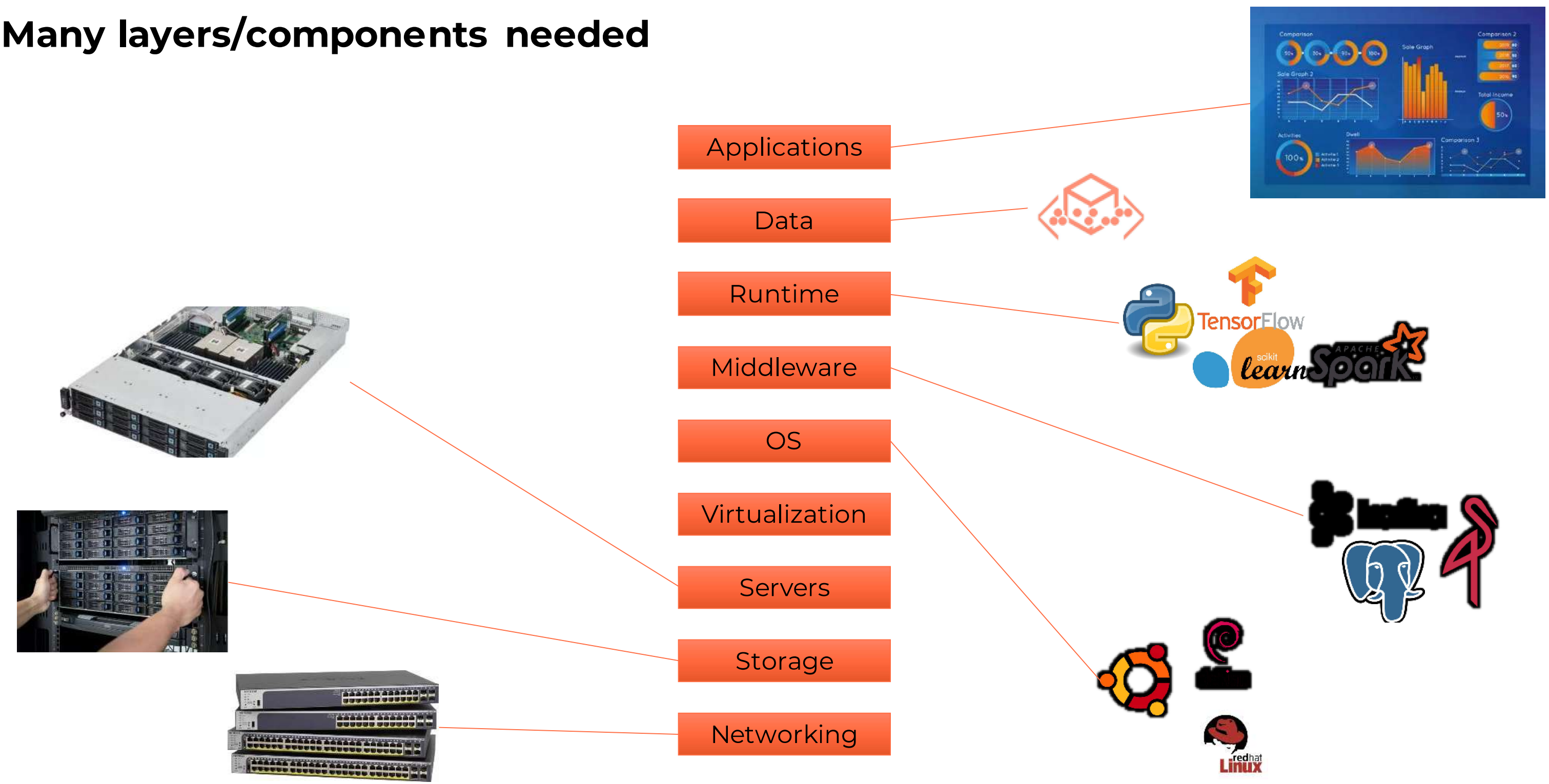
EVIDEN

01 Data platform Foundation

Dissection of an application

Many layers/components needed



Many ways to manage these layers

Different service offers

You manage | SP manage

Private cloud

Public cloud

On premise	BMaaS	IaaS	CaaS	PaaS	FaaS	SaaS
Applications	Applications	Applications	Applications	Applications	Applications	Applications
Data	Data	Data	Data	Data	Data	Data
Runtime	Runtime	Runtime	Runtime	Runtime	Runtime	Runtime
Middleware	Middleware	Middleware	Middleware	Middleware	Middleware	Middleware
OS	OS	OS	OS	OS	OS	OS
Virtualization	Virtualization	Virtualization	Virtualization	Virtualization	Virtualization	Virtualization
Servers	Servers	Servers	Servers	Servers	Servers	Servers
Storage	Storage	Storage	Storage	Storage	Storage	Storage
Networking	Networking	Networking	Networking	Networking	Networking	Networking

