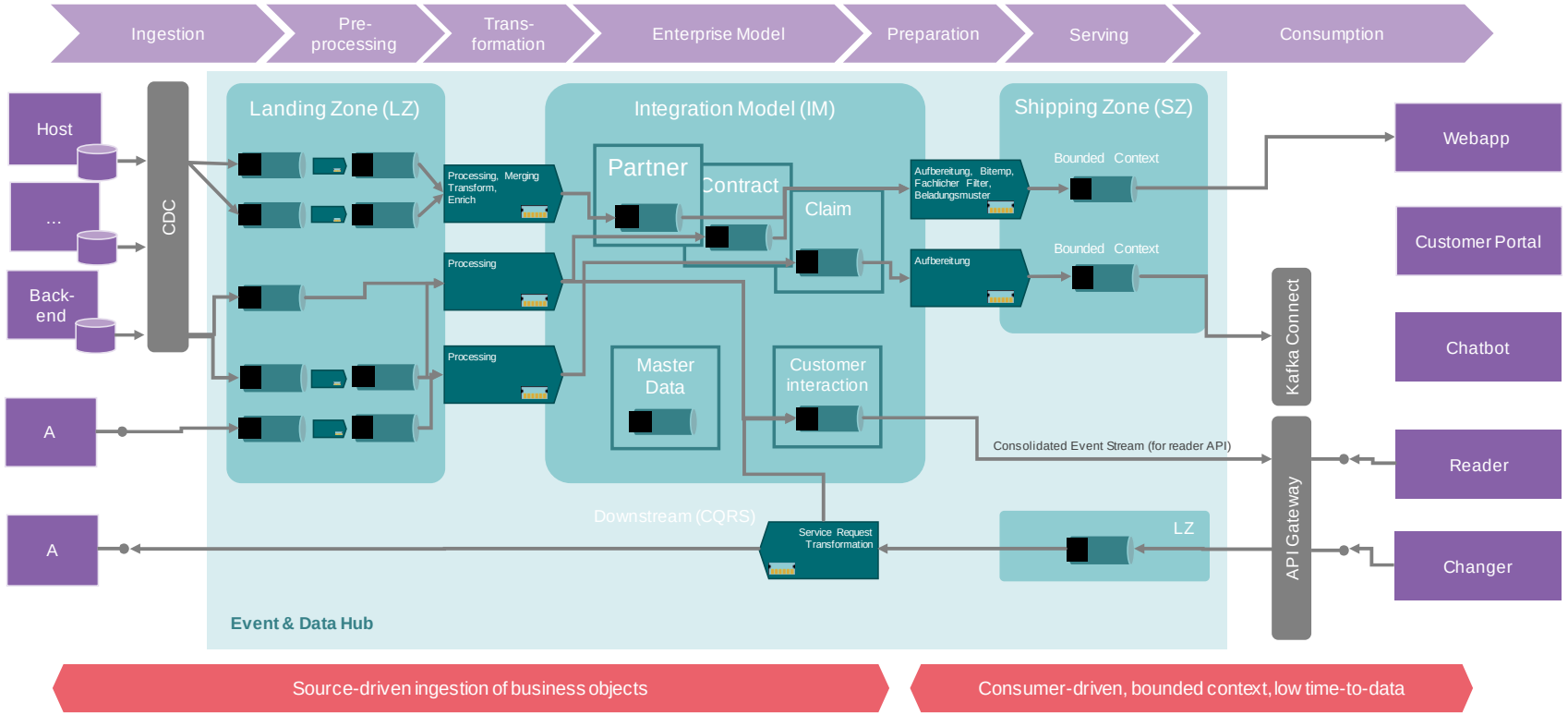
Multi-tenant Kafka Streams

Kafka Streams generates (a lot of) internal topics, possible naming conflict

Some Challenges of

Collaboration

