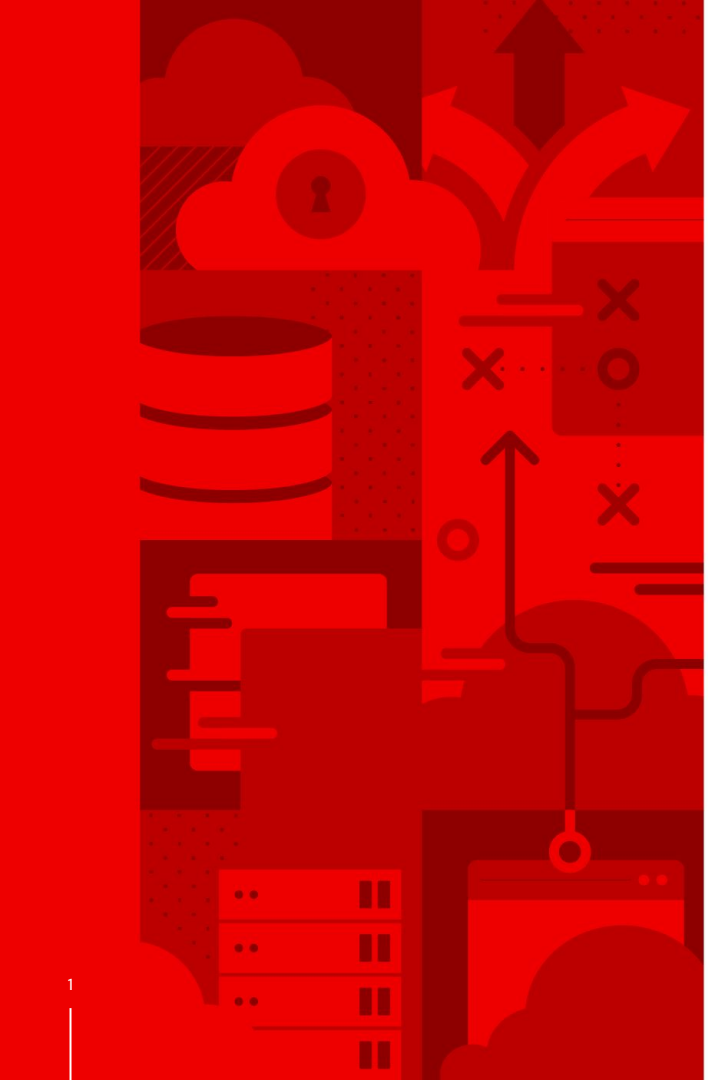


Embedding AI

Don’t burn your fingers in the heat of the AI revolution

Camille Nigon
Solutions Architect

Maarten Vandeperre
Solutions Architect

Red Hat


OOOH AI

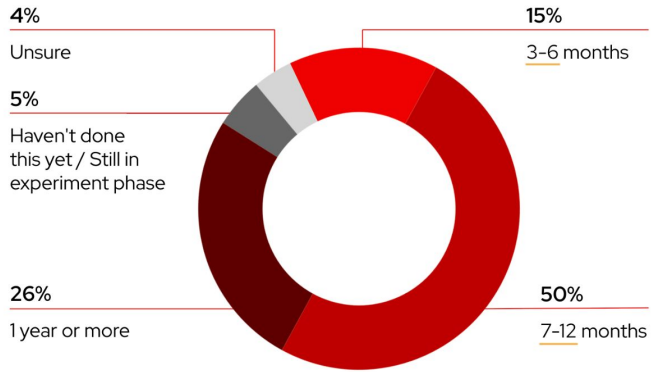
Gartner Predicts 30% of Generative AI Projects Will Be Abandoned After Proof of Concept By End of 2025

SYDNEY, Australia, July 29, 2024

Operationalizing AI is still a challenging process

What is the average AI/ML timeline from idea to operationalizing the model?

Half of respondents (50%) say their average AI/ML timeline from idea to operationalizing the model is 7-12 months.



Source: Gartner Peer Insights, Open Source AI for Enterprise survey, 2023

https://www.informatica.com/blogs/the-surprising-reason-most-ai-projects-fail-and-how-to-avoid-it-at-your-enterprise.htm
https://www.gartner.com/en/newsroom/press-releases/2024-07-29-gartner-predicts-30-percent-of-generative-ai-projects-will-be-abandoned-after-proof-of-concept-by-end-of-2025

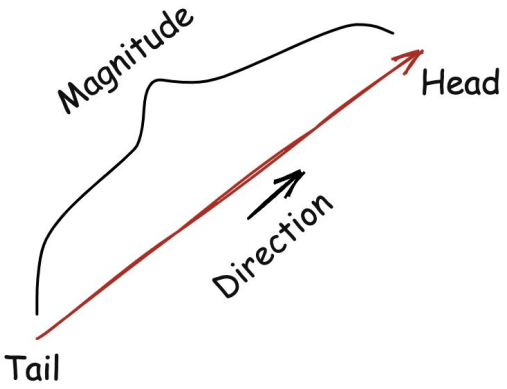
What Happened?!

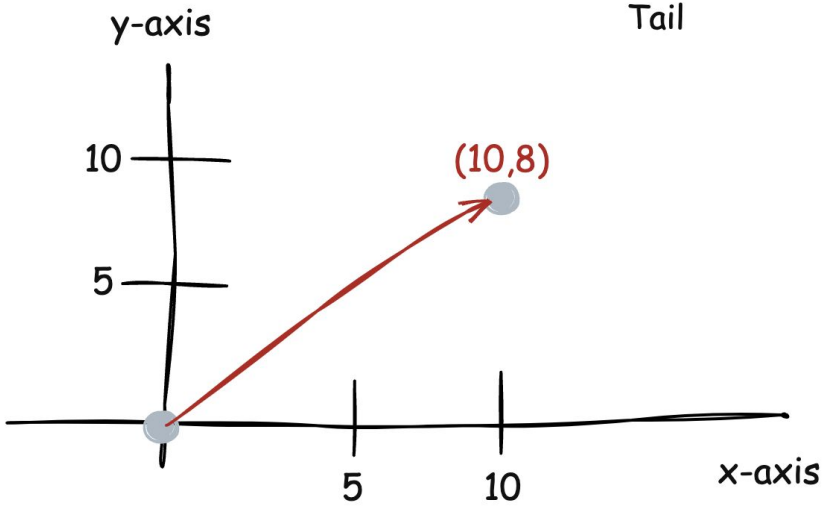
What happened?!

- Unrealistic targets
- AI not seen as implementation detail ⇒ No need to find things it should fix
- Treated as a toy ⇒ We'll need to bring it to production
- How can we fix this?
  - Concepts
  - Challenges & proposals

What happened?!

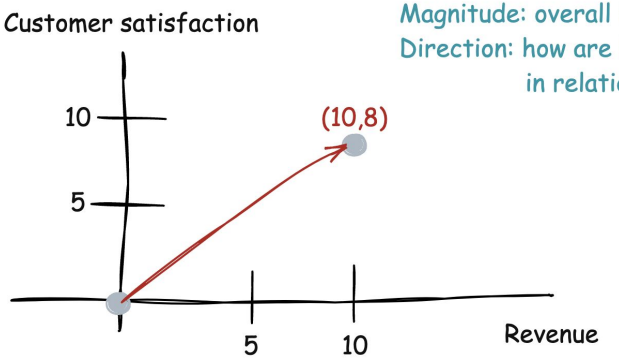
- Unrealistic targets
- Treated as a toy ⇒ we'll need to bring it to production
- How?
  - **⇒ Concepts**
  - Challenges & proposals



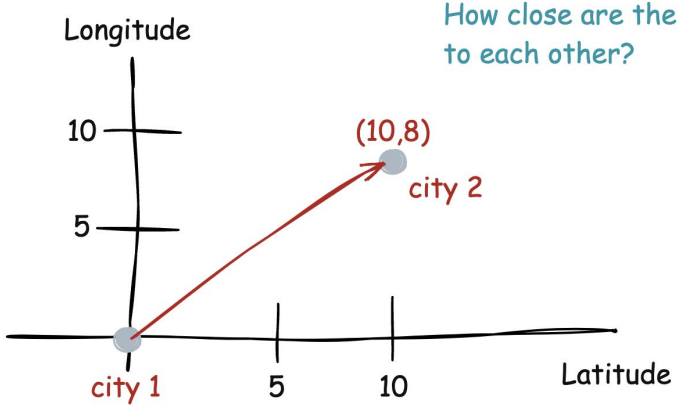


Vectors

- A list of numbers that represent complex data in a simplified, numerical form.
- can exist in two-dimensional (2D) space, three-dimensional (3D) space, or even higher dimensions, depending on the context.



Magnitude: overall impact
Direction: how are both factors evolving in relation to each other



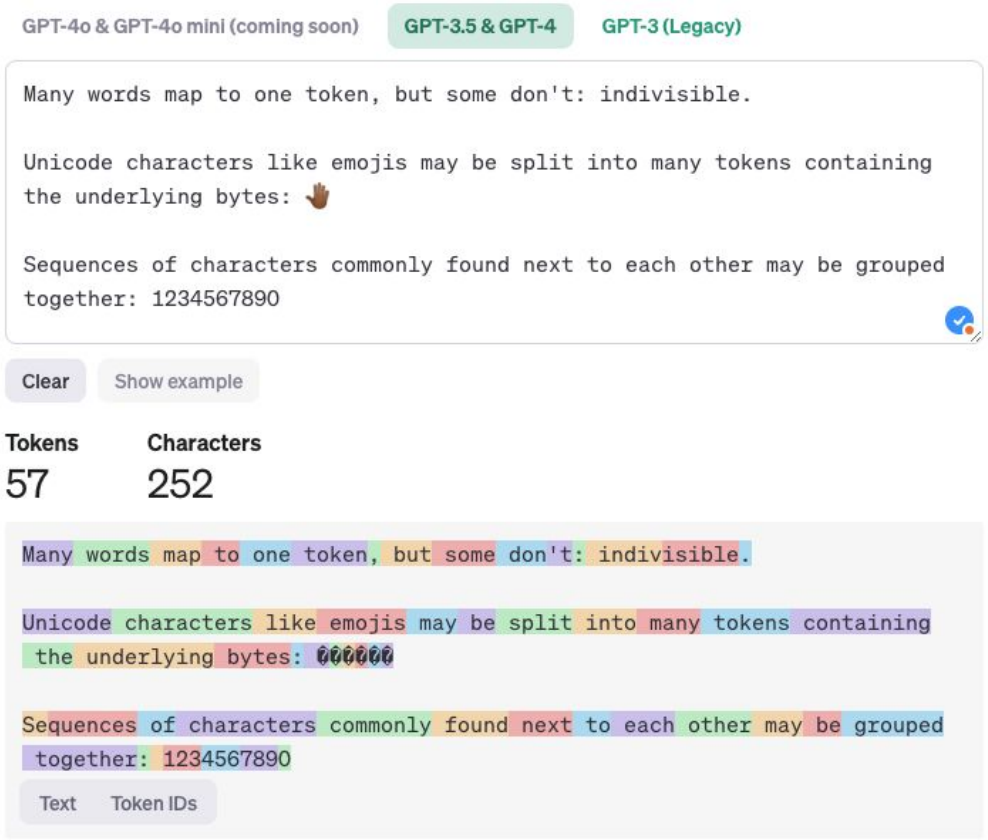
How close are the cities to each other?

Vectors

- Practical examples.
- AI example:

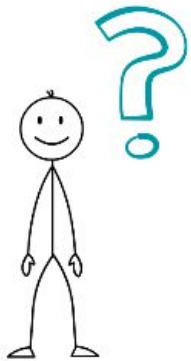
  How close are words to each other.

  !! N-dimensional space.



Tokens

- A basic unit of text that a model works with. Tokens can be entire words, sub-words, or even characters.
- Represent chunks of the original data (e.g., text) after it's been broken down. For example, the sentence “I love AI” could be tokenized as ["I", "love", "AI"].
- AI: text ⇒ tokens ⇒ vectors.



