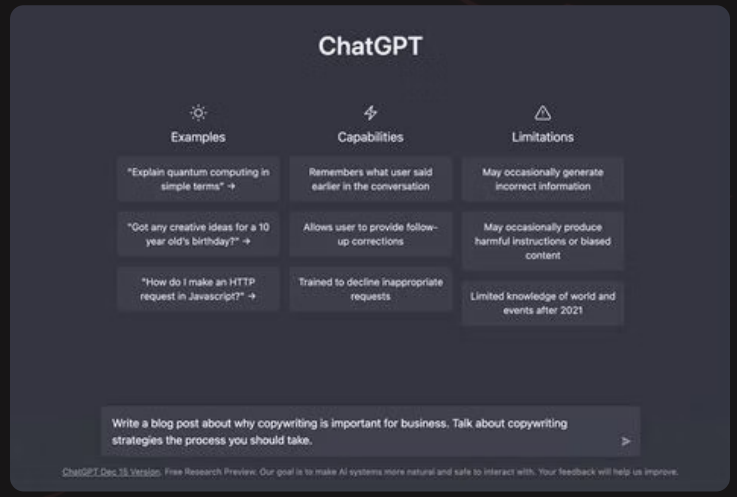
Day2 - Introduction to MLOps

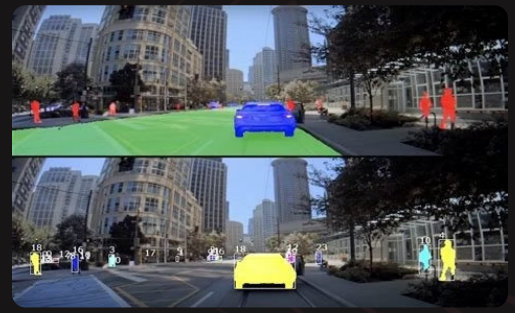
02476 Machine Learning Operations

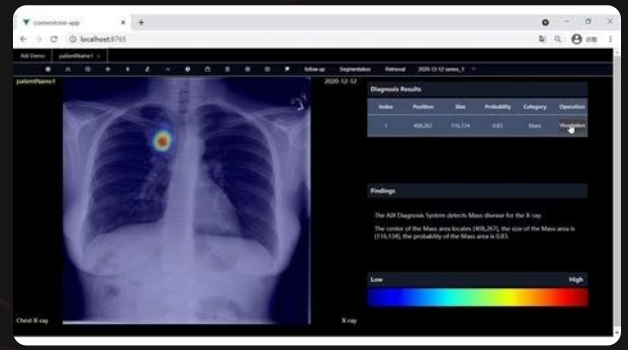
Nicki Skafte Detlefsen, Associate Professor, DTU Compute

January 2026

Let's agree that ML/AI is fantastic







💡 AI is a key component of what we call industry 5.0

💡 It can solve problems on unprecedented scales

But errors does happen

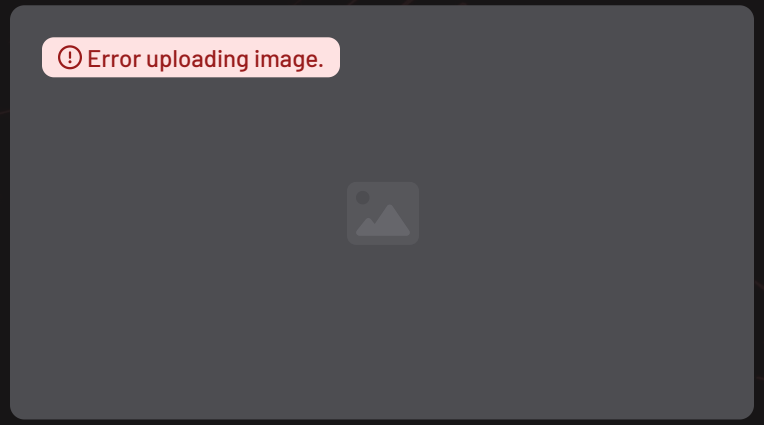
Error uploading image.