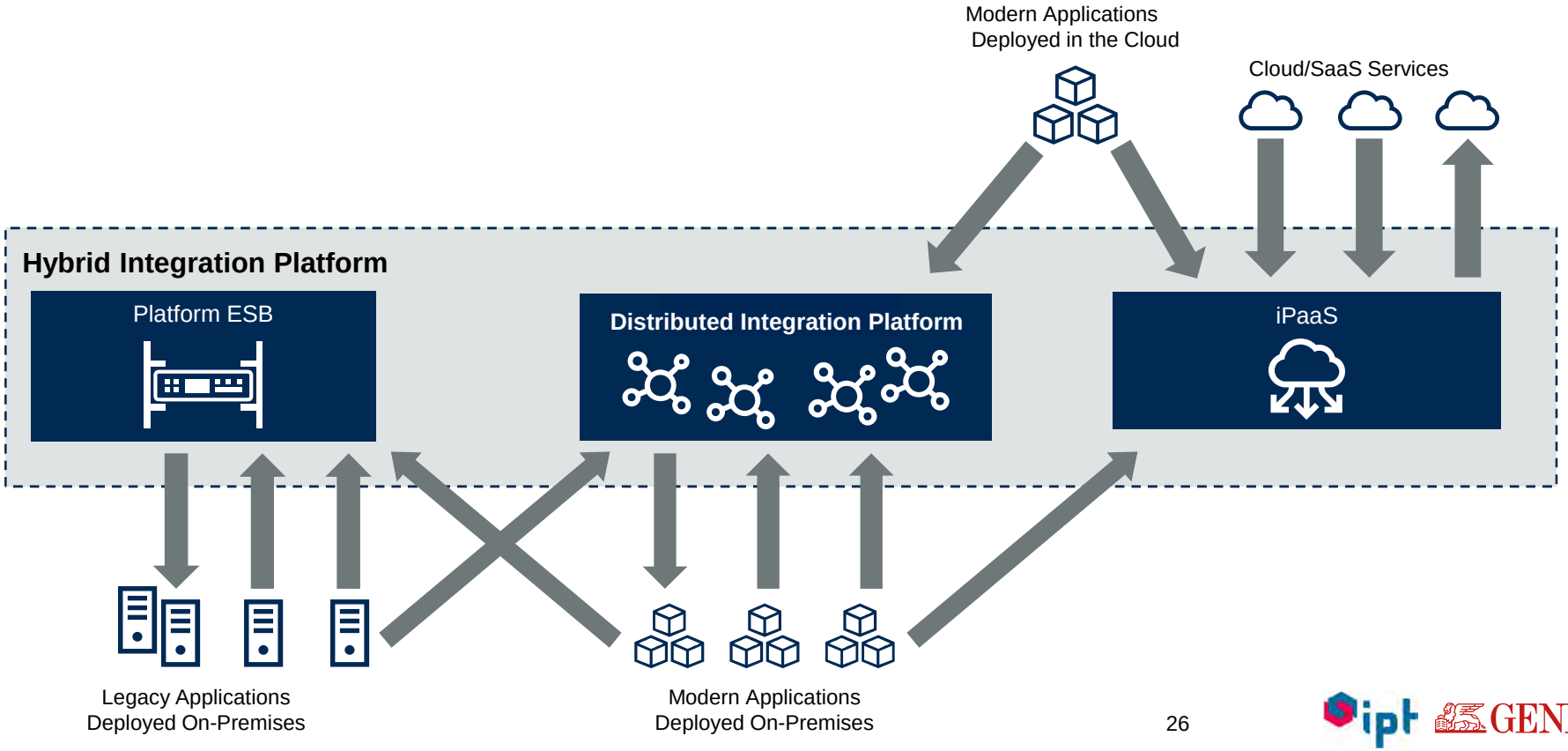
- Structure, patterns, continuous enabling, reviews
- Promote self-service & ease of use
- AVRO and Schema Registry (Data Governance)
- Landing – Integration – Shipping