Problem statement:

We need to identify/search for similarities
in between words or sentences, similarity search.

- **Dog**: "A loyal, domesticated animal known for companionship."
- **Cat**: "A small, domesticated animal that is often independent."
- **Wolf**: "A wild animal related to dogs, living in packs."

- **Dog**: ["A", "loyal", "domesticated", "animal", "known", "for", "companionship"]
- **Cat**: ["A", "small", "domesticated", "animal", "that", "is", "often", "independent"]
- **Wolf**: ["A", "wild", "animal", "related", "to", "dogs", "living", "in", "packs"]

- "loyal": [0.9, 0.2, 0.3]
- "domesticated": [0.7, 0.8, 0.6]
- "animal": [0.6, 0.7, 0.7]
- "wild": [0.4, 0.9, 0.5]
- "companionship": [0.9, 0.1, 0.5]

- **Dog vector**: [0.75, 0.5, 0.5]
- **Cat vector**: [0.65, 0.6, 0.55]
- **Wolf vector**: [0.55, 0.8, 0.55]

Animal	Vector
Dog	[0.75, 0.5, 0.5]
Cat	[0.65, 0.6, 0.55]
Wolf	[0.55, 0.8, 0.55]

"domesticated, wild, and related to dogs"

- **Query vector**: [0.6, 0.65, 0.55]

- **Dog vector vs. Query vector**: 0.98 (very close)
- **Cat vector vs. Query vector**: 0.75 (not so close)
- **Wolf vector vs. Query vector**: 0.95 (close)

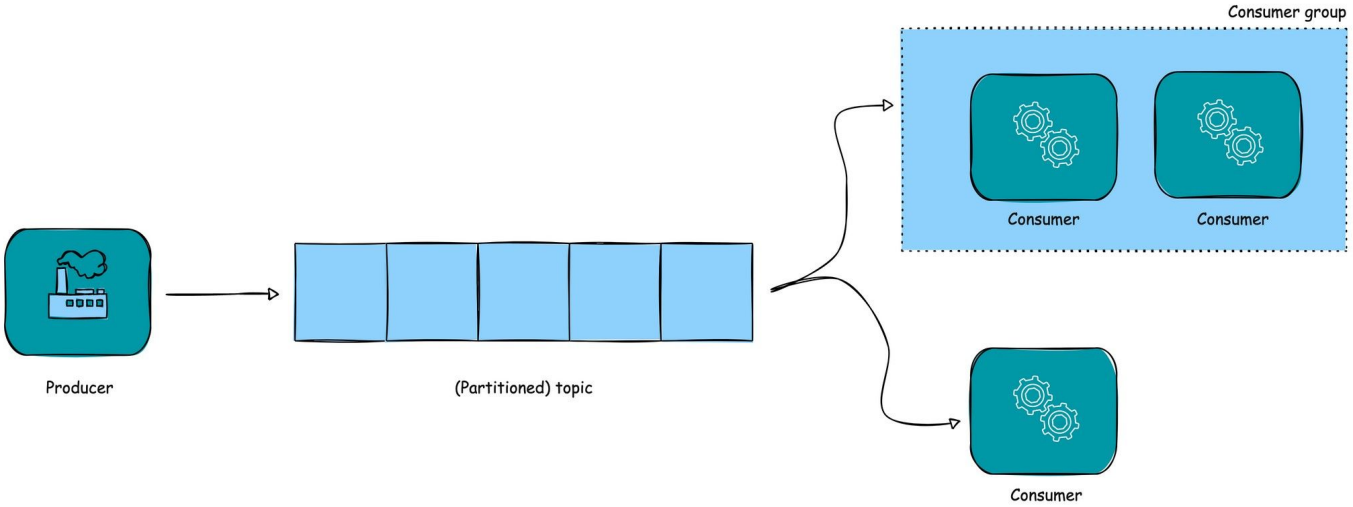
The database returns **Dog** as the most relevant animal, but **Wolf** is also quite similar.

Biased/racist AI
Model bias

- Racial bias.
- Education bias.
- Social class bias.
- Age bias.
- Gender bias.
- ...
- Sometimes not bad to have ... bias (!!! Listen to explanation first).

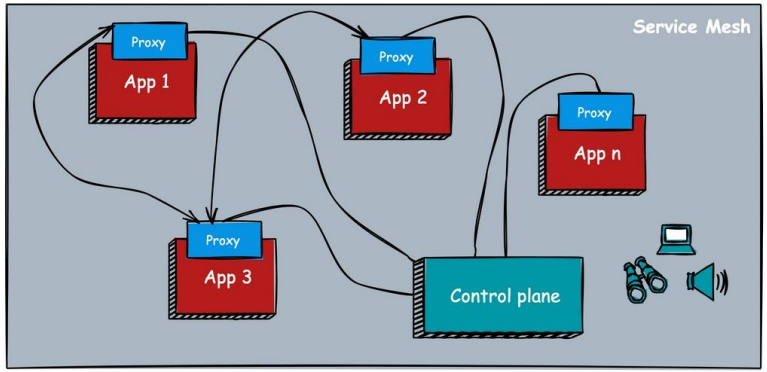
Model drift

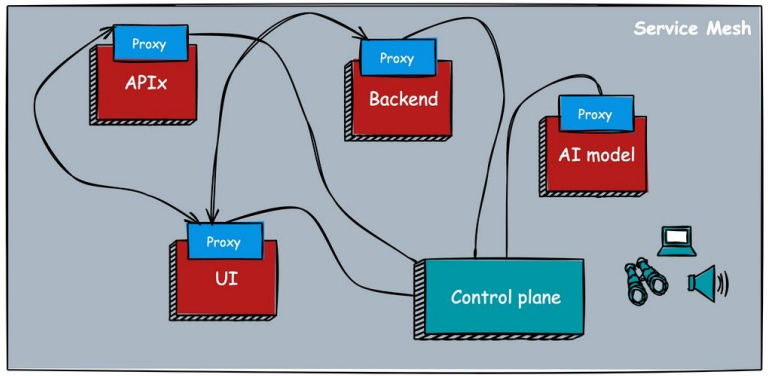
- E.g., husky or wolf.
- Color scale.
- Orientation.
- Context
- ...



Kafka

- Producer
- Topics (message boxes)
- Consumer groups
- Consumers



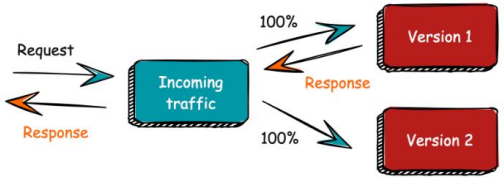


Communication: App x can communicate with App y within the service mesh

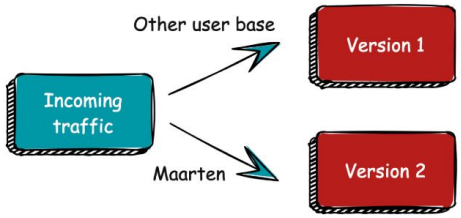
Service mesh

- Advanced routing
- mTLS
- Observability & monitoring

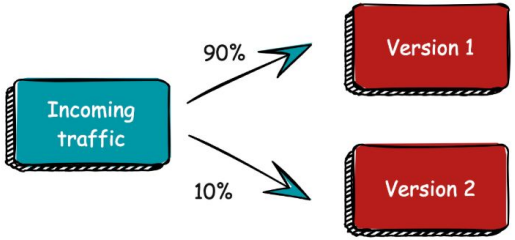
Mitigate deployment risks in a distributed application environment



Validation - Mirroring



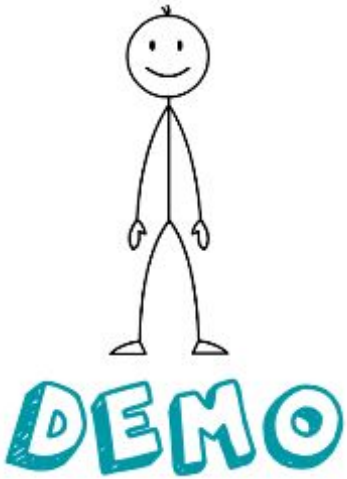
Testing - Canary releases



Rolling out - Blue/green deployments

Service mesh

- Deployment strategies
- Mirrorring
- Canary
- Blue/green


DEMO

What happened?!

- Unrealistic targets
- Treated as a toy ⇒ we'll need to bring it to production
- How?
  - **⇒ Concepts**
    - Vectors
    - Tokens
    - Vector database
    - Bias & drift
    - Kafka
    - Service Mesh
  - Challenges & proposals

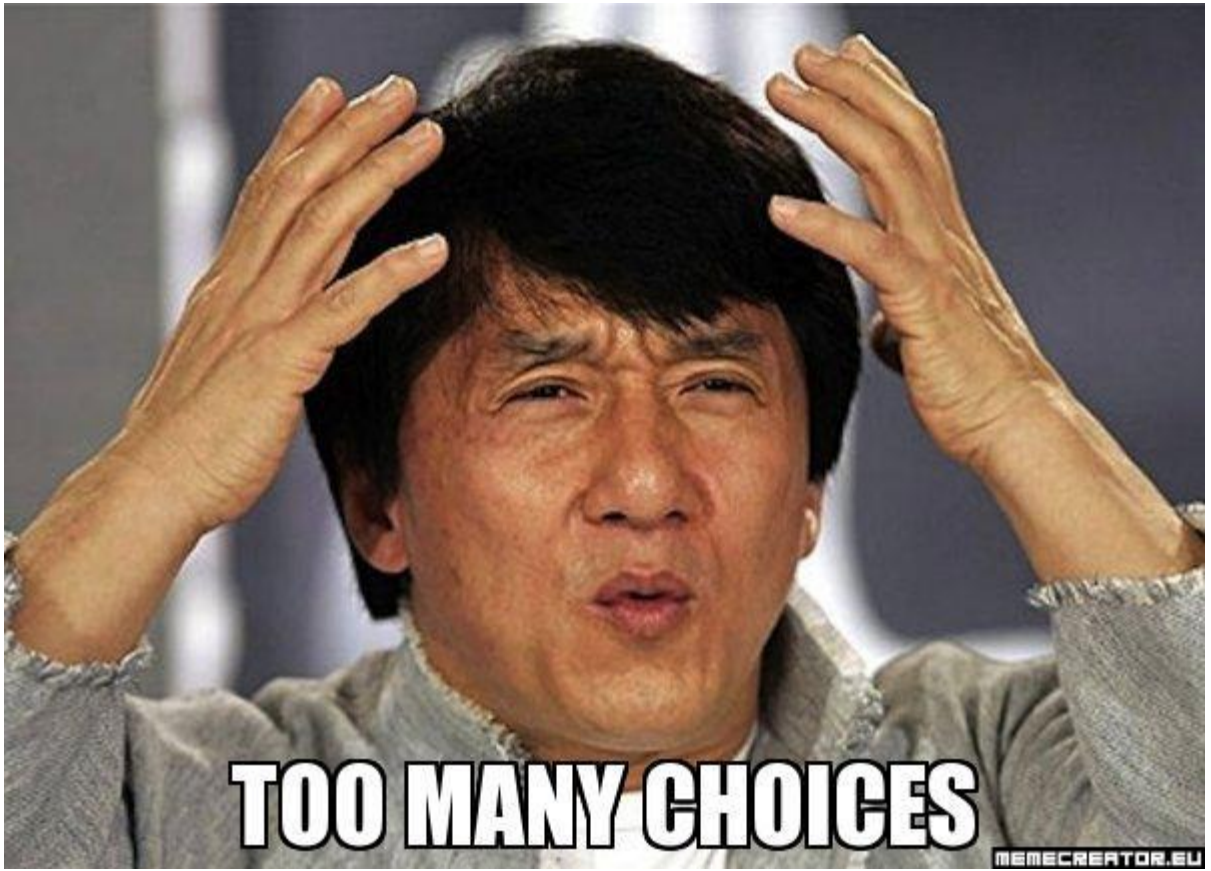
What happened?!