💡AI, similar to humans, sometimes takes the wrong decision

💡But due to its unpresented scale, things can quickly go very wrong

The duality of AI

Developing AI is easy – Running AI in production is hard

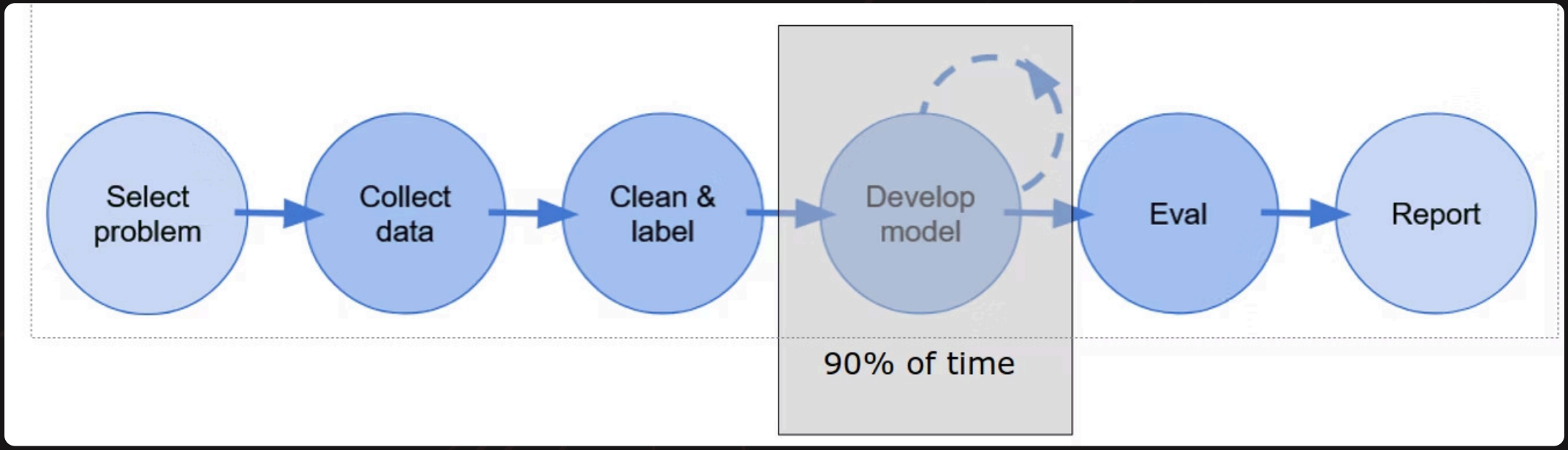


Why is this?

Why do we focus on modelling?