Some Challenges of

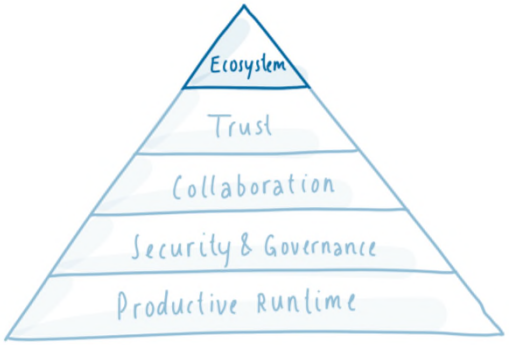
Trust

- Tracability, lineage and topology
- Content-level encryption
- Who is responsible for integrations?
- Centralized responsibility vs. open platform → Anti-ESB pattern



Some Challenges of

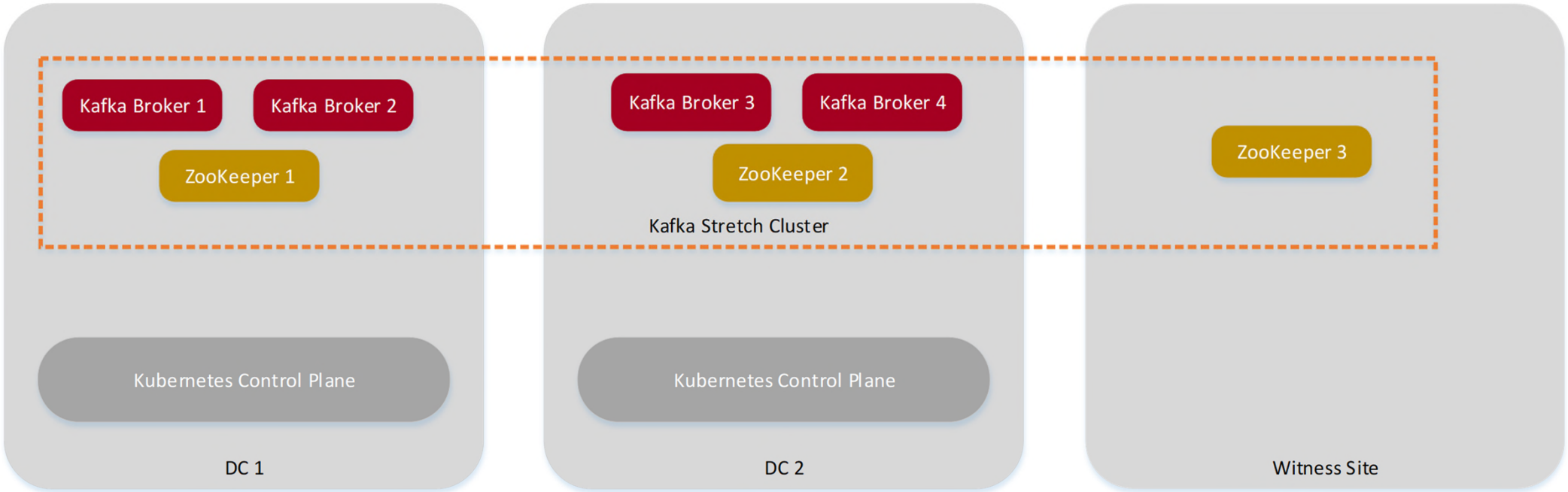
Ecosystem



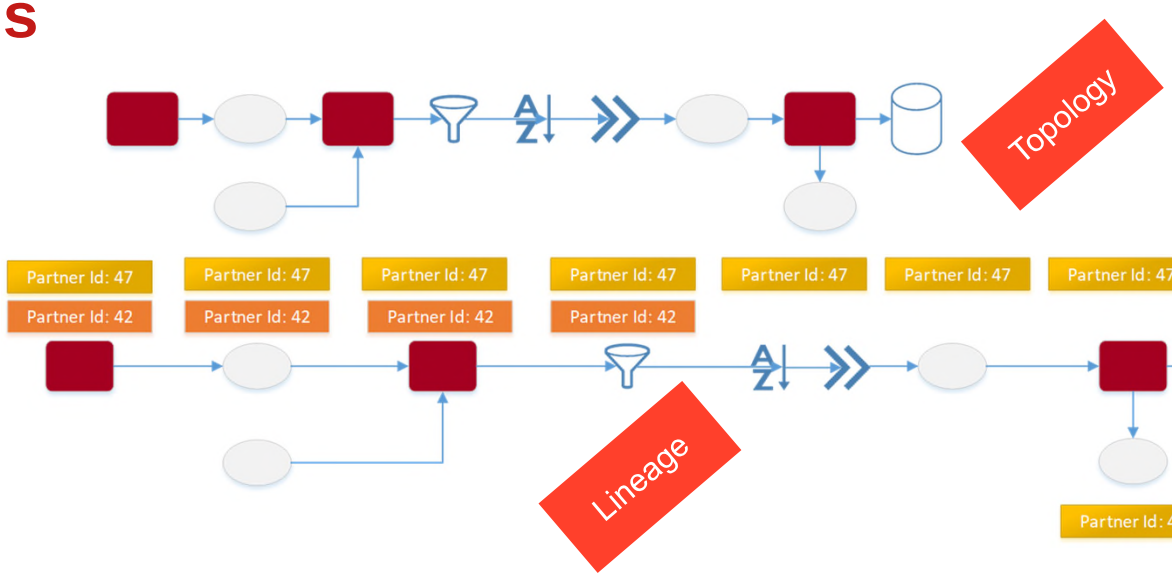
- No insights yet
  - Level 3 achieved