- Unrealistic targets
- Treated as a toy ⇒ we'll need to bring it to production
- How?
  - Concepts
  - **⇒ Challenges & proposals**

Challenge

Tools

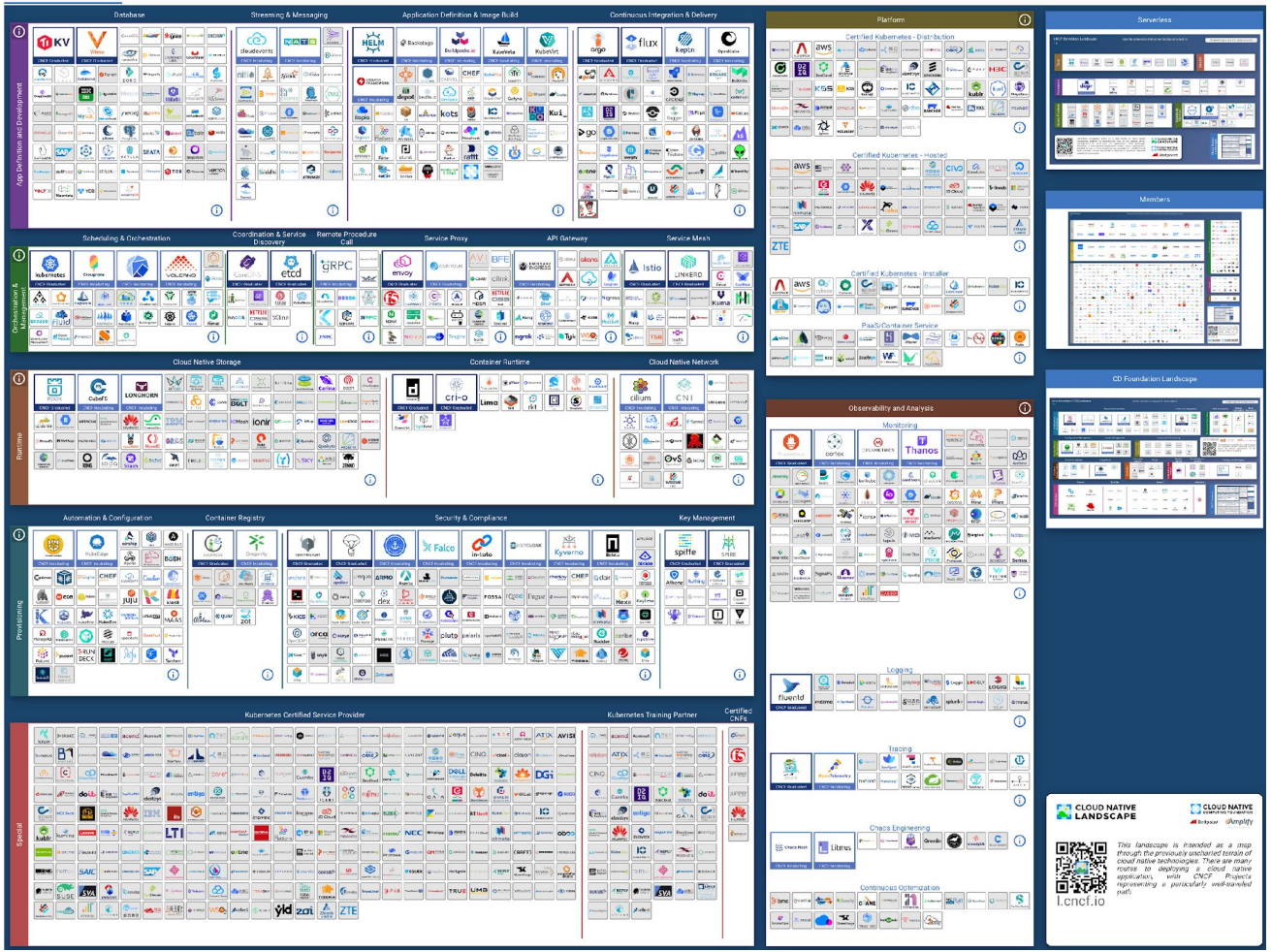
Tools: Too much choice



The Cloud-Native Grocery Store

Where's Waldo?

or Wally?

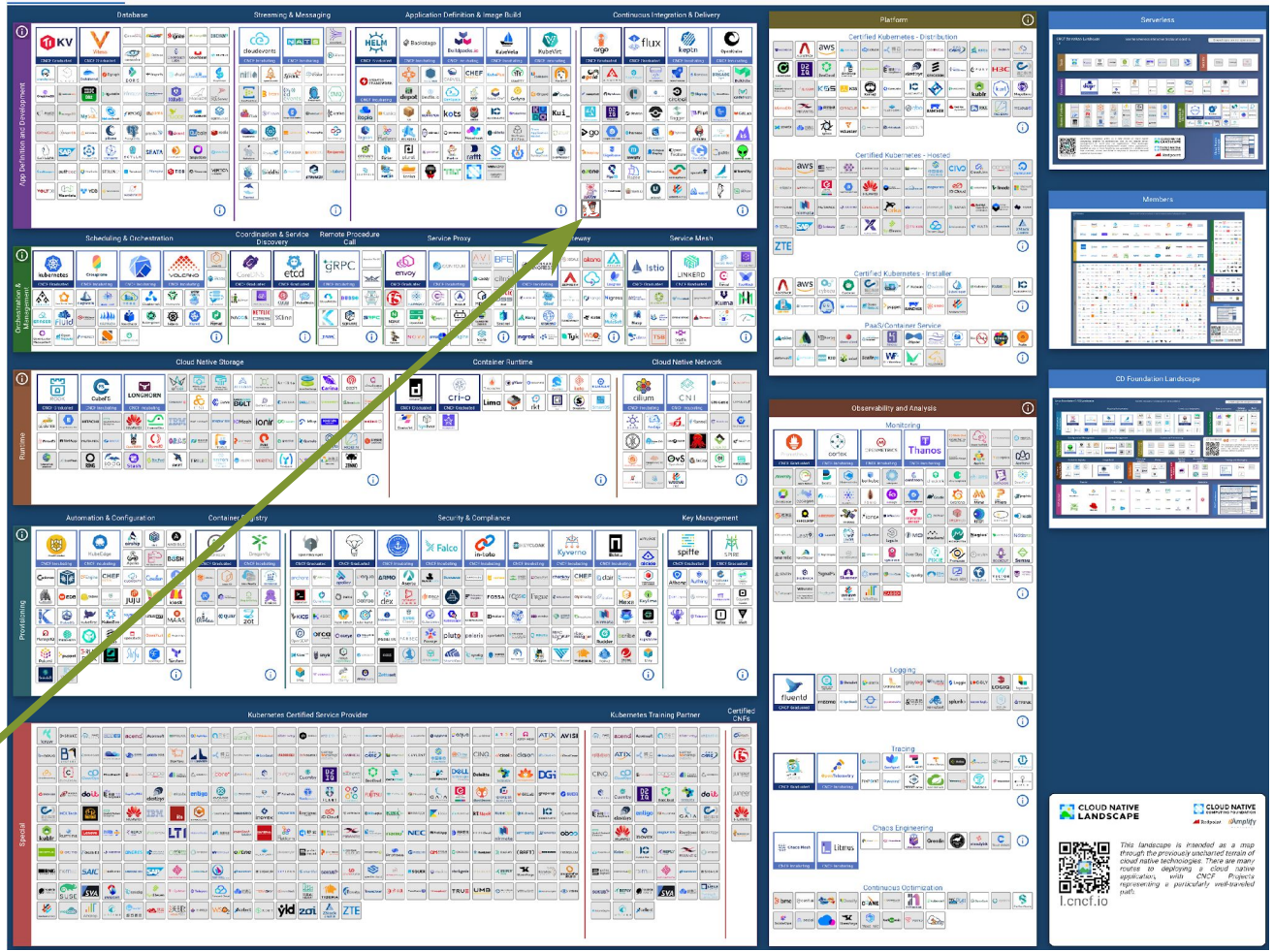


Source: https://landscape.cncf.io/

The Cloud-Native Grocery Store

Where's Waldo?

or Wally?



Source: https://landscape.cncf.io/

Upstream projects

RAY

CodeFlare

ModelMesh

Kubeflow

PyTorch

TensorFlow

KServe

jupyter

TrustyAI

Community projects

OPEN DATA HUB

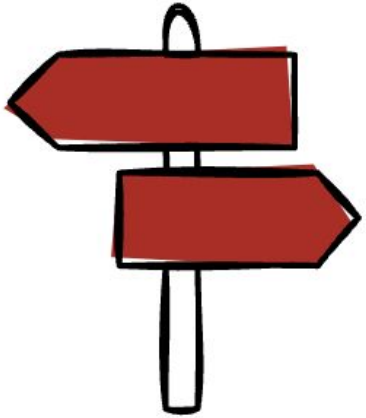
AI Platform powered by Open Source

Product

Red Hat OpenShift AI

Challenge

Big bang: no step-by-step/agile approach



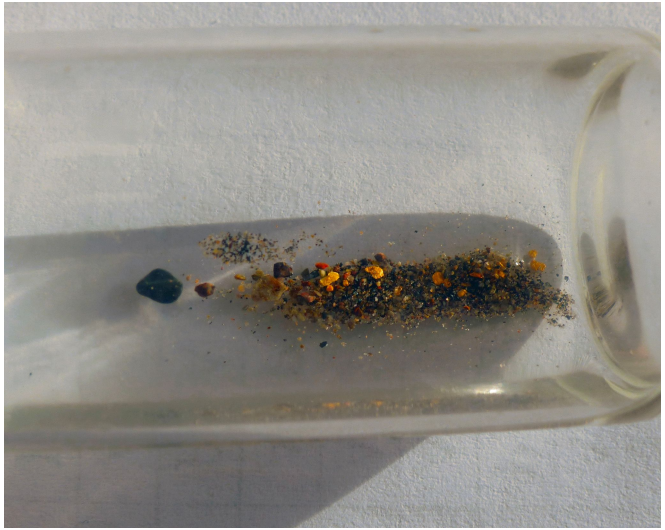
Let's take a little detour
and talk about gem mining.

Step 1:
Filter rocks from water

- Huge amounts of water passing by.
- Huge amounts of rocks passing by.
- Only subset of rocks collected.
- Can be collected by anyone.

Step 2:
Filter out the rocks with crystals inside

- Only subset of rocks contains crystals.
- Checked by a gemologist/geologist.



Step 3:

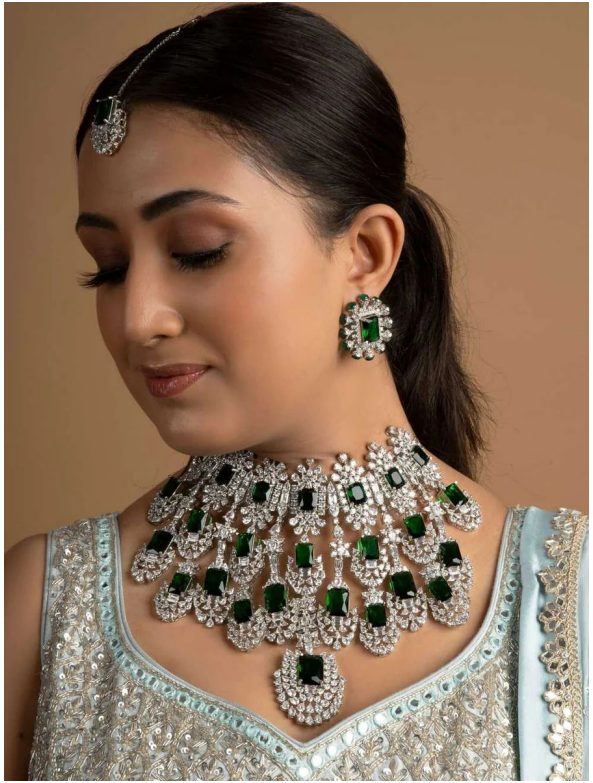
Extract crystals from the rocks

- From rocks with crystals.
- Extracted by gemologist/geologist.
- Resulting in polished gems.

Step 4:

Create jewelry

- Done by a jeweler.
- Can be sold to anyone.
- Anyone can wear it.