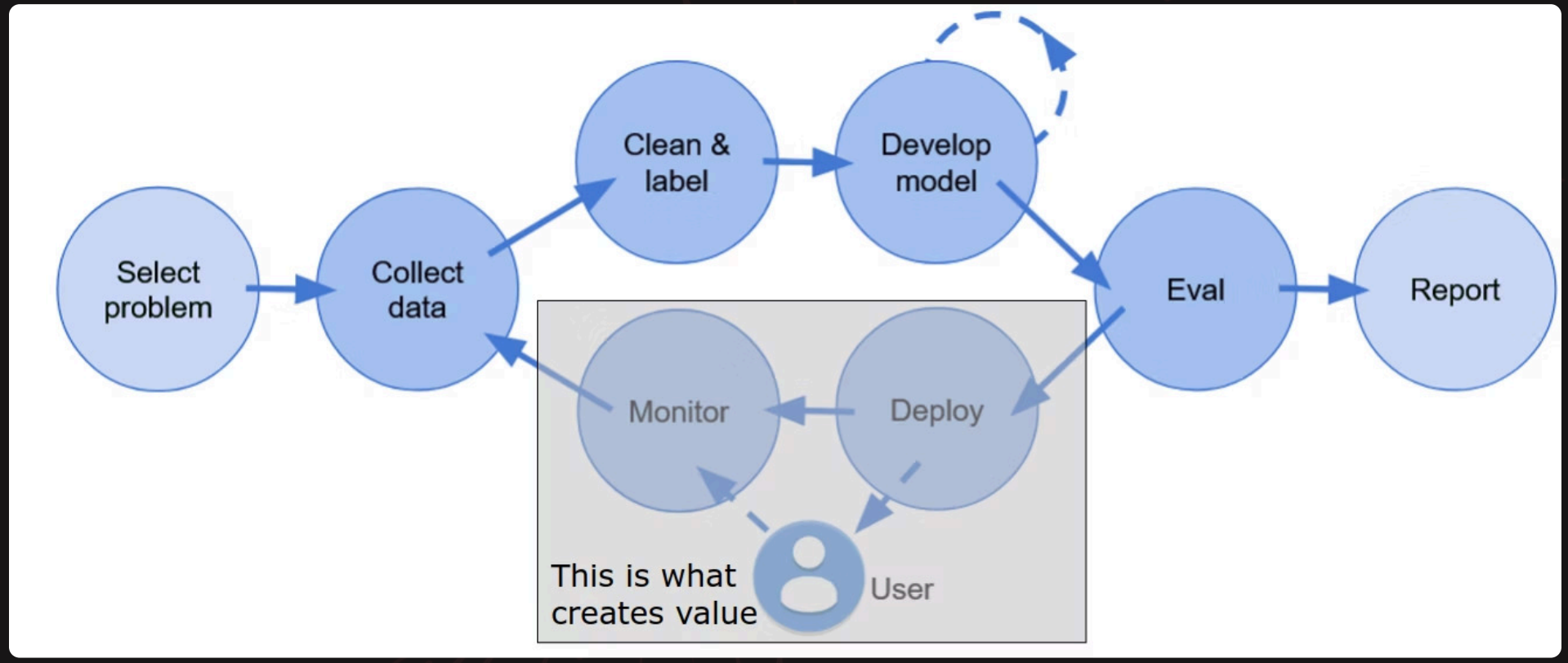
Because we teach people it!

Courses / Projects are linear in nature



Feedback is grades / funding

Machine Learning in the real world



AI is Software

At its core, Artificial Intelligence, including Machine Learning, is a specialized form of software. It comprises algorithms, code, and data designed to perform specific tasks, just like any other application.

✅ We have been developing software for 30+ years

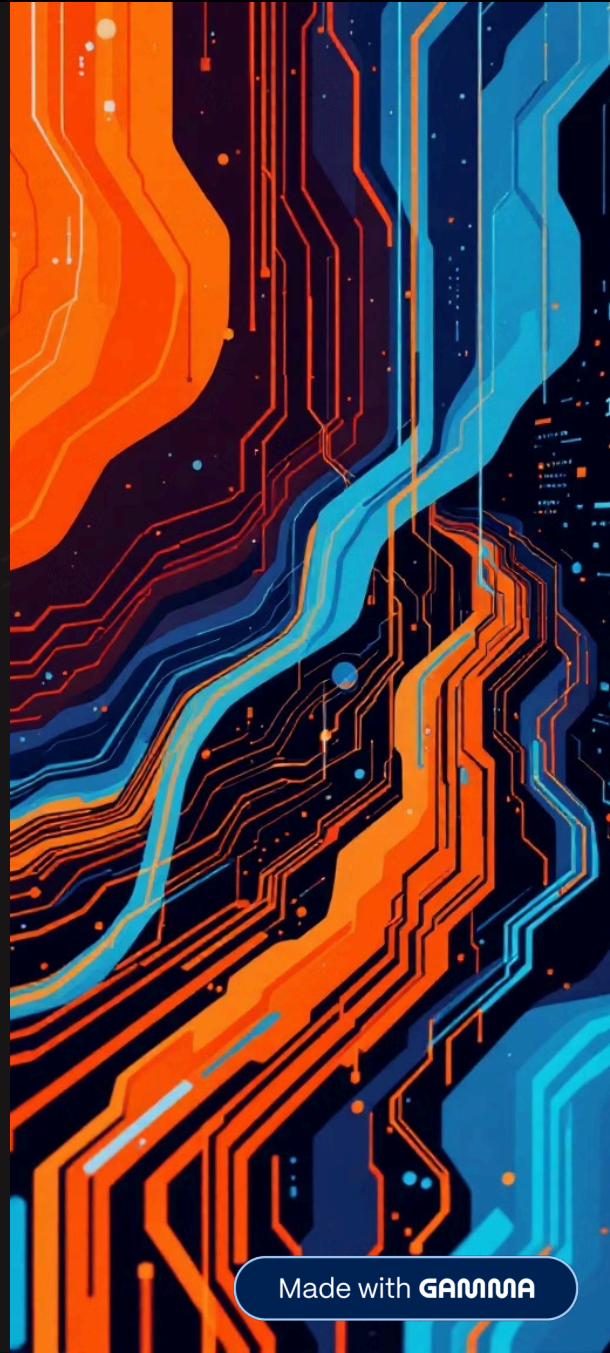
✅ We have a lot of tools for software development