  - Level 4 in progress
  - Level 5 is undiscovered territory

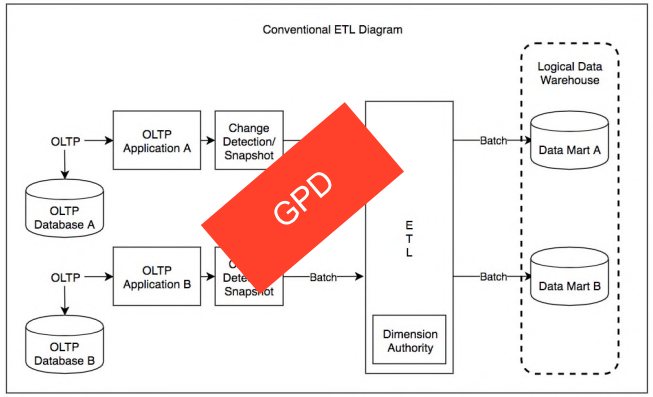
Next Steps

Multi DC


Kafka Broker 1
Kafka Broker 2
ZooKeeper 1
Kafka Broker 3
Kafka Broker 4
ZooKeeper 2
Kafka Stretch Cluster
ZooKeeper 3
Kubernetes Control Plane
Kubernetes Control Plane
DC 1
DC 2
Witness Site

Data Flow Extensions





Gitops

Copa Desired State

Git repo L3 "CDS"