Step 5:

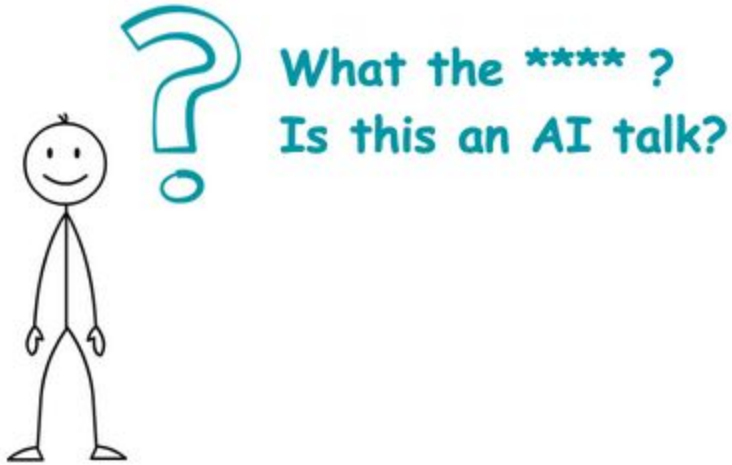
Wear the jewels

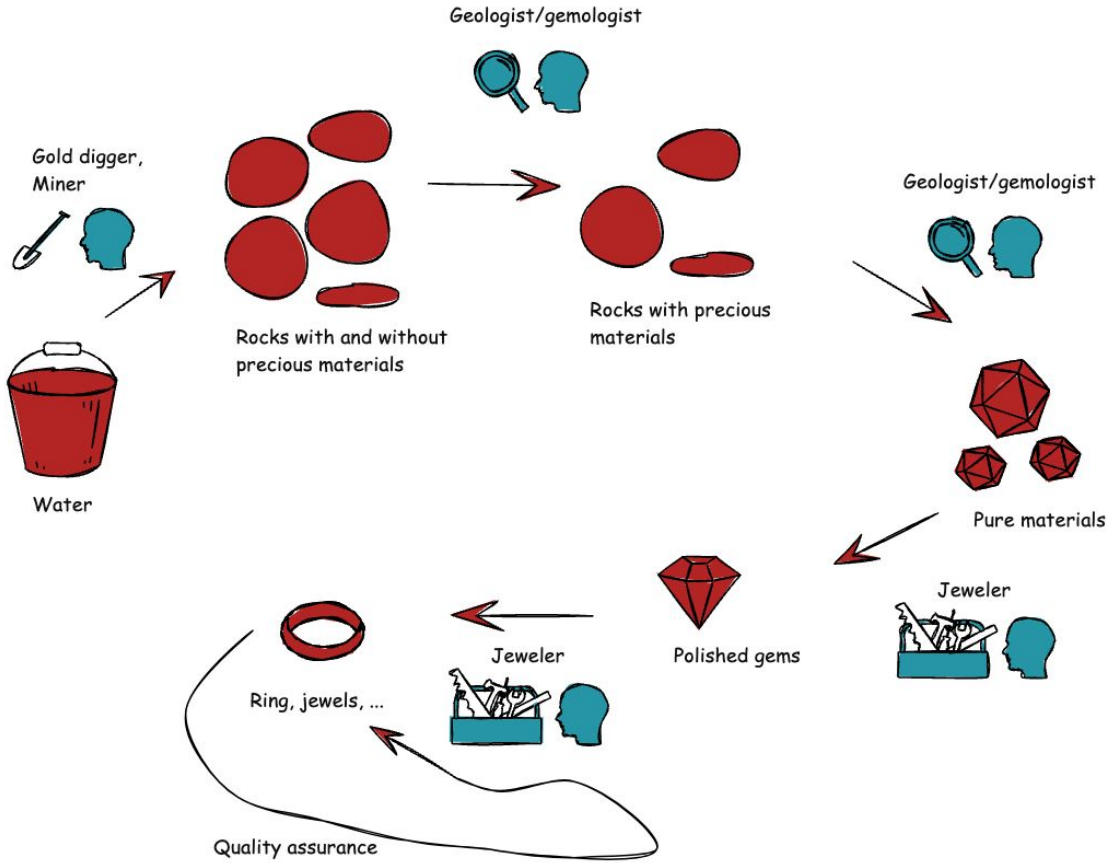
- Anyone can wear it.

Sources:

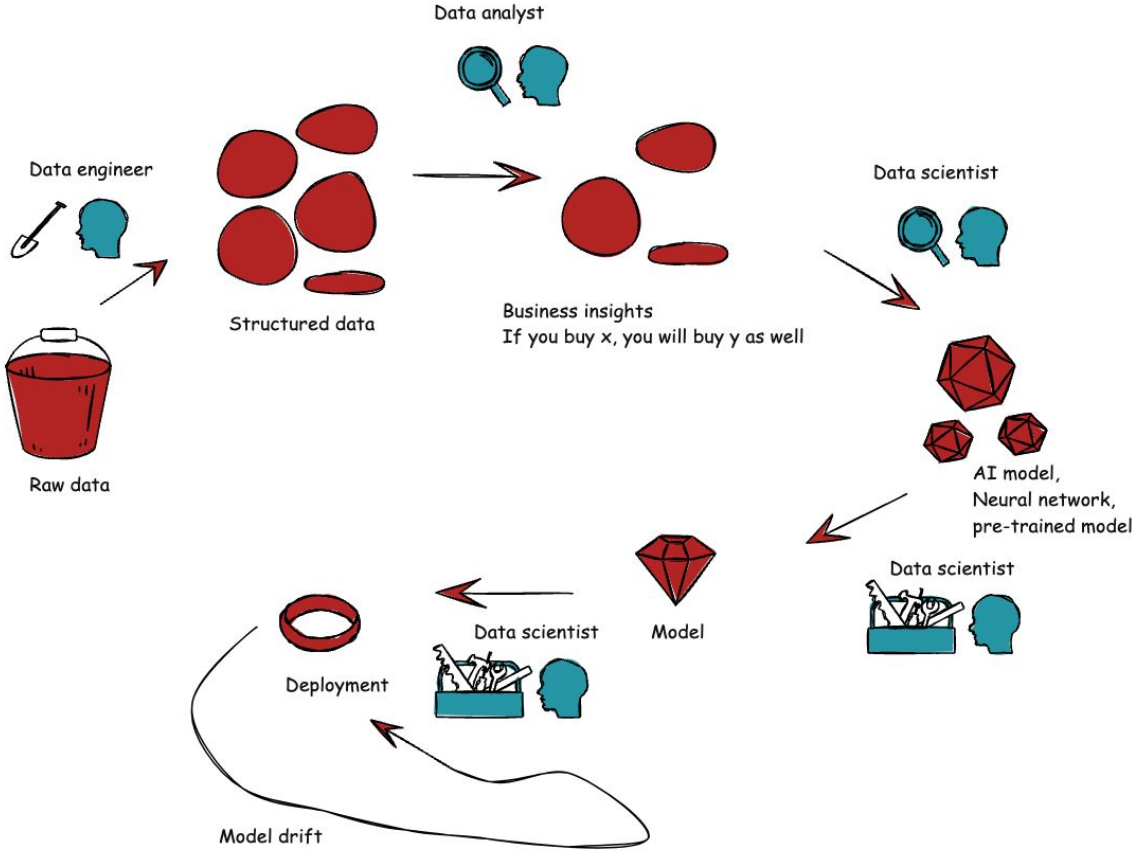
https://queensjewels.in/cdn/shop/files/QJ2982-transformed.jpg?v=1688986573&width=900

https://news.bitcoin.com/dutch-national-bank-says-gold-can-re-start-economy-in-case-of-total-collapse/

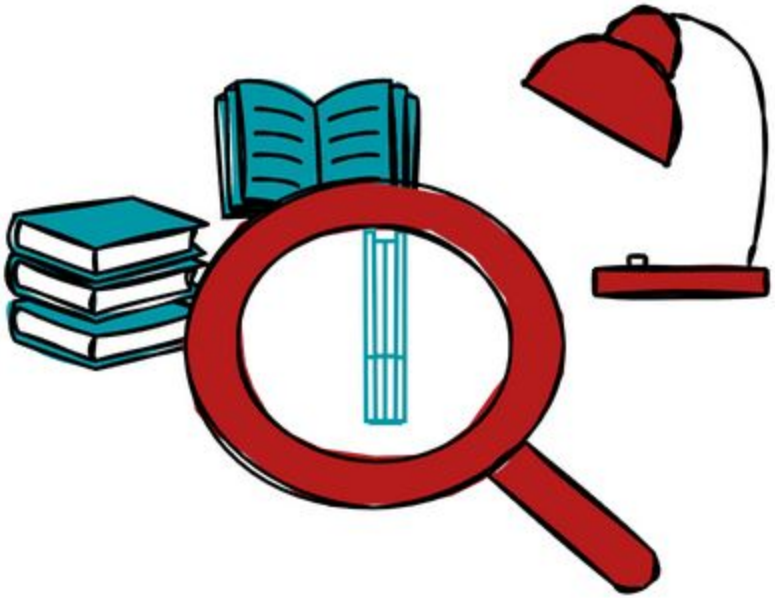

What the **** ?
Is this an AI talk?


Geologist/gemologist
Gold digger,
Miner
Rocks with and without
precious materials
Rocks with precious
materials
Geologist/gemologist
Water
Pure materials
Jeweler
Polished gems
Jeweler
Ring, jewels, ...
Quality assurance

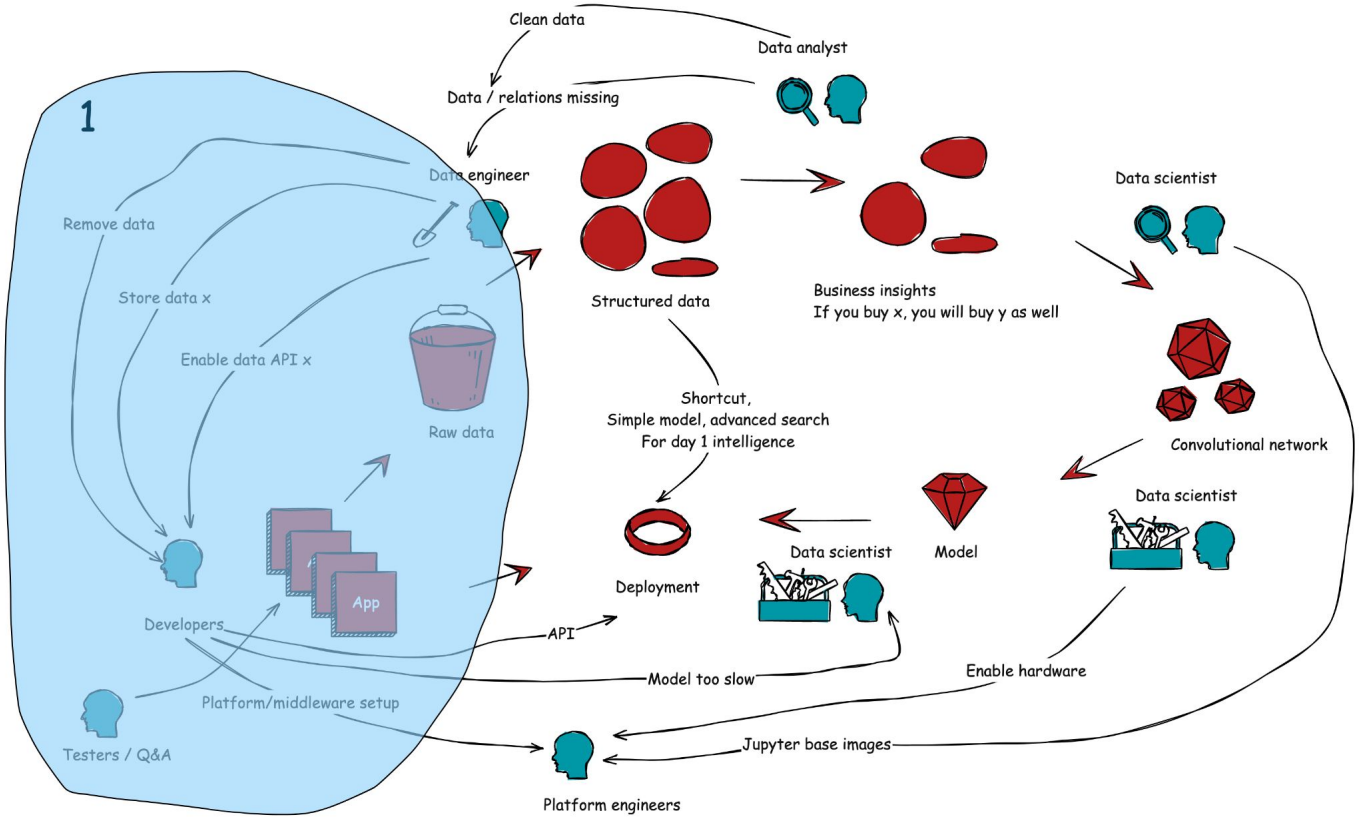
Gem mining


Data analyst
Data engineer
Structured data
Business insights
If you buy x, you will buy y as well
Data scientist
Raw data
AI model,
Neural network,
pre-trained model
Data scientist
Model
Data scientist
Deployment
Model drift