❌ Software can break

❌ Software needs to be maintained

❌ Delivering software suffers from the "last mile problem"

Lets look at two problems AI inhere from software



The Challenge: The "Last Mile" of Machine Learning

Most machine learning models never create value because the path from a data scientist's laptop to a live production environment is broken.

The "Works on My Machine" Problem

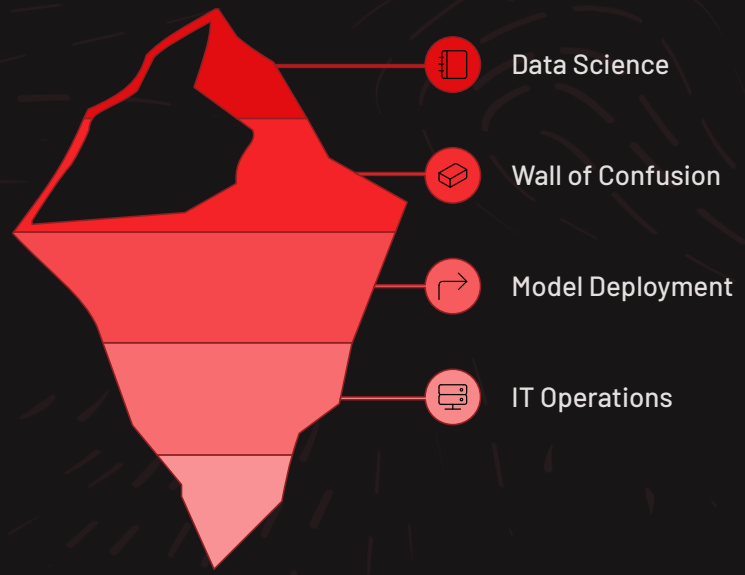
- Models developed in isolated environments (Jupyter notebooks) are difficult to reproduce, test, and scale.
- A significant gap exists between what data scientists build and what operations teams can support.

Manual, Slow, and Risky Handoffs

- Deploying models often involves manually "throwing them over the wall" from Data Science to IT/Engineering.
- This process is slow, error-prone, and lacks clear ownership.

Models Decay in the Real World

- The environment changes, and so does data; today's accurate model might be tomorrow's failure.
- Without monitoring and retraining, performance degrades silently, leading to poor business outcomes.



The Challenge: Hidden Technical Debt

The long-term cost of short-term solutions and neglected maintenance.

Unseen Interdependencies

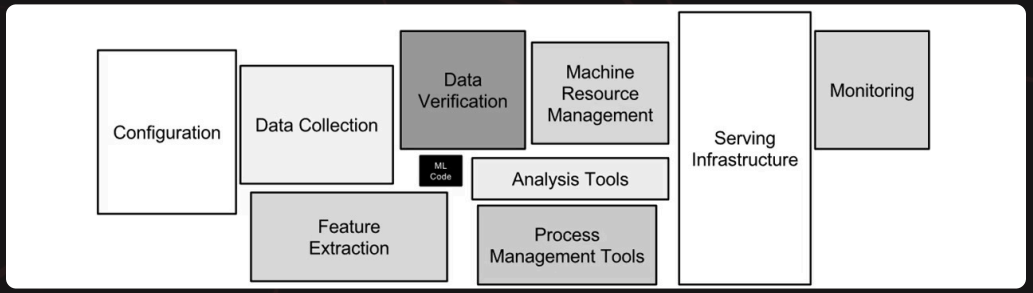
Intricate ML systems with undocumented connections cause cascading failures, difficult to diagnose.

Silent Data Degradation

Data pipelines degrade over time from schema changes or data drift, eroding model performance silently.

Undocumented Logic

Poorly documented training logic and configurations hinder reproducibility, auditing, and future modifications.



[1] Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., & Dennison, D. (2015). Hidden Technical Debt in Machine Learning Systems.

If AI is software, maybe the solution comes from software?

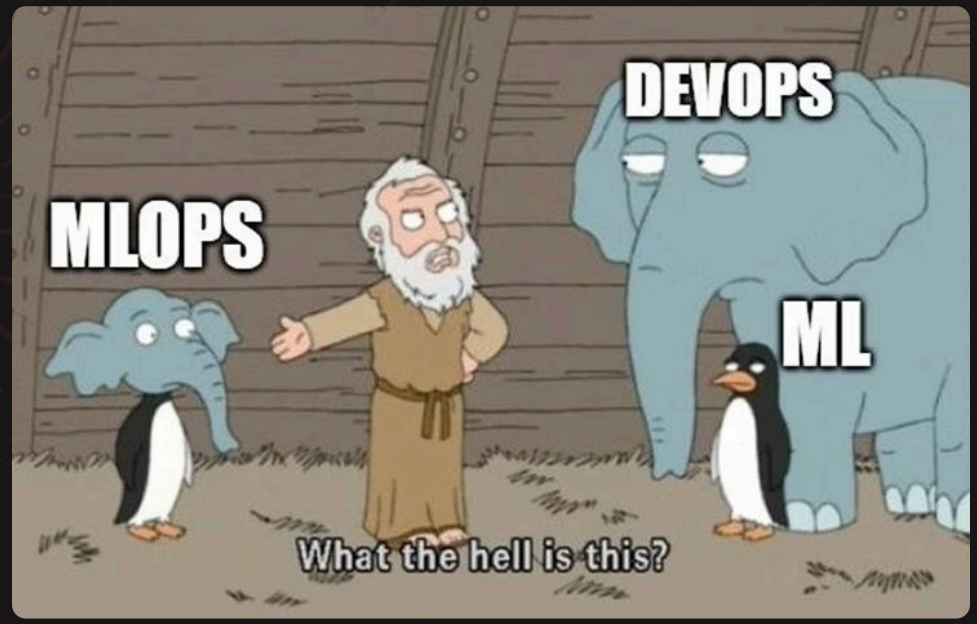
DevOps = Developer operations

- Dates to late 80s and early 90s
- Around 2007/2008 rose to popularity to remove the separation between software development with its operations part/IT department

💡This is both a joke and not.

💡MLOps is directly derived from DevOps.

💡Therefore, let's try to understand DevOps first.



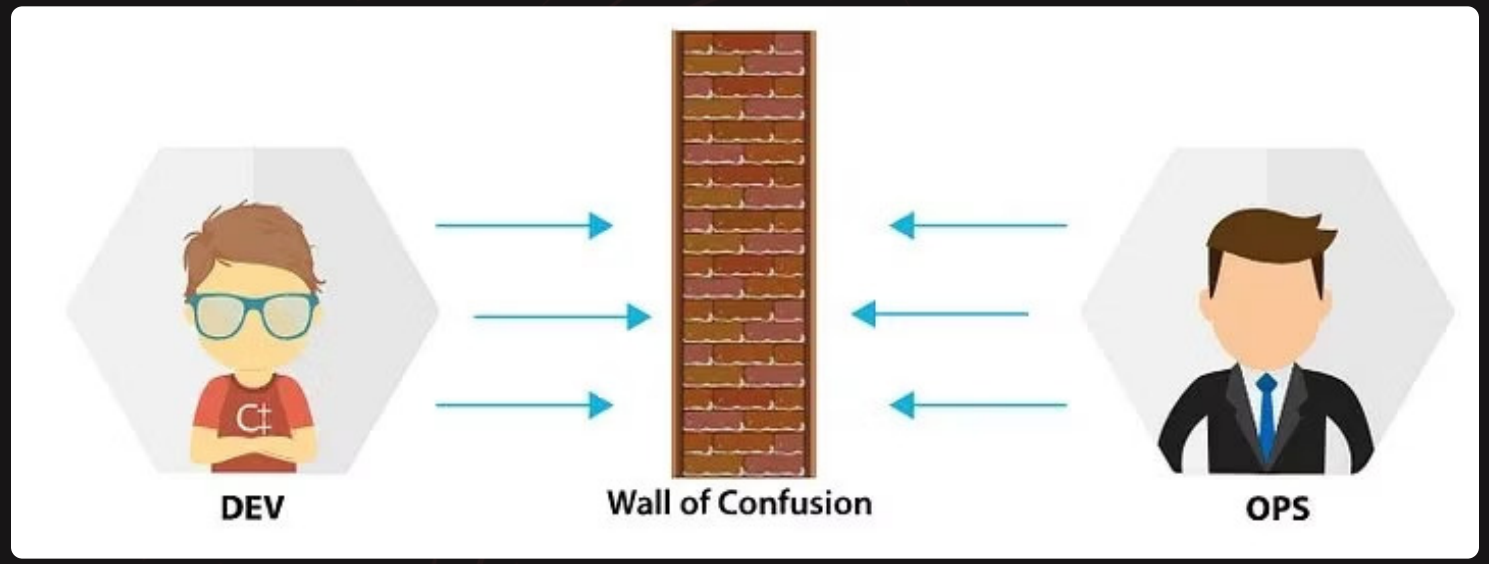
The core problem

There are in general two teams in software development

💡 Dev team = development and improvement of software

💡 Ops team = infrastructure and operations

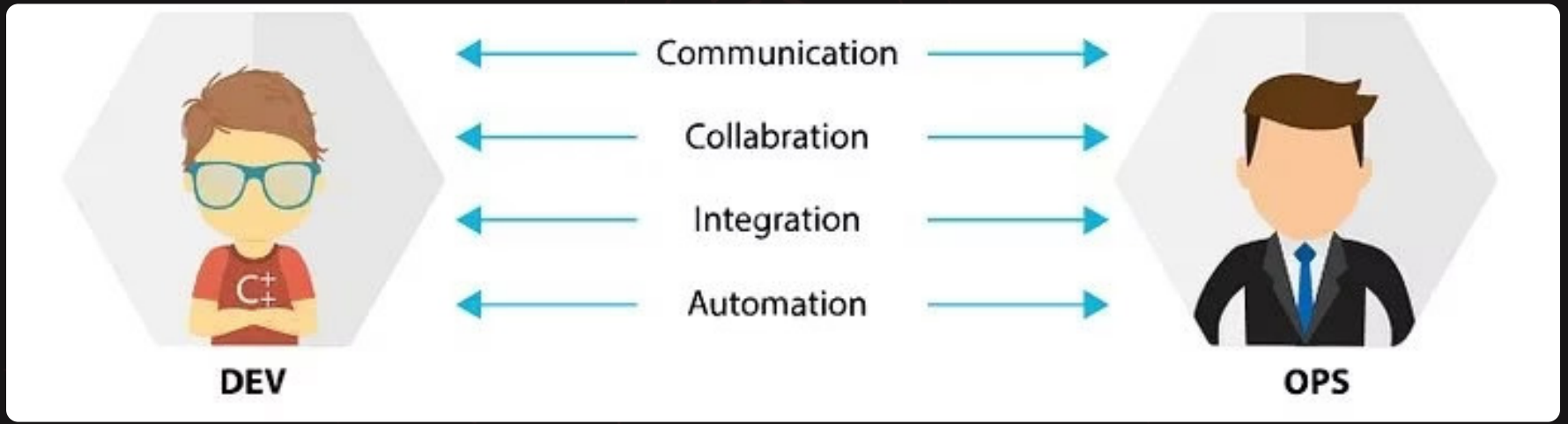
If these two teams does not communicate, then Dev may develop software that Ops cannot operationalize or Ops may setup the wrong infrastructure in relation to what Dev is creating



So, what is DevOps?

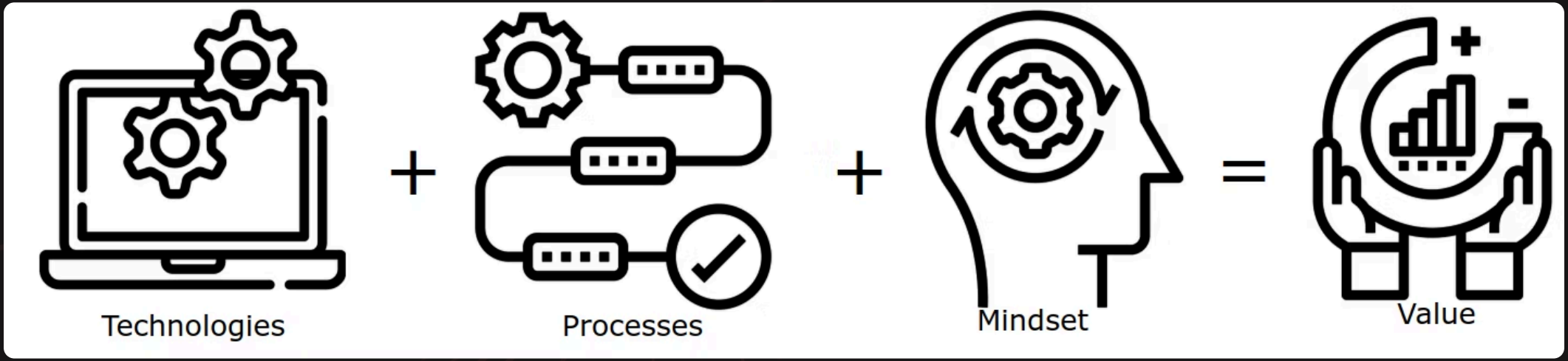
This is the closest to a definition that I could find

> DevOps is a set of practices that combines software development (*Dev*) and IT operations (*Ops*). It aims to shorten the systems development life cycle and provide continuous delivery with high software quality. It's an combination of human mindset, processes and technologies that continuously creates value.



So, what is DevOps?

DevOps is a set of practices that combines software development (*Dev*) and IT operations (*Ops*). It aims to shorten the systems development **life cycle** and provide continuous delivery with high software quality. It's an combination of **human mindset**, **processes** and **technologies** that continuously creates value.



Technology, Processes, Mindset

Use technologies that support the different parts of the lifecycle

Error uploading image.

Implement processes to make sure everyone is in sync about the lifecycle

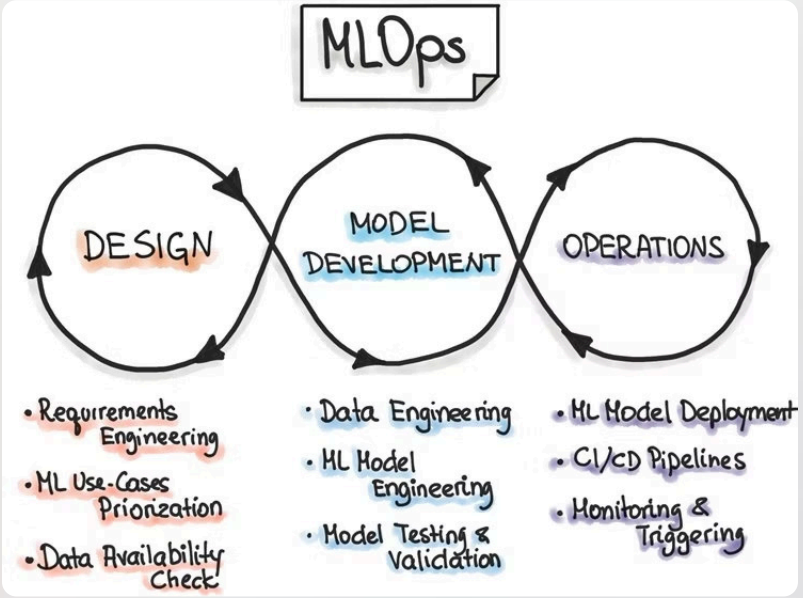
Error uploading image.

Always consider all part of the lifecycle, not just its parts

But then MLOps must be

Is a set of **tools**, **processes**, and **mindset** that aim to make **ML Lifecycle** create value.

To MLOps (verb): To **harmonize** the creative process of data science with the discipline of software engineering, creating a continuous, automated loop that moves models from an experimental notebook to a reliable production environment.

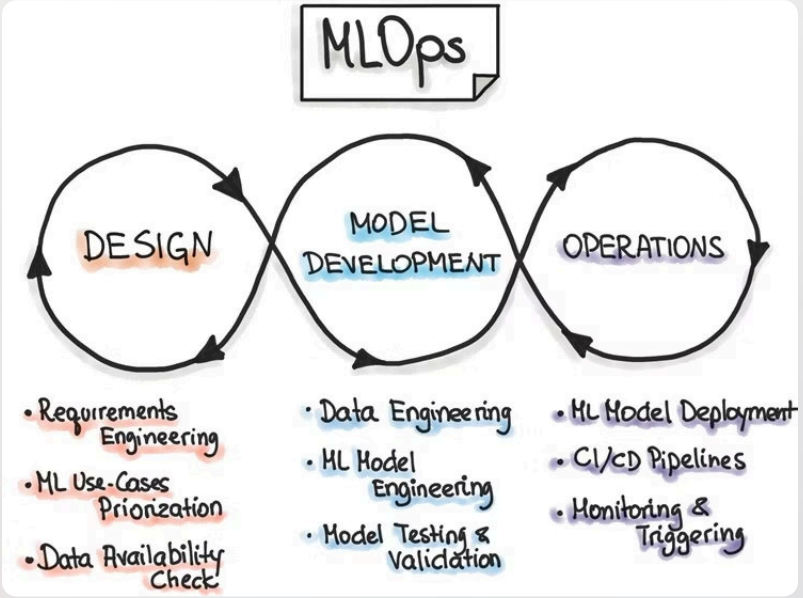


Design Phase

🔥Business understanding

🔥Data understanding

🔥Designing the ML-powered software

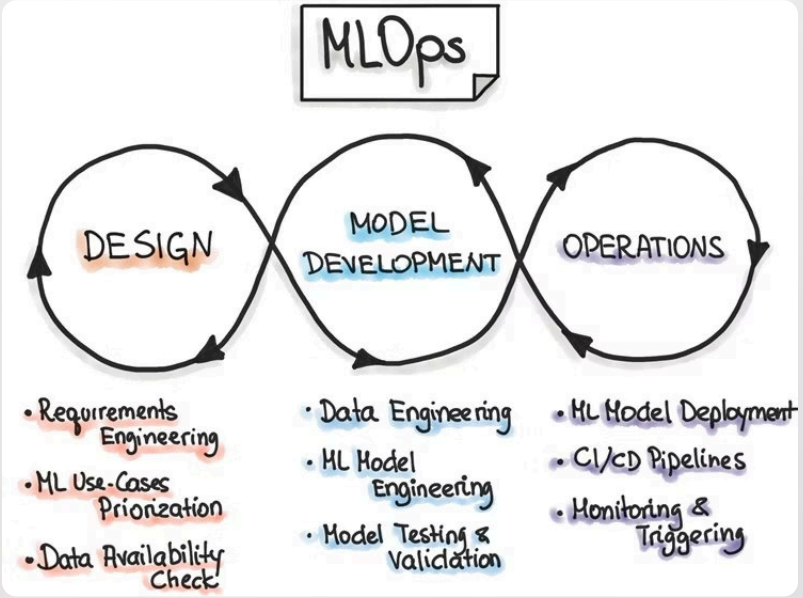


Modelling Phase

🔥Model engineering

🔥Data engineering