Depends on

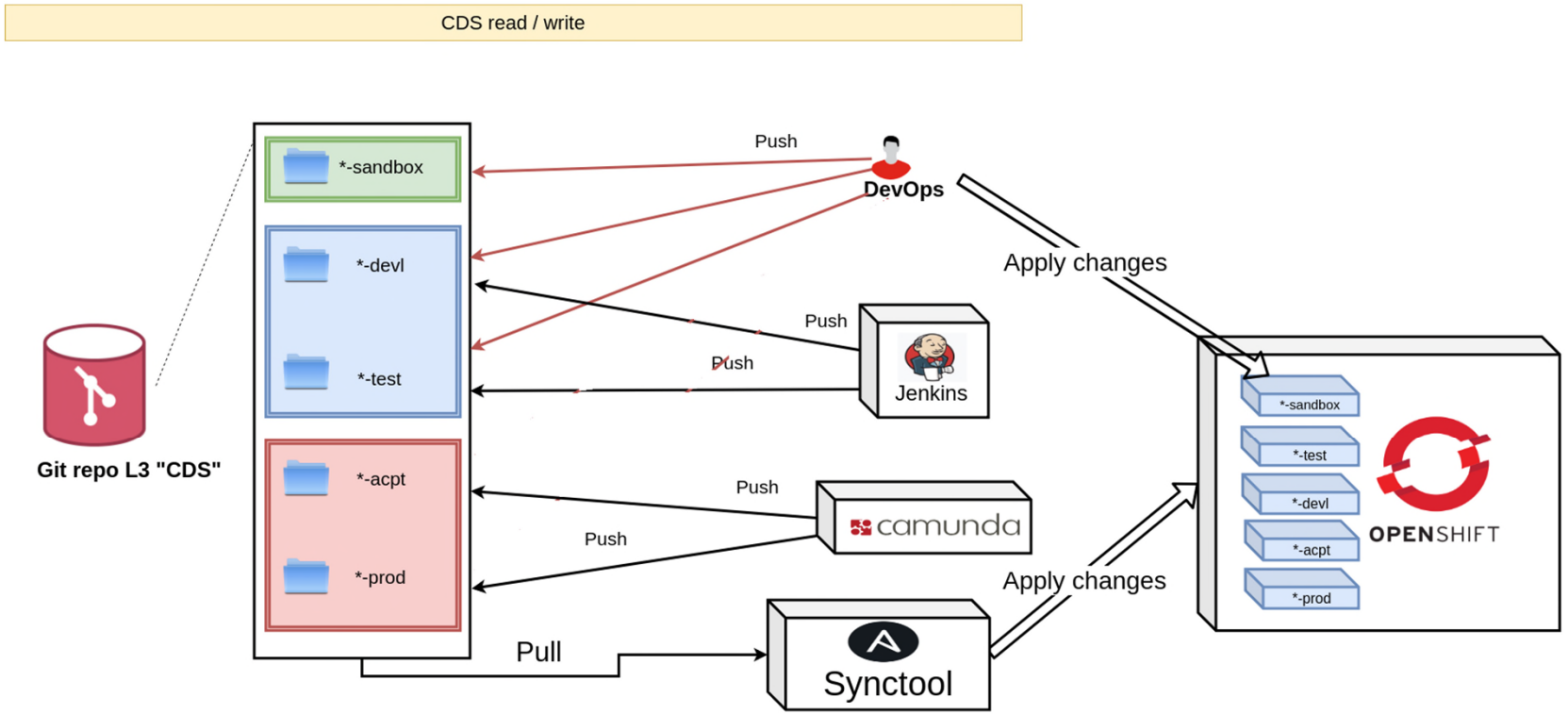
Deploy Image

Git repo L2

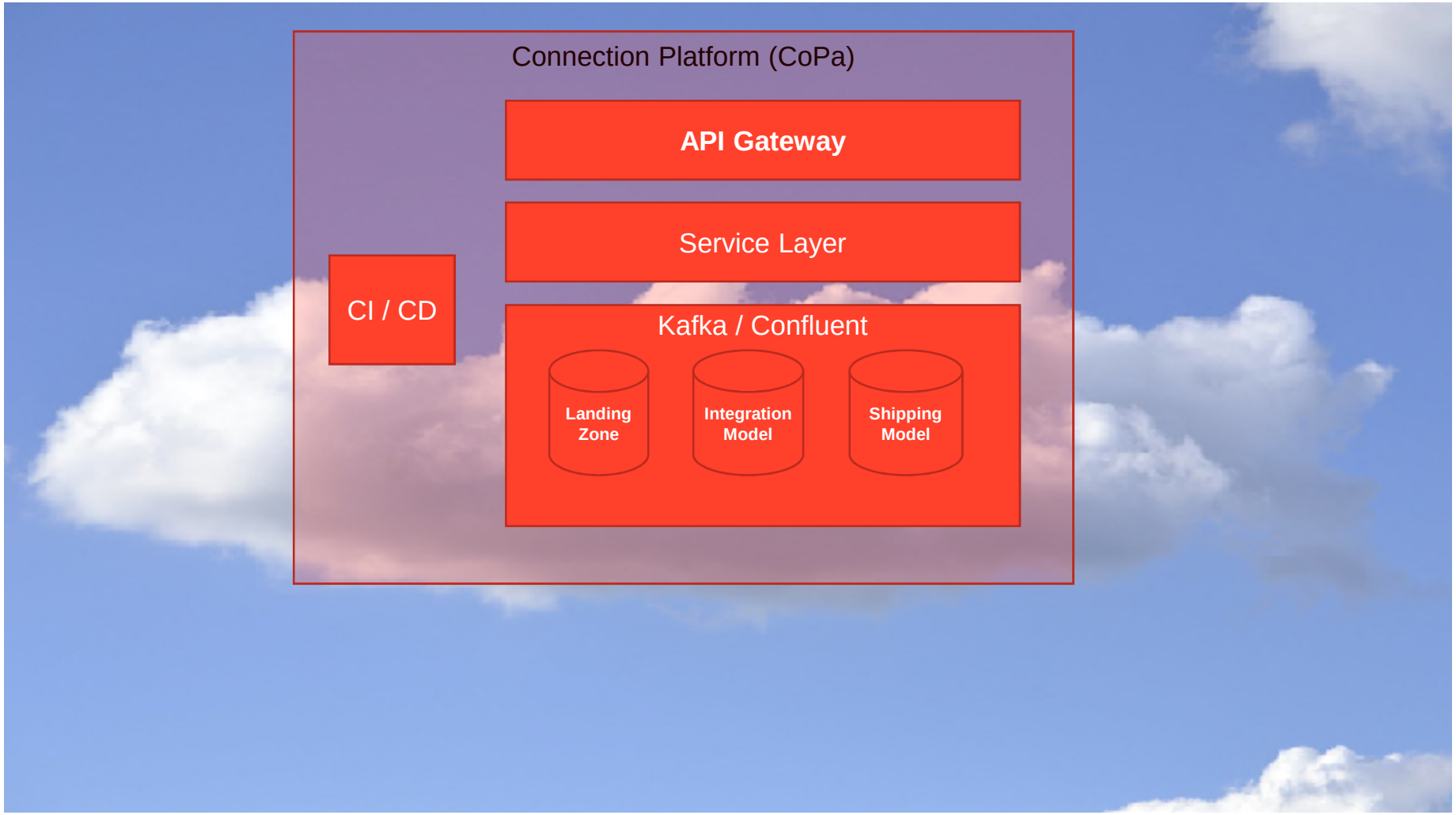
Depends on

Build Image

Git repo L1

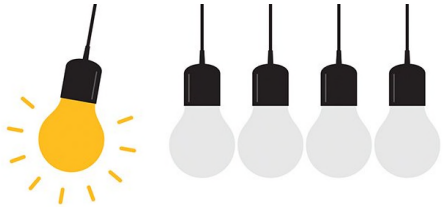


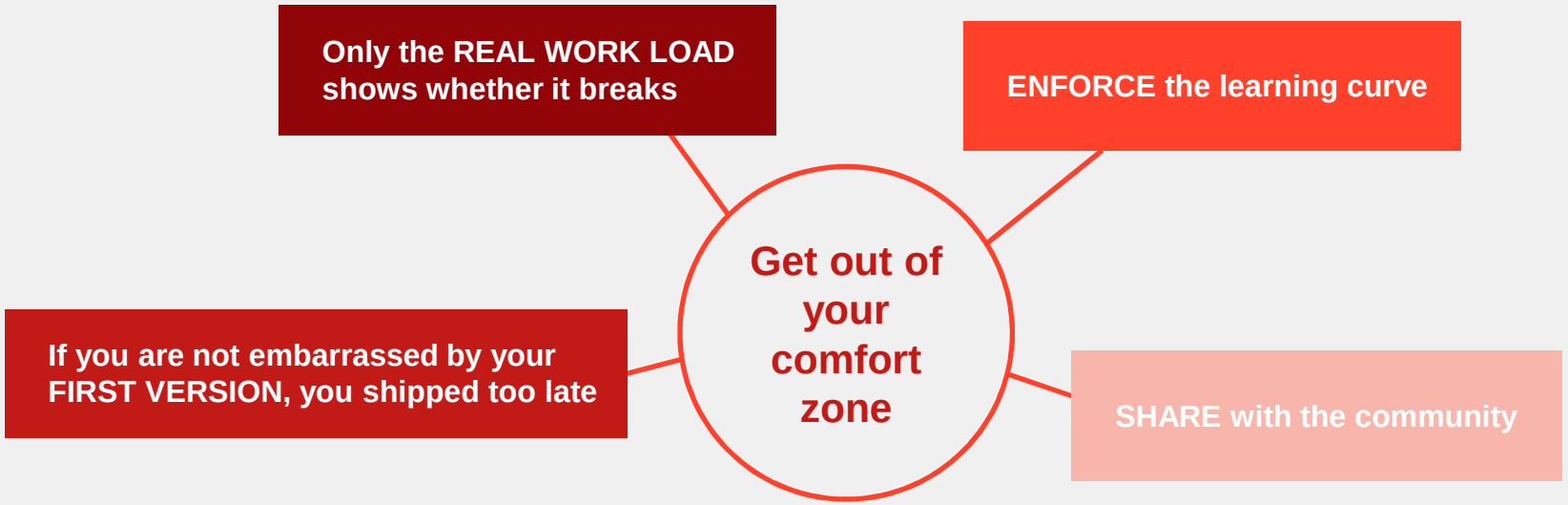
Silver Linings in the Cloud



Key Take Aways

Learnings on ‘Fast Time to Market’

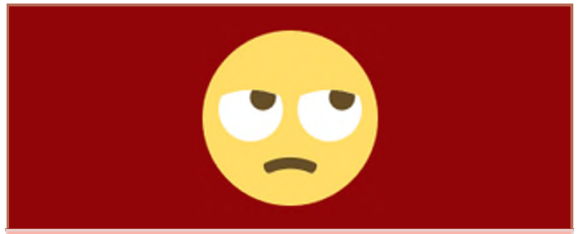




Learnings on ‘Apache Kafka’