ML process



ROAD TOWARDS AI

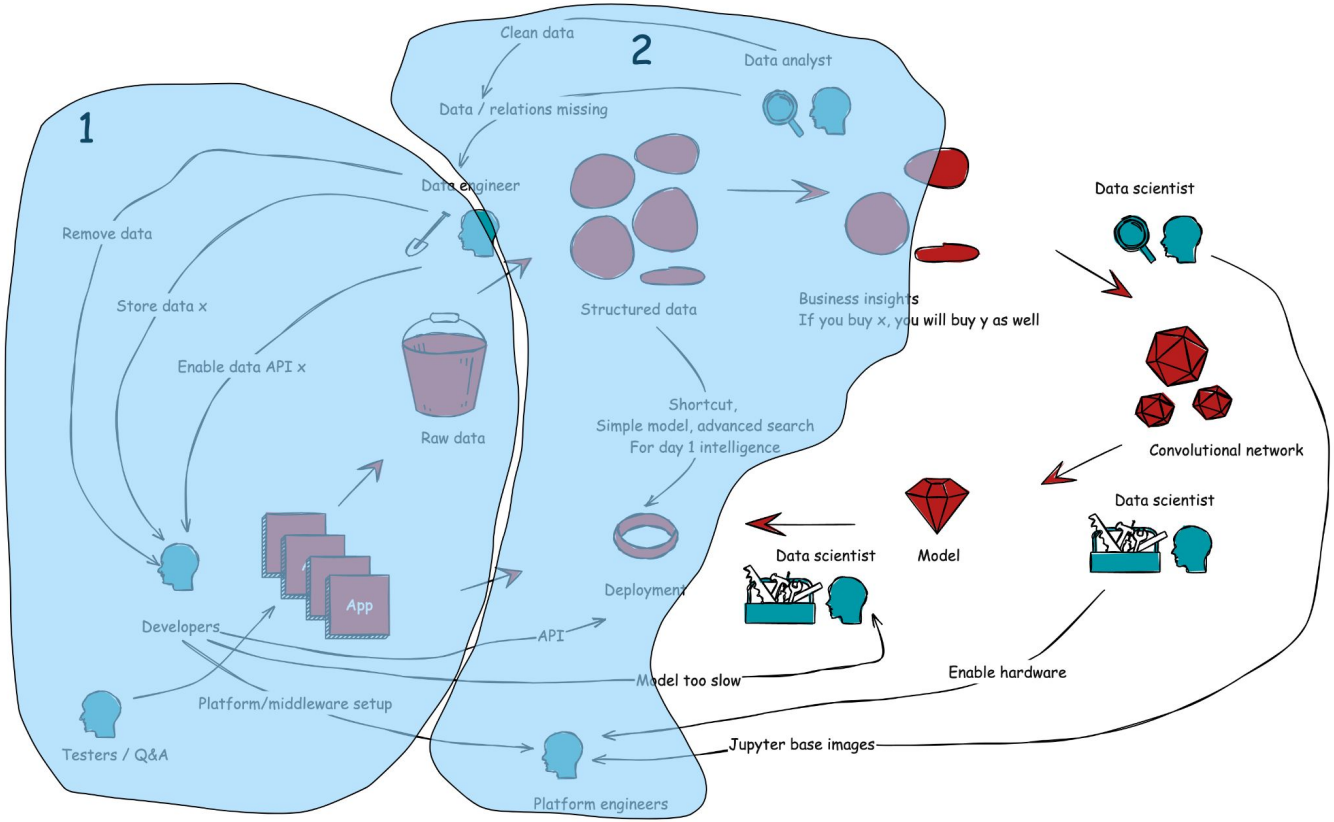


Road towards AI

- App platform.
- OpenShift Container Platform.
- Application Foundations.
- (Data Foundations).
- (RHEL).
- (Ansible).

⇒ App platform

⇒ +- Data platform



Road towards AI

The step in between.

Day 1 intelligence.

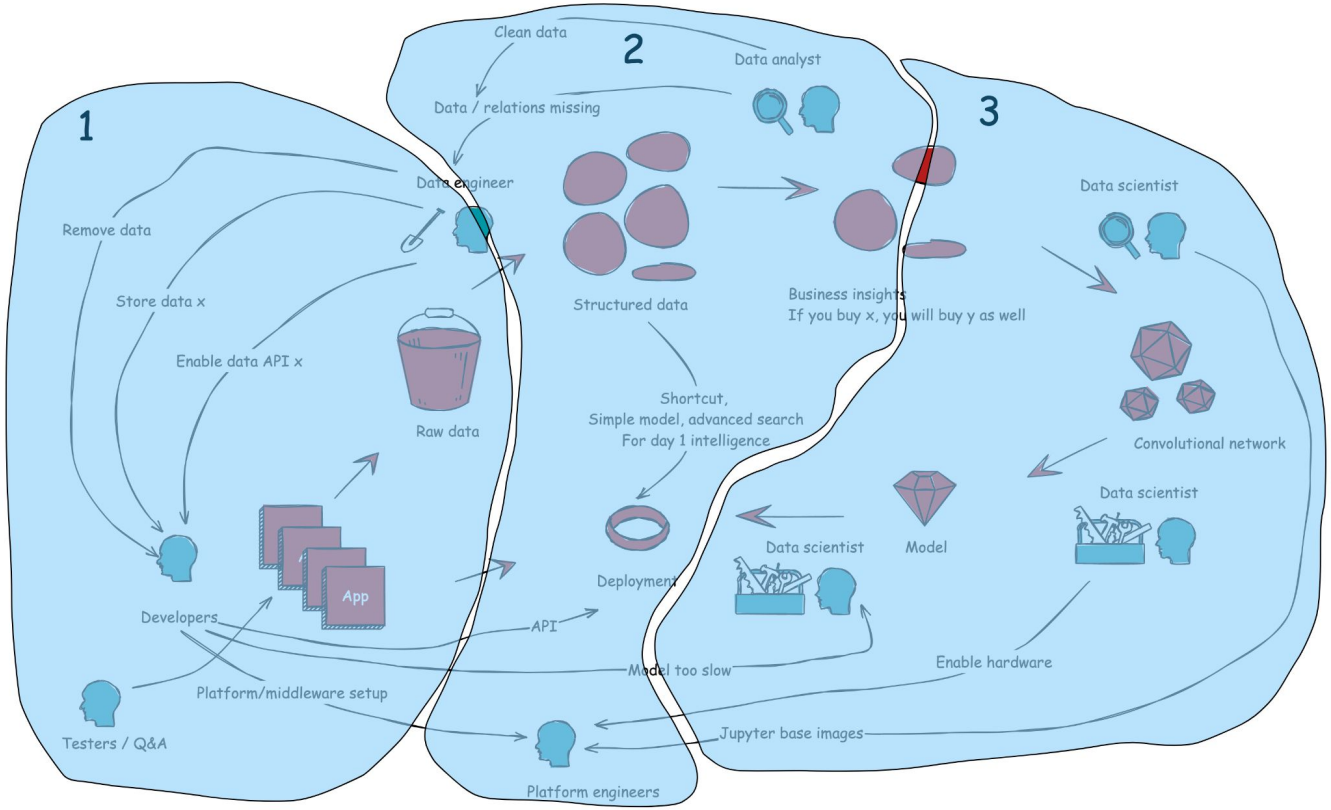
- **Data platform.**
- **Application Foundations.**

⇒ **App platform.**

⇒ **Data platform.**

⇒ **AI.**

- Graph databases.
- Spark.
- …



Road towards AI

- OpenShift AI.

⇒ App platform.

⇒ Data platform.

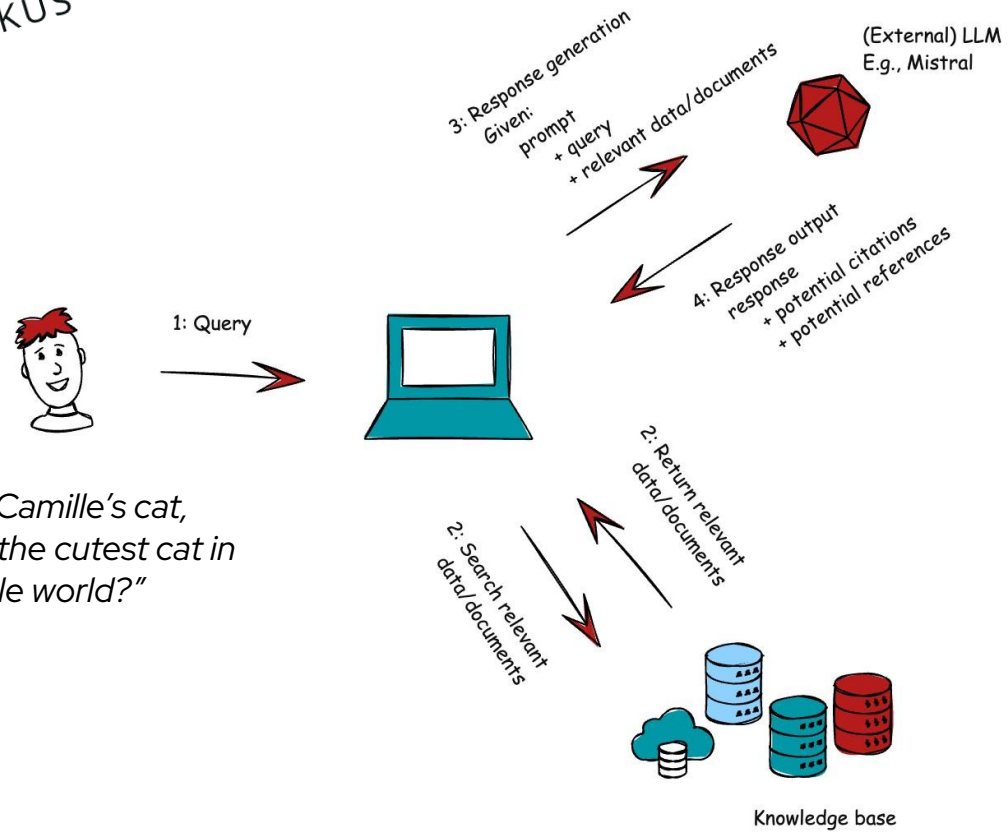
⇒ AI.

⇒ ML.

Challenge

Model enhancements

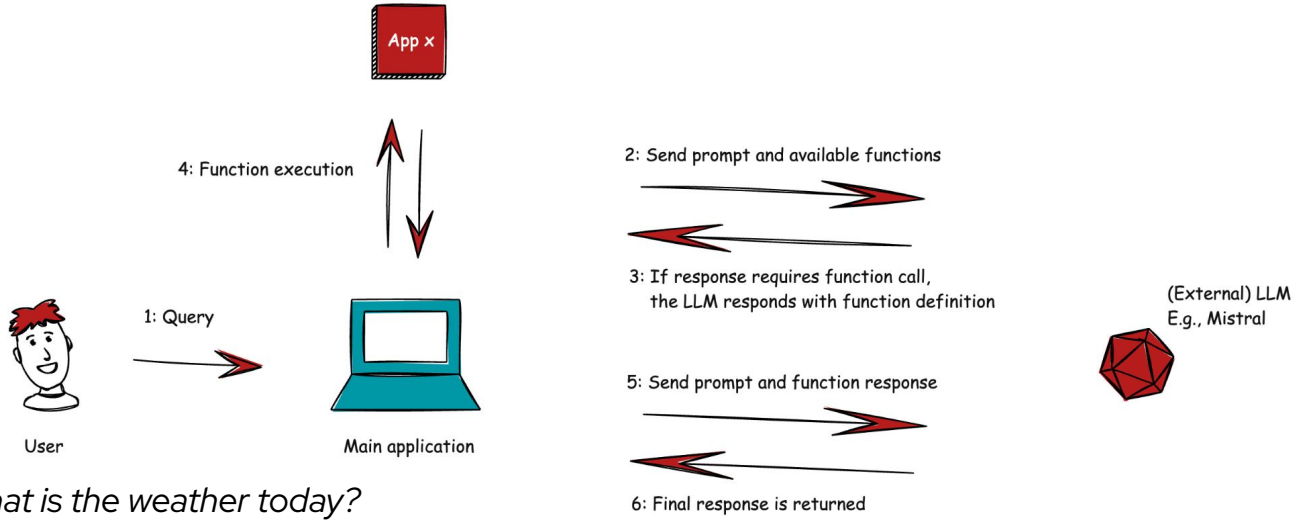
QUARKUS



"Why is Camille's cat, Mitsou, the cutest cat in the whole world?"

RAG

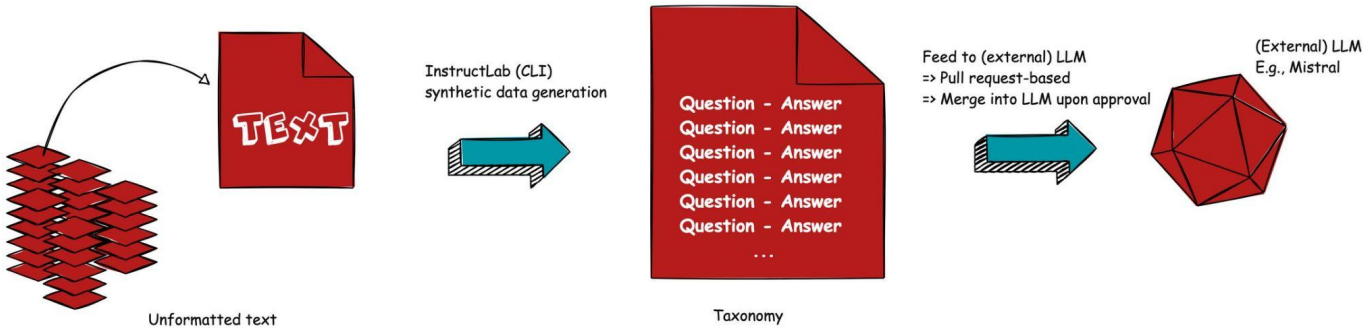
- https://redhat-developer-demos.github.io/quarkus-tutorial/quarkus-tutorial/17_prompts.html
- Protect data.
- No knowledge, use case, time, ... to fine-tune, train a model yourself.
- Source references.
- Less scalable, bigger models.
- Quarkus: next-gen java



What is the weather today?

Function calling

- https://redhat-developer-demos.github.io/quarkus-tutorial/quarkus-tutorial/17_prompts.html
- External call(-out).
- External APIs.
- Get factual data.



InstructLab

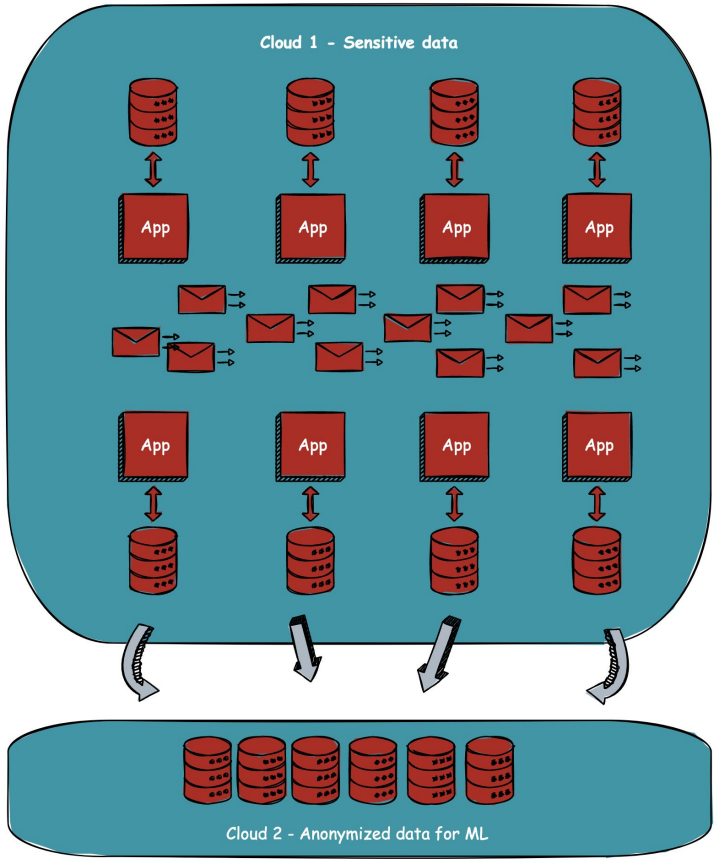
- https://github.com/instructlab
- Open collaboration.
- Open way of contributing.
- Less data science knowledge.

Maarten Vandeperre
Maître des charmes Java appliqués et des rituels d'appel de fonctions

Camille Nigon
Professeure en intelligence magique

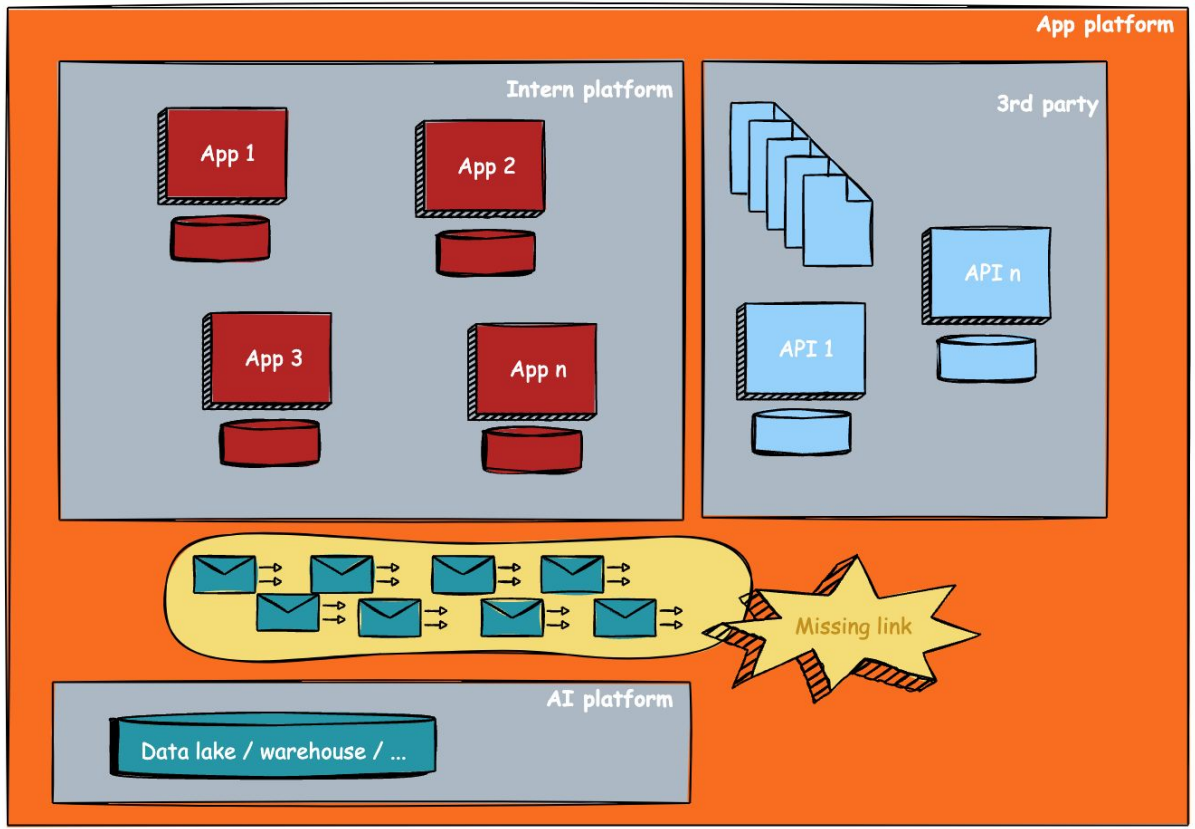
Challenge

Sovereignty + lack of dedicated environments



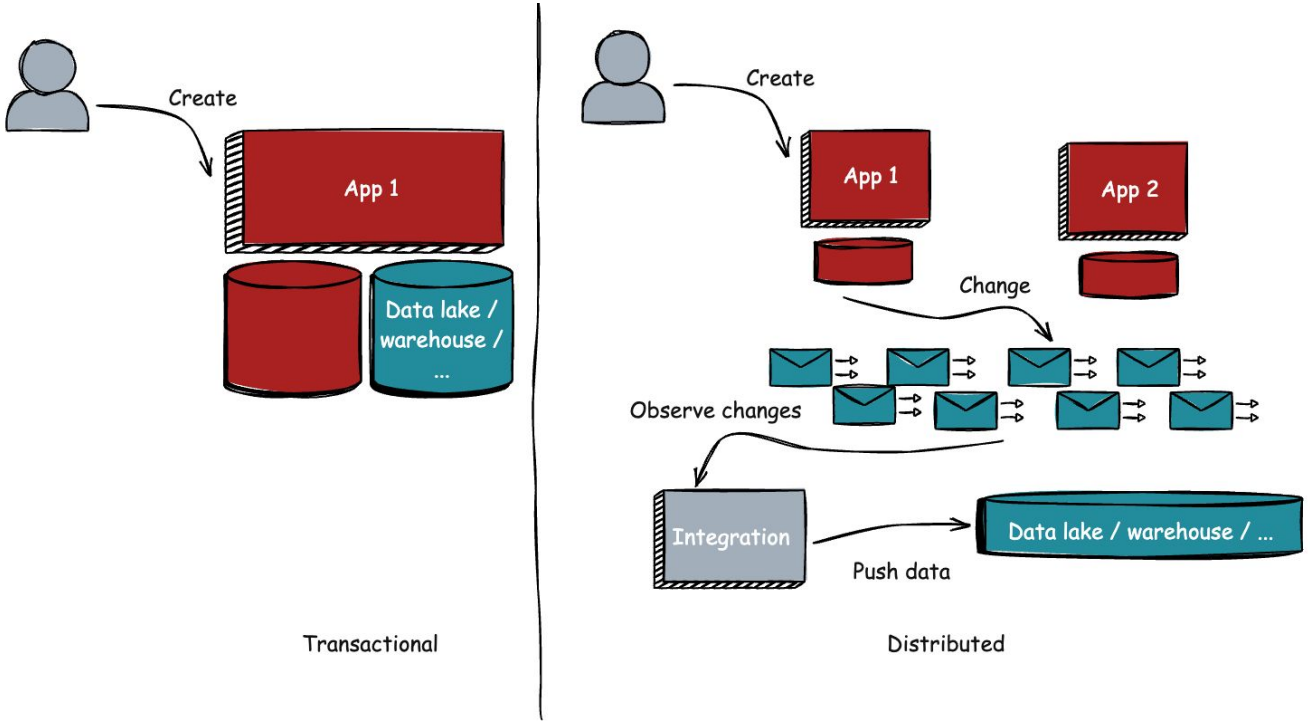
Hybrid cloud

- Where do you want to run the models?
- Where do you want to train the models?
- IoT?
- Where is your application platform?
- Where is your data platform?
- Where is your data?
- What if another “Broadcom” happens?



Data sources

- App platforms.
- External APIs.
  (E.g., weather data).
- Files.
  (E.g., good old Excel or CSV).



Data transformations

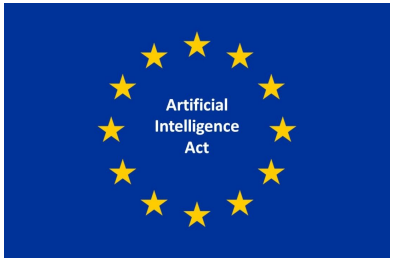
-

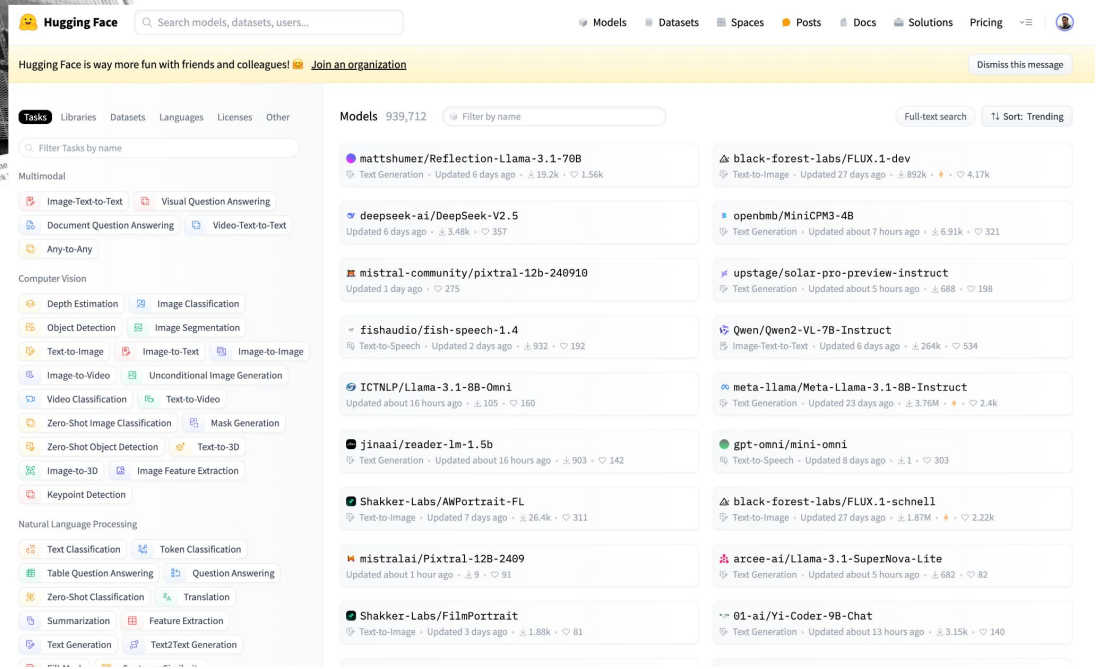
Data replication

- Different use cases.
- Optimizations.
- Data security.
- Anti-corruption (layer).
- https://github.com/maarten-vandeperre/cdc-based-integration-example

Challenge

Model flexibility

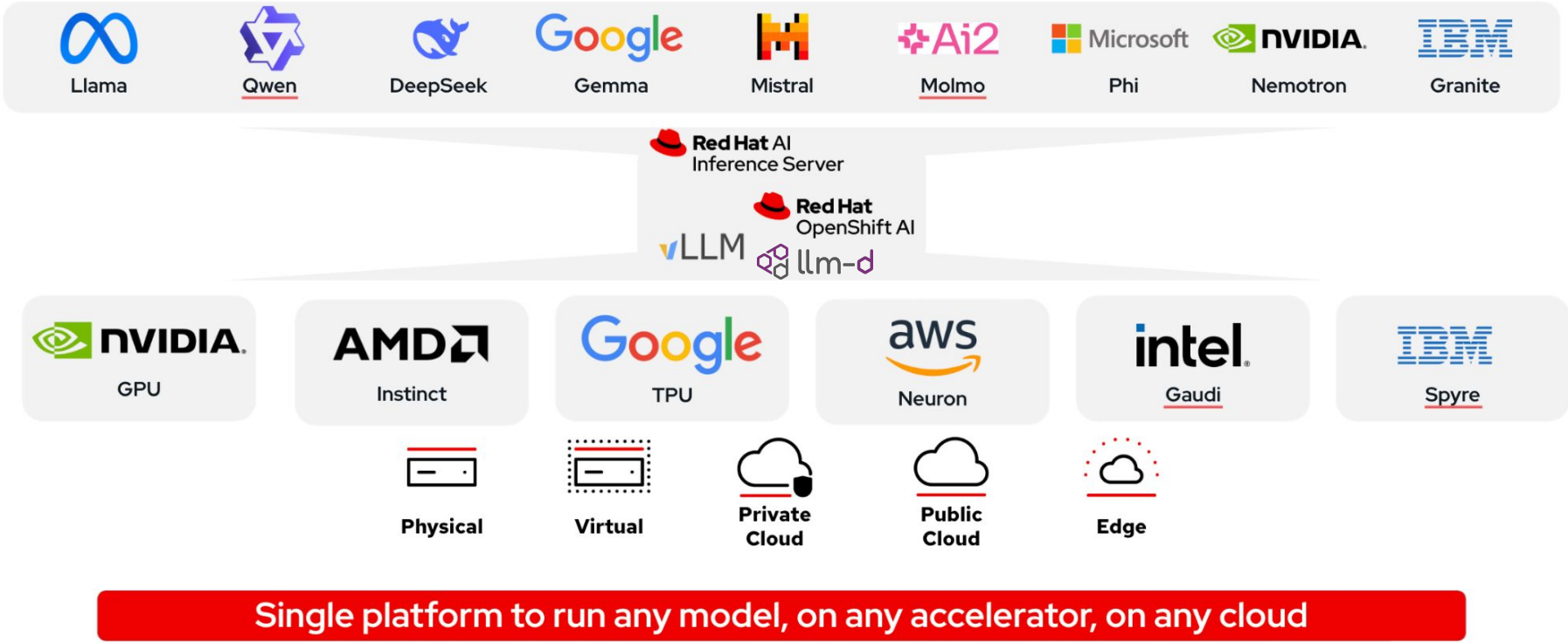




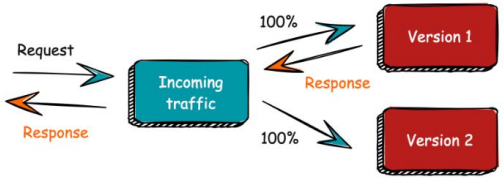
Model selection

- Better capabilities?
  E.g., code migration use case.
- Licensing?
  E.g., New York Times use case.

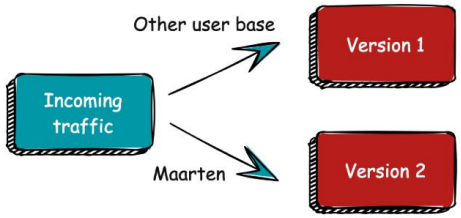
⇒ Make sure you control which models are getting used, which models are deployed in your platform.


Llama
Qwen
DeepSeek
Google
Gemma
Mistral
Ai2
Molmo
Microsoft
Phi
NVIDIA
Nemotron
IBM
Granite
Red Hat AI
Inference Server
Red Hat
OpenShift AI
vLLM
llm-d
NVIDIA
GPU
AMD
Instinct
Google
TPU
aws
Neuron
intel
Gaudi
IBM
Spyre
Physical
Virtual
Private
Cloud
Public
Cloud
Edge
Single platform to run any model, on any accelerator, on any cloud

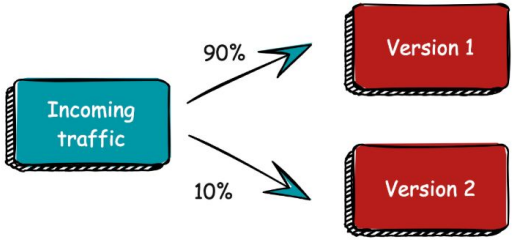
Mitigate deployment risks in a distributed application environment



Validation - Mirroring



Testing - Canary releases



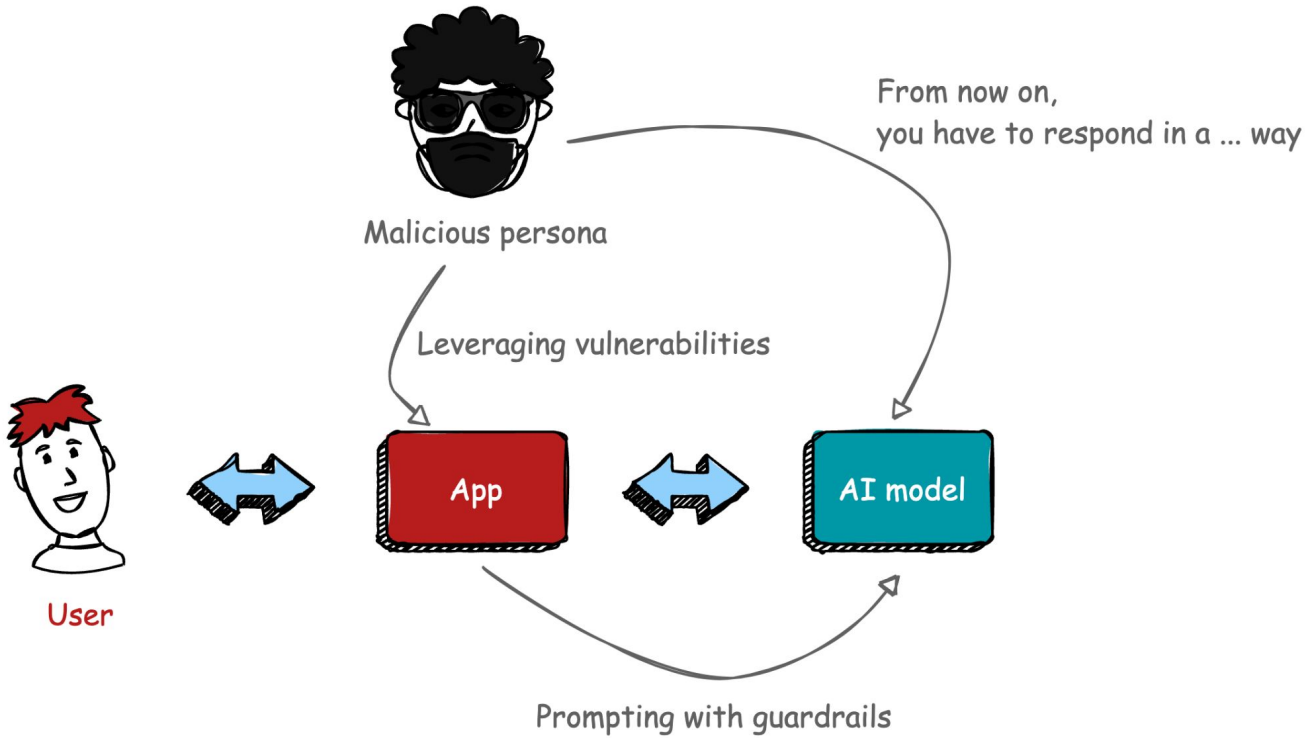
Rolling out - Blue/green deployments

Service mesh

- Deployment strategies
- Mirrorring
- Canary
- Blue/green

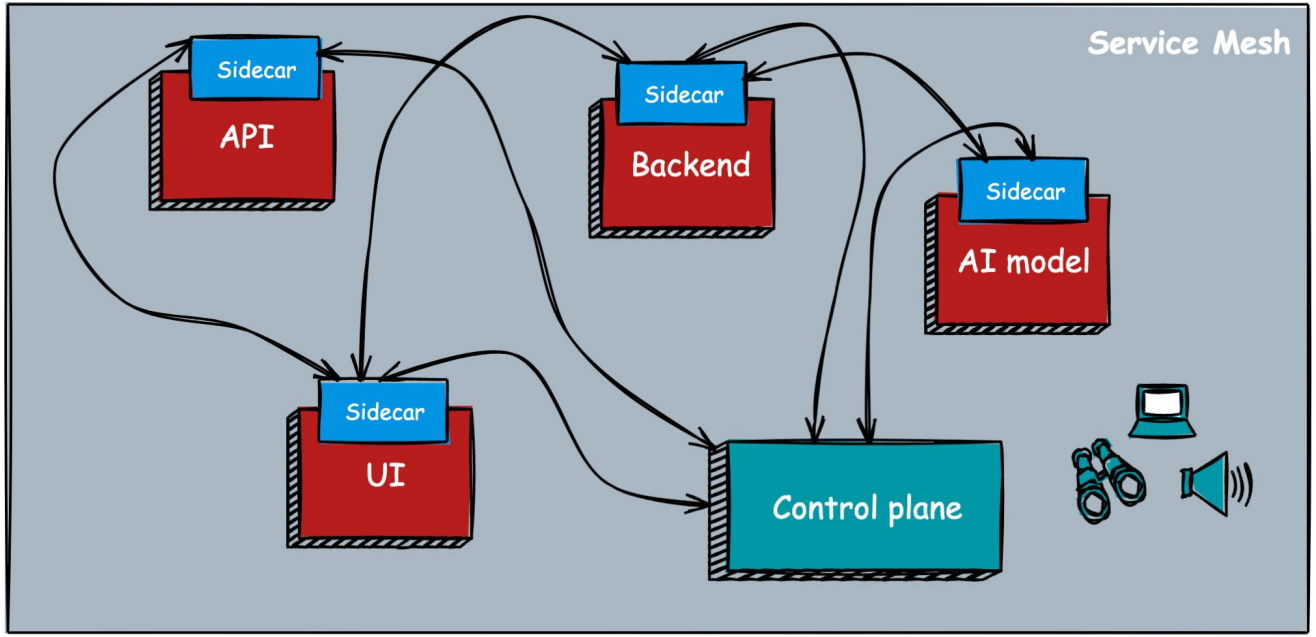
Challenge

Model security



Jailbreak

- methods used to bypass or manipulate the intended restrictions, safeguards, or ethical guidelines set by the developers of these models.
- make the model perform tasks it wasn't designed to do. E.g., to test its limits, expose vulnerabilities, or to extract information and capabilities not normally accessible.
- can lead to harmful outputs from the model, misuse of the AI data breaches, and ethical concerns regarding the misuse of AI technology.

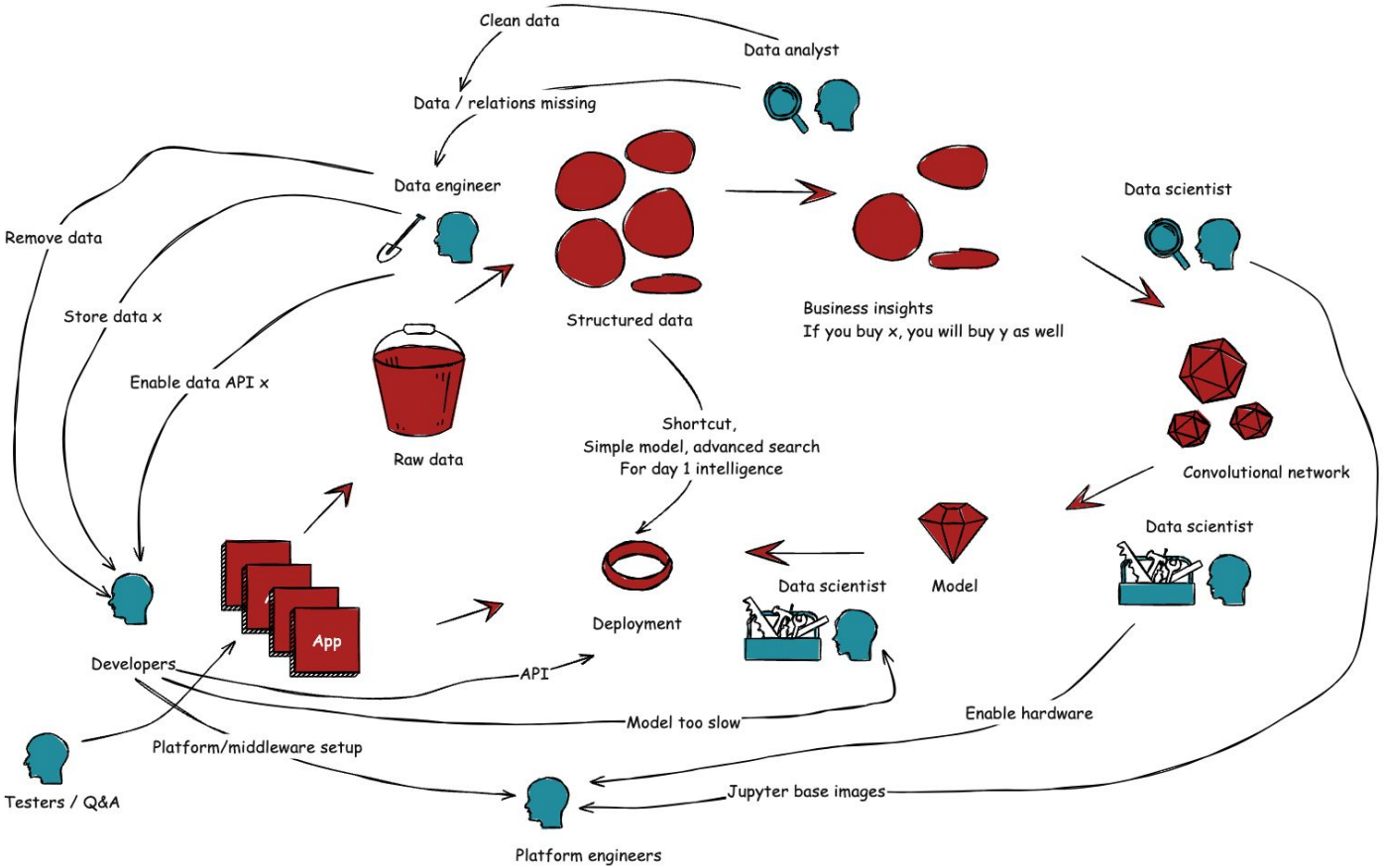


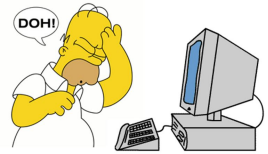
Service mesh

- Observe, monitor and protect your models with service mesh.
- Throttling of model calls.
- Avoid DDOS.
- Limit access to AI model.

Challenge

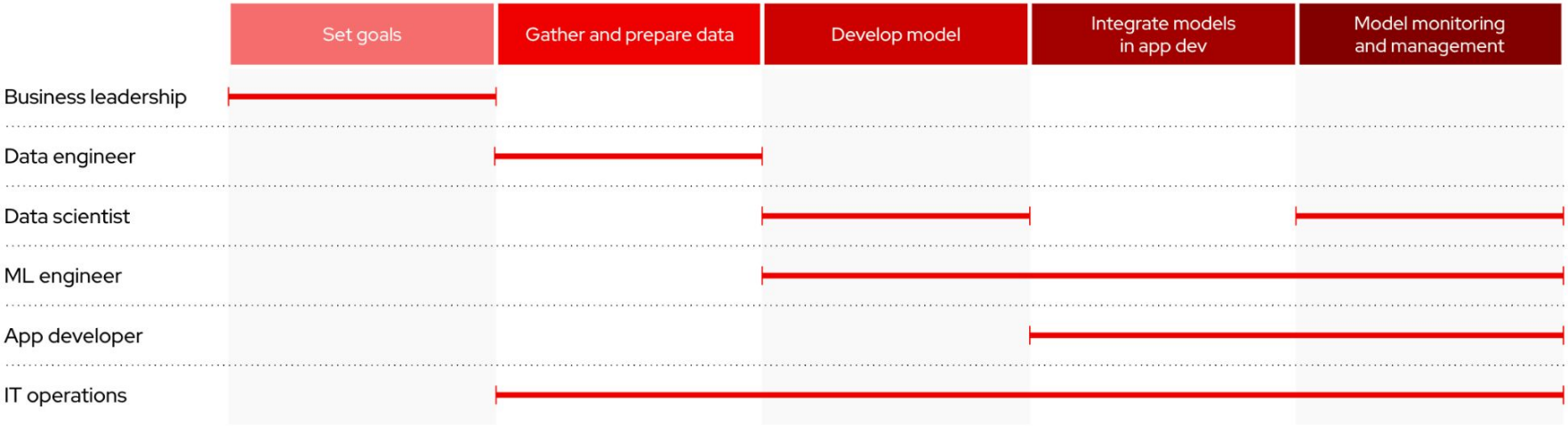
Personas


Clean data
Data analyst
Data / relations missing
Data engineer
Remove data
Store data x
Enable data API x
Structured data
Business insights
If you buy x, you will buy y as well
Data scientist
Raw data
Shortcut,
Simple model, advanced search
For day 1 intelligence
Convolutional network
Data scientist
Model
Data scientist
Deployment
App
Developers
API
Model too slow
Enable hardware
Platform/middleware setup
Testers / Q&A
Jupyter base images
Platform engineers


DOH!

Operationalizing AI/ML is not trivial

Every member of your team plays a critical role in a complex process



Traditional App Delivery

Dev

- Applications
- App Deploy
- App Build
- Entitlement/Billing
- Runtime
- Java
- Guest OS
- Guest Virtualization
- Clustering/HA
- Host Virtualization
- OS
- Hardware

Ops

Key:

- Dev Responsible
- Ops Responsible

Traditional App Delivery

Dev

- Applications
- App Deploy
- App Build
- Entitlement/Billing
- Runtime
- Java
- Guest OS
- Guest Virtualization
- Clustering/HA
- Host Virtualization
- OS
- Hardware

Ops

Key:

- Dev Responsible
- Ops Responsible

Modern App Delivery

Usability

Observability

Quality

Dev

- Applications
- App Deploy
- App Build
- Entitlement/Billing
- Runtime
- Java
- Guest OS
- Guest Virtualization
- Clustering/HA
- Host Virtualization
- OS
- Hardware

Ops

Dev

Applications

Platform Engineering

App Build

App Deploy

Entitlement/Billing

Runtime

Java

Guest OS

Ops

Entitlement/Billing

Guest Virtualization

Clustering/HA

Host Virtualization

OS

Hardware

Internal Developer Platform

IDP

What happened?!

- Unrealistic targets
- Treated as a toy ⇒ we'll need to bring it to production
- How?
  - Concepts
  - **⇒ Challenges & proposals**
    - Tools ⇒ Platform engineering - internal developer platforms
    - Big bang: No step-by-step/agile approach ⇒ Road towards AI (KISS)
    - Model enhancements
    - Sovereignty + lack of dedicated environments ⇒ Data platforms
    - Model flexibility ⇒ Kubernetes (Kubeflow), vLLM, service mesh
    - Model security ⇒ Service mesh
    - # Personas ⇒ Platform engineering

What did we learn today?

1. AI is more than ML.
2. You need an agile application platform.
3. You need a solid data platform/layer.
4. You need deployment strategies.
5. You need to protect the models.
6. Monitor bias and drift.

[7. Red Hat is here to help.]

The six amendments of production-grade AI.

Resources

- https://www.linkedin.com/in/camille-nigon/

- https://www.linkedin.com/in/maarten-vandeperre/

- https://github.com/camille-maarten/production-grade-ai
- https://developers.redhat.com/articles/2025/06/16/how-kafka-improves-agentic-ai
- https://developers.redhat.com/articles/2025/06/16/how-use-service-mesh-improve-ai-model-security
- https://developers.redhat.com/

Thank you!

Red Hat