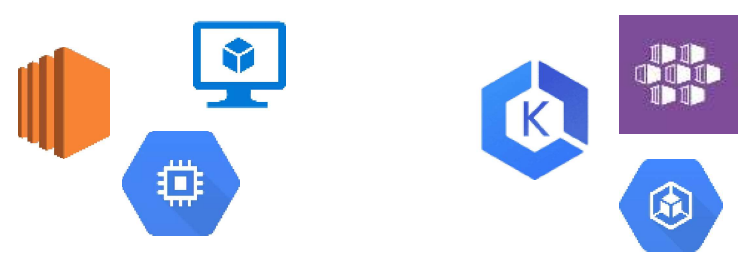
Many ways to manage these layers

Some examples

Private cloud

Public cloud

On premise	BMaaS	IaaS	CaaS	PaaS	FaaS	SaaS
- Datacenter	- X86 servers - NAS storage	- AWS EC2 - Azure VM - Google GCE	- AWS EKS - Azure AKS - Google GKE	- AWS Appsync - Azure App Service - Google App Engine - Heroku	- AWS Lambda - Azure Function - Google Cloud Function	- Salesforce - Office360 - AWS Redshift - Azure Synapse - Google BigQuery



Few or no vendor locking

Migration can be very complex depending on the data service used.
Strong vendor locking, but Dev and Ops teams are strongly reduced.

« Cloud provider agnostic » ⟷ « NoOps experience »

Cloud or not cloud ?

No « one size fit all » solution, many deployment modes available

Full cloud

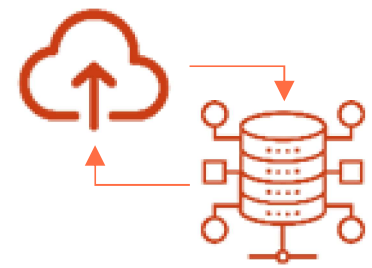
Agile, no ops, secured

Could be expensive

Multi cloud

Best of breed

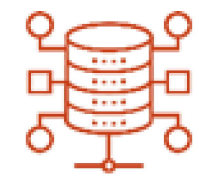
Complex management
Lots of external network



Hybride

Trade off between both worlds
Offloading

Complex management



On premise

Usually well mastered by companies

Not scalable, lake of agility and TTM

Vision

Atos is investing a lot in Sovereignty & Data securization

Atos is a founding member of Gaia-X, launched in may 2020. Gaia-x is a European initiative for improving the interoperability and the sovereignty of the cloud. Atos is working closely with Gaia-x organization to define the future standards for sharing data (data space).

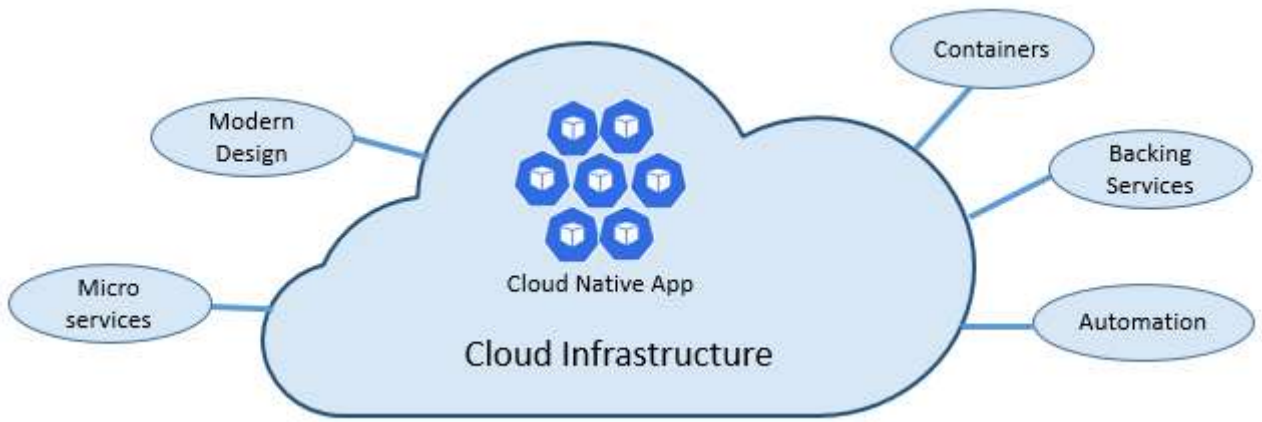
In July 2020, European court of justice has denounced the privacy shield between USA/Europe especially for GDPR compliancy issues & cloud act extraterritoriality application. This decision has accelerated the needs and investments on sovereign approach especially for public sector.

Atos has been selected by French Ministry of Defense for developing its sovereign BigData platform. This success has given birth in 2021 to a joint venture between Thales and Atos called Athea for handling the next steps of this strategic program. Atos Codex Data Platform is a civil fork of this sovereign platform based on open sources.

Reccently Atos has launched Atos OneCloud Sovereign Shield, which is a comprehensive edge to cloud platform ecosystem and highly secure service that improves the level of control clients have over the data they produce and exchange, helping them regain control and effectively deal with legal dependencies.

Cloud native architectures and applications

The core concepts



- Modern design => 12factor
- Microservices => cut down the monolith
- Containers => isolation
- Backing Services => Don't do all by yourself
- Automation => code once, run many

The Twelve-Factor App

I. Codebase
One codebase tracked in revision control, many deploys
II. Dependencies
Explicitly declare and isolate dependencies
III. Config
Store config in the environment
IV. Backing services
Treat backing services as attached resources
V. Build, release, run
Strictly separate build and run stages
VI. Processes
Execute the app as one or more stateless processes
VII. Port binding
Export services via port binding
VIII. Concurrency
Scale out via the process model
IX. Disposability
Maximize robustness with fast startup and graceful shutdown
X. Dev/prod parity
Keep development, staging, and production as similar as possible
XI. Logs
Treat logs as event streams
XII. Admin processes
Run admin/management tasks as one-off processes

XIII. API First
XIV. Telemetry
XV. Authentication/Authorization

ML in production

The real challenges

How to infer at scale to handle prediction spikes ?

When do we need to start re-training the model and which datasets should we use?

Are resources (CPU, GPU) being used efficiently ?

Is the model over or under scaled ?
Are the inference response times acceptable ?

Performance

How to containerize a model ?

Is there prediction protocol standards (http, grpc) ?

How to handle different framework and model format (Ttensorflow, Pytorch, Onnx,...) ?

Do you know the answers of all these questions ?

Build

How can you calculate accuracy if you don't know/get the truth immediately (feedback loop, ground truth)?

Is it necessary to run in production the best model or a combination of several models?

Which metrics to follow on a model (precision, recall, input data distribution, ...)?

Business impact

How can we ensure that the right versions of the models are deployed and that they use the right data for their prediction?

How to recover the training datasets in order to analyse the deviant behaviours of the model afterwards?

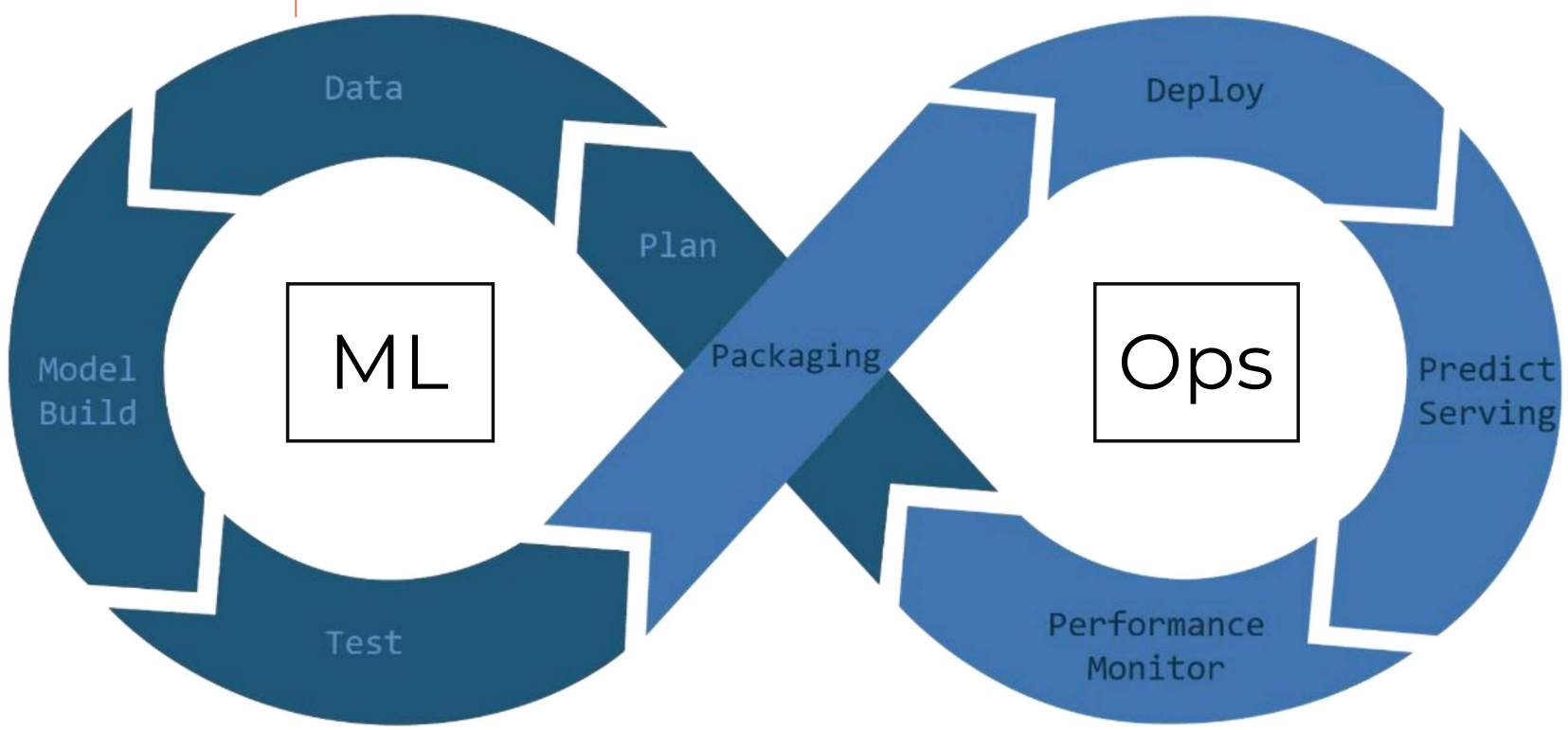
Exploit

MLOps

A methodology to industrialize ML

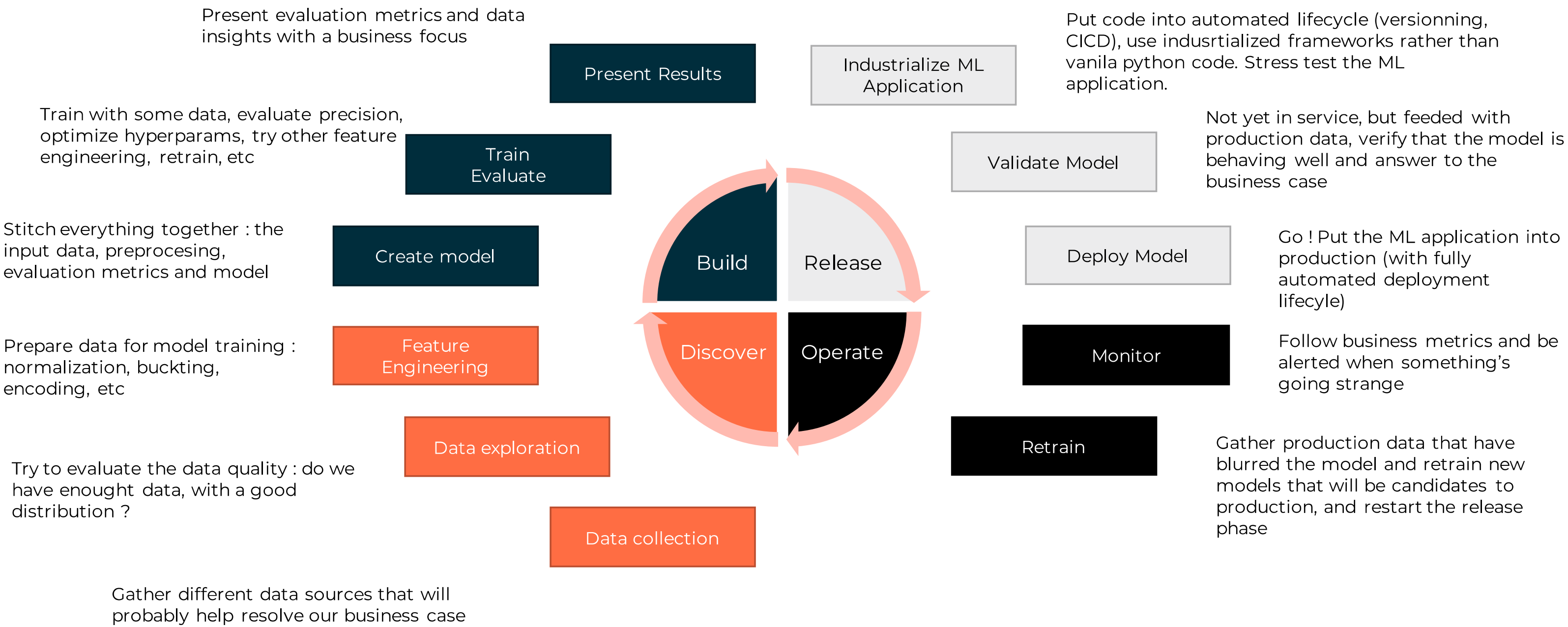
Data WorkFlow :
- Automated Data Ingestion
- Automated Data Analysis
- Automated Data Transformation

Deploy & Run :
- Serving at scale
- Explaining
- Monitoring & logging

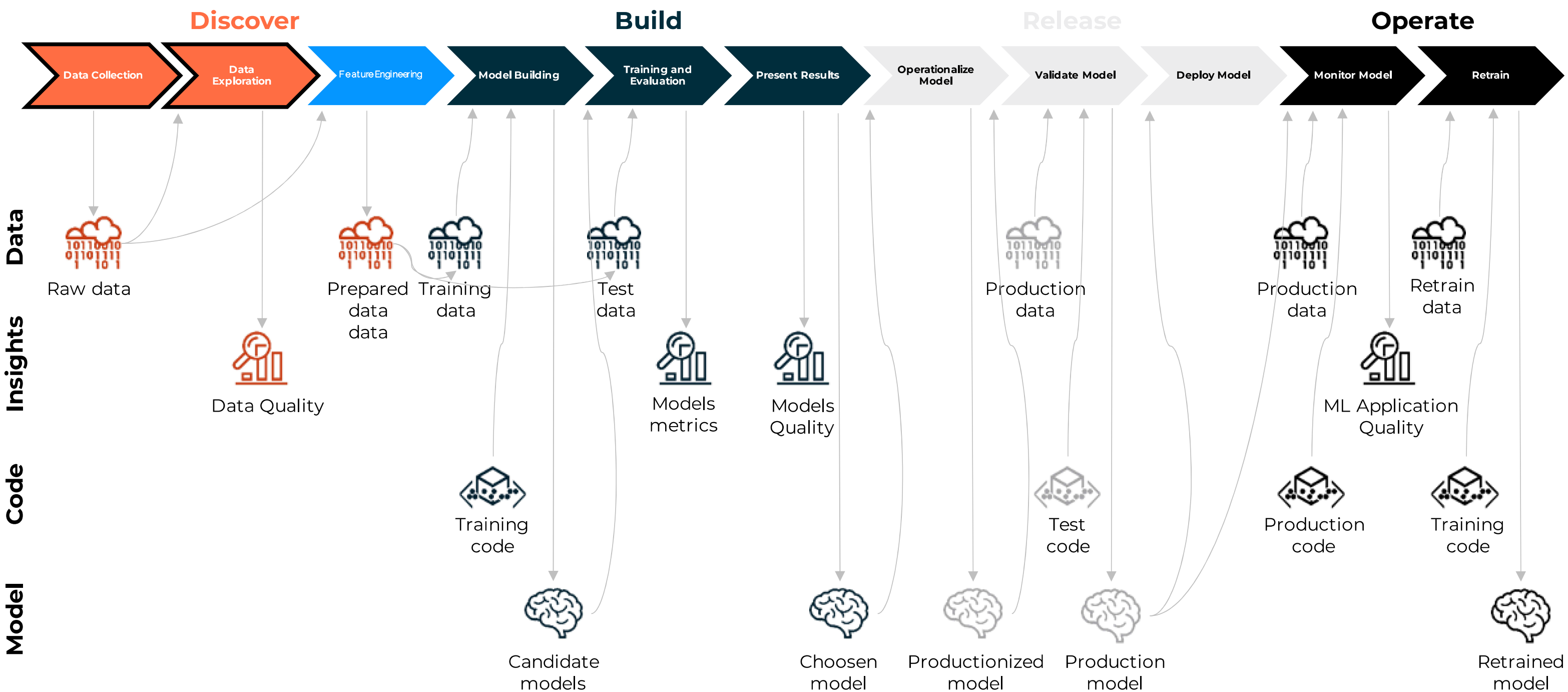


Model Operationalization :
- Training at scale
- Tuning
- Governance

ML lifecycle



ML lifecycle and artifacts



DataOps

What the heck is behind this buzz word ?

- Comes from Agile + Devops + Lean Development
- Same ambitions than now wellknown devops
  - Federate different teams around the product/value
  - Industrialize human and technical process
    - Automate most of dev/build/push actions into target environment
  - Help to handle the technical complexity of the ecosystem
- Key concepts
  - Value first
  - Collaboration
  - Automate
  - Orchestration, test, monitor
  - Security
- Objectives
  - Improve data and analytics quality
  - Reduce TTM

DataOps is a collaborative data management practice focused on improving the **communication**, **integration** and **automation** of data flows between data managers and data consumers across an organization.

Source : Gartner

Data Platform Functional Architecture

Lets create it

Tell me what are the components of a data platform

Security ?

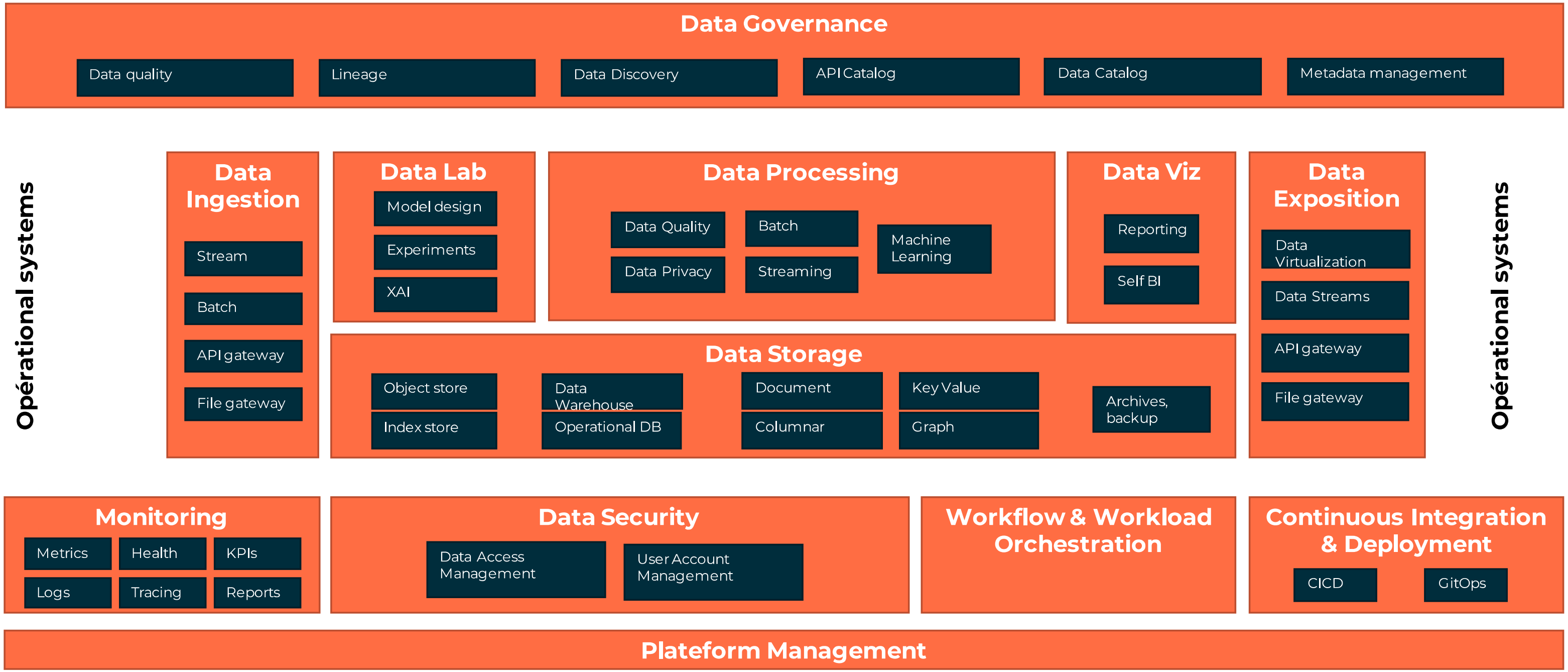
Processing ?

Data ingestion ?

...

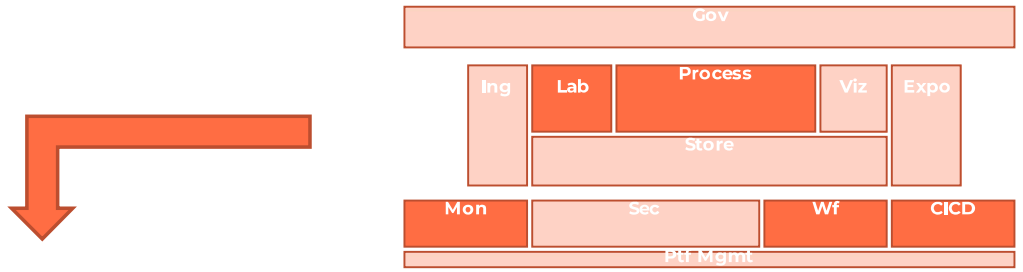
Data Platform Functional Architecture

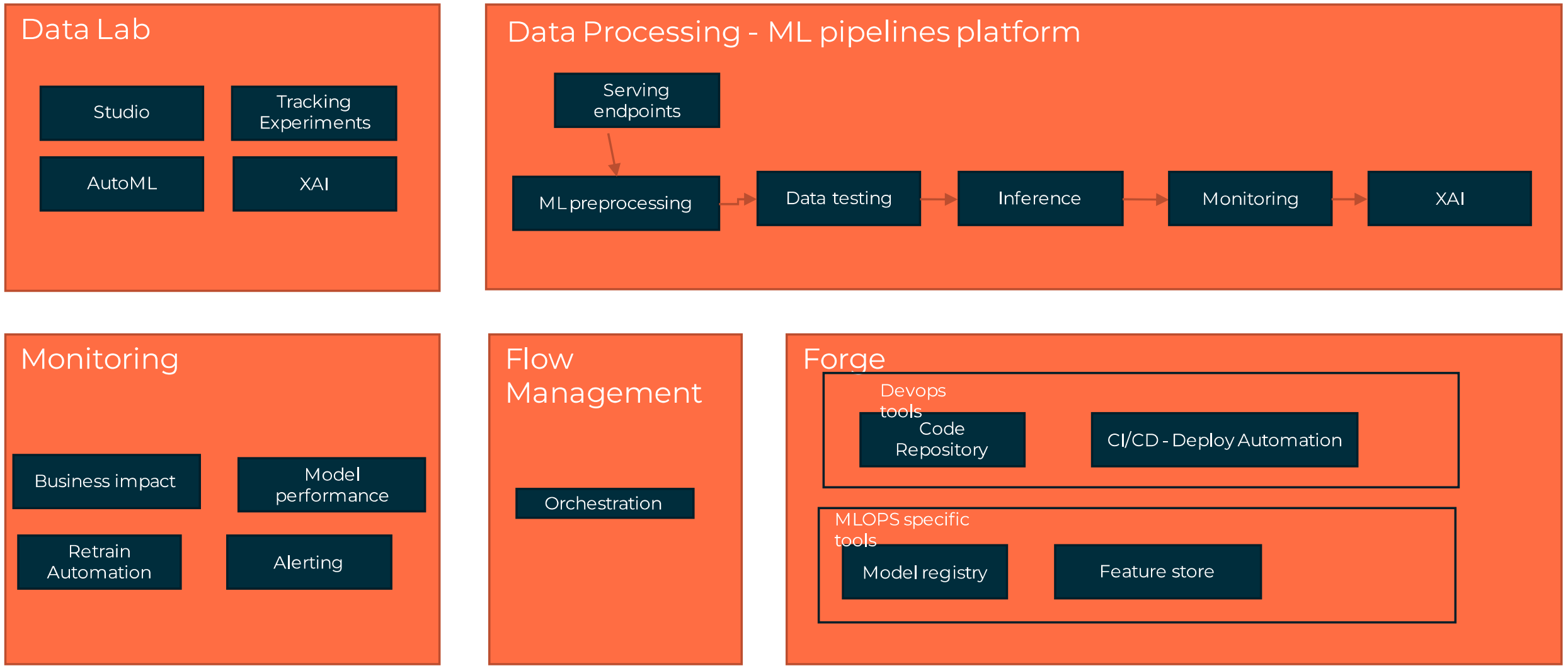
Big Big Picture



Data Platform Functional Architecture

MLOps Focus

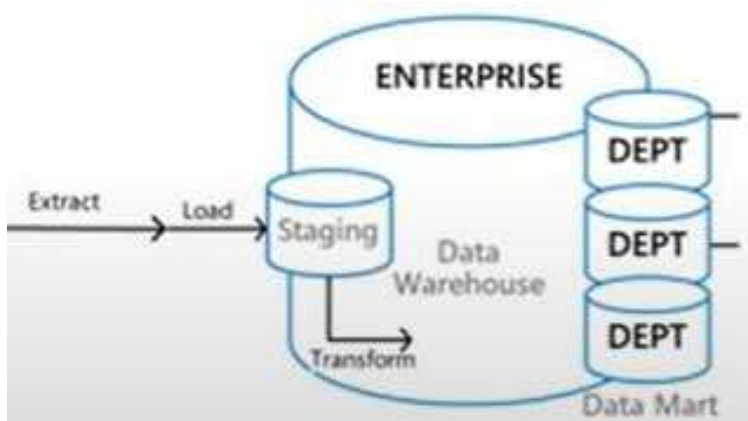




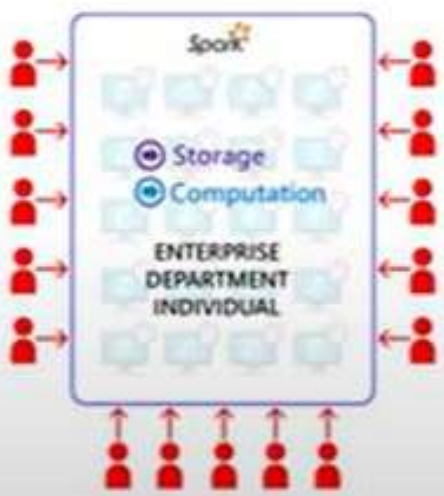
From datalake to mesh

The beginning

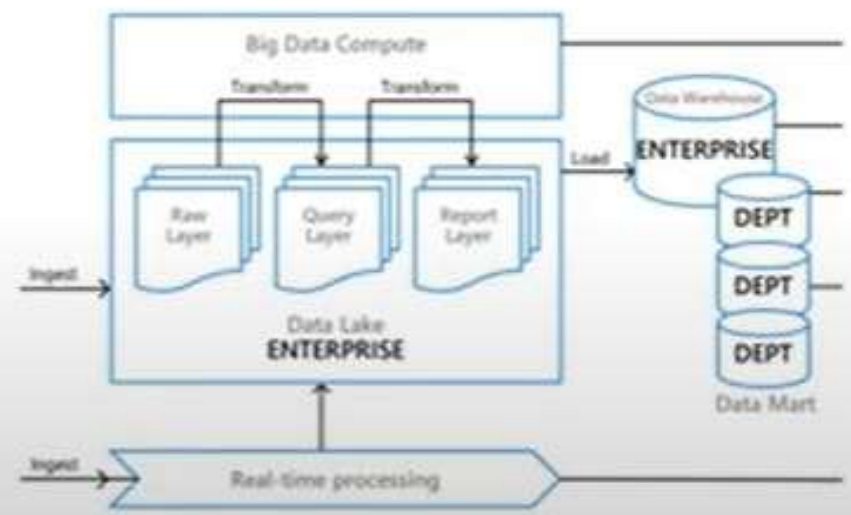
Late 1980s
Data Warehouse



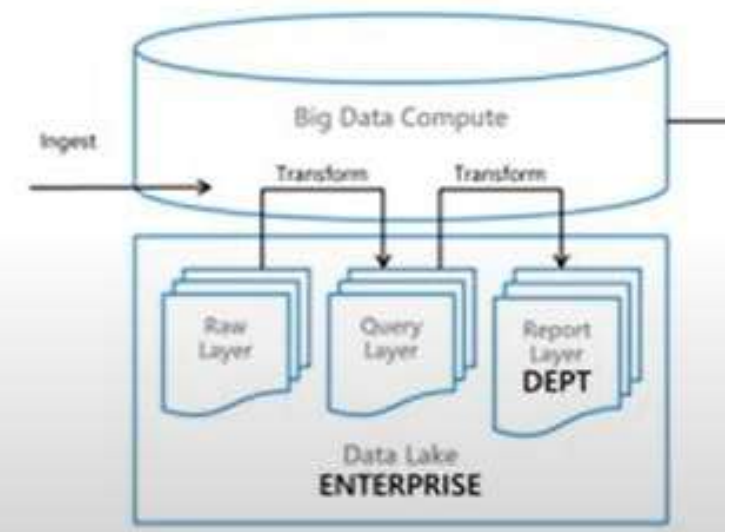
Late 2000s
Data Lake



Mid 2010s
Cloud Data Platform



Early 2020s
Data Lakehouse



Data Warehouse : structured storage that concentrate all the data of the enterprise, used for analytical purposes (reports, dashboards)
Issue : Hard to scaleup

Data Mart : structured storage oriented for a specific use (performing filtering, renormalization, etc)

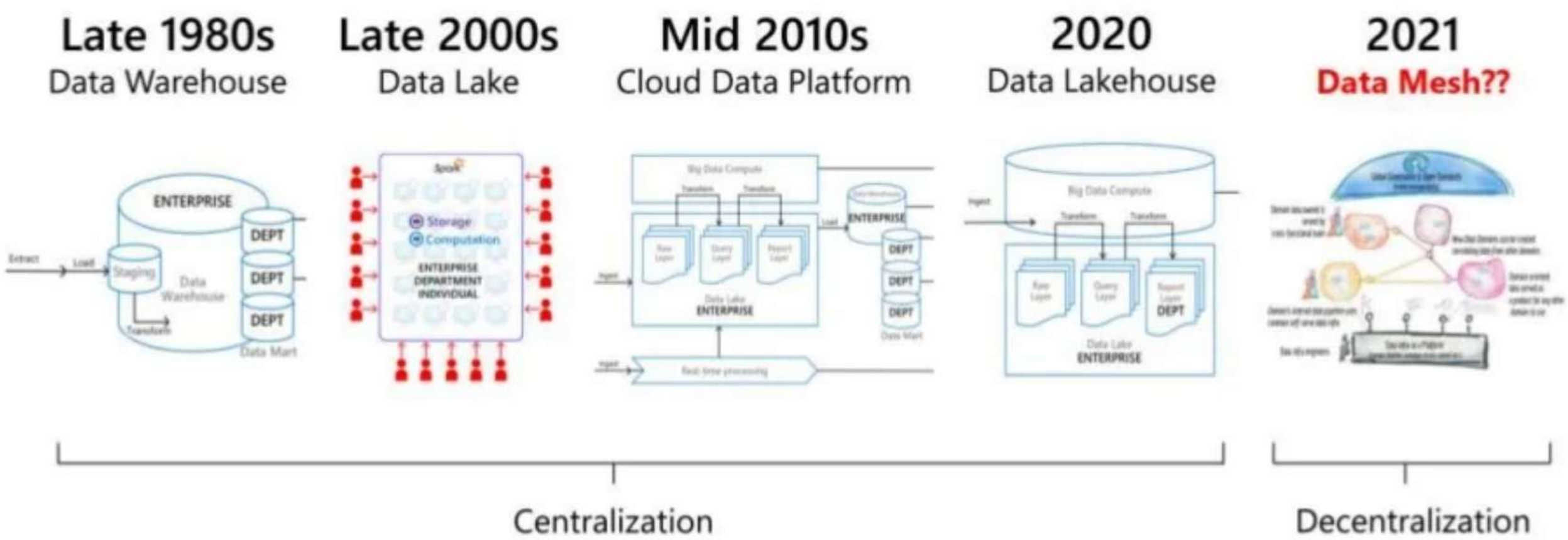
Data Lake : Huge amount of unstructured (and structured) storage with a scalable compute power and a centralized point for data analysis.
Issue : slow (batch oriented technologies), strong coupling between storage and compute

Data Platform : Cloud offering (easy access, agile, scalable) with a complete set of data services : from the enterprise data lake to different complementary products (streaming layer, data mart, etc) with a best of breed approach
Issue : requires strong technical skills to use or operate (especially onprem)

Data Lakehouse : scalable structured processing on top of heterogenous data in a lake
Issue : still a centralized approach with potential bottleneck on central data engineering team

From datalake to mesh

Mesh, the new paradigm



Data Mesh : instantiation of several data platform for each business domain of the enterprise that all are connected through the mesh (catalogue, norms, APIs, etc)
=> Full details in data governance course later

Quizz

What we've learn

Question				
In IaaS mode, should we manage the storage layer	Y	N		
In BMaaS mode, can we use our own Middelware	Y	N		
I need to focus on data application development, which service do I need ?	IaaS	CaaS	PaaS	SaaS
Hybride mode is when a client use mutliple cloud providers	Y	N		
A cloud native application is mainly composed of stateless processes	Y	N		
Data exploration is part of MLOps lifecycle	Y	N		
What is NOT an output of the "Presenting Results" phase during Build	Metrics on models quality	Evaluation data	A choosen model	Business vizualisation
DataOps is a technology to industrialize data	Y	N		
Are data warehouses an extinguished specie since data lakes ?	Y	N		
Is there API gateway in an MLOps Architecture ?	Y	N		

Quizz

What we've learn

Question				
In IaaS mode, should we manage the storage layer	Y	N		
In BMaaS mode, can we use our own Middelware	Y	N		
I need to focus on data application development, which service do I need ?	IaaS	CaaS	PaaS	SaaS
Hybride mode is when a client use mutliple cloud providers	Y	N		
A cloud native application is mainly composed of stateless processes	Y	N		
Data exploration is part of MLOps lifecycle	Y	N		
What is NOT an output of the "Presenting Results" phase during Build	Metrics on models quality	Evaluation data	A choosen model	Business vizualisation
DataOps is a technology to industrialize data	Y	N		
Are data warehouses an extinguished specie since data lakes ?	Y	N		
Is there API gateway in an MLOps Architecture ?	Y	N		

In Iaas mode we manage OS layer
With PaaS mode we develop application and data layer
When a client use multiple cloud providers, it's Multicloud
A cloud native app is based on stales processes, see point 6 of manifesto
Evaluation data is an input of "Presenting Results" phase, not an output
DataOps is not a technology, it's a framework and a management practice
Data warehouse is still the structured part on top of datalake for analytics
API Gateway is a component of big data architecture, not MLOps architecture

In Practice

Lab Content

https://github.com/A709509/aiengineerPolytech

- Discover
  - **Notebook** on KubeFlow
    - Exo1: explo/viz
      - Getting open data from public api and push it to s3
      - Quick analysis with python
    - Exo2: dwh/viz
      - Push data to **CH** table + **postgres** table
      - Visualization with **superset**
    - Exo3: stream
      - Push un event to a **kafka** topic
      - Event visualization with **akhq**
      - Read it form a consumer
      - Bonus : use a kafka engine in **CH** and see event within **superset**