- Mind your Storage Technology
- Don’t Under-Provision Broker and ZK
- Mind Offset Retention

- Configure for Durability
- Rolling Upgrade is Cool
- Use Compact Topics and a Schema Registry

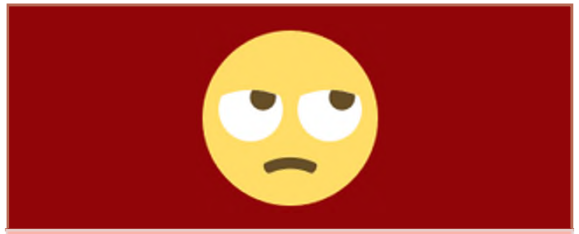


- Security Takes Time
- Disaster Recovery (Geo-Redundancy) and Multi-DC (Stretching) are Hard

Learnings on ‘OpenShift’

- Mind your Storage Technology
- The underlying Infra matters
- Beware of Bundled Monitoring & Logging

- Enterprise Ready Security
- Community
- Bring your Nodes Down Often



- Don’t Over-Provision
- Recent Acquisition

Questions?

https://commons.wikimedia.org/wiki/Category:Raised_hands_at_press_conferences#/media/File:President_Trump's_Trip_to_Vietnam_News_Conference_(32296565547).jpg

Kontakt

Thomas	thomas.peter@generali.com
Abhishek	abhishek.pande@generali.com
Yves	ybr@ipt.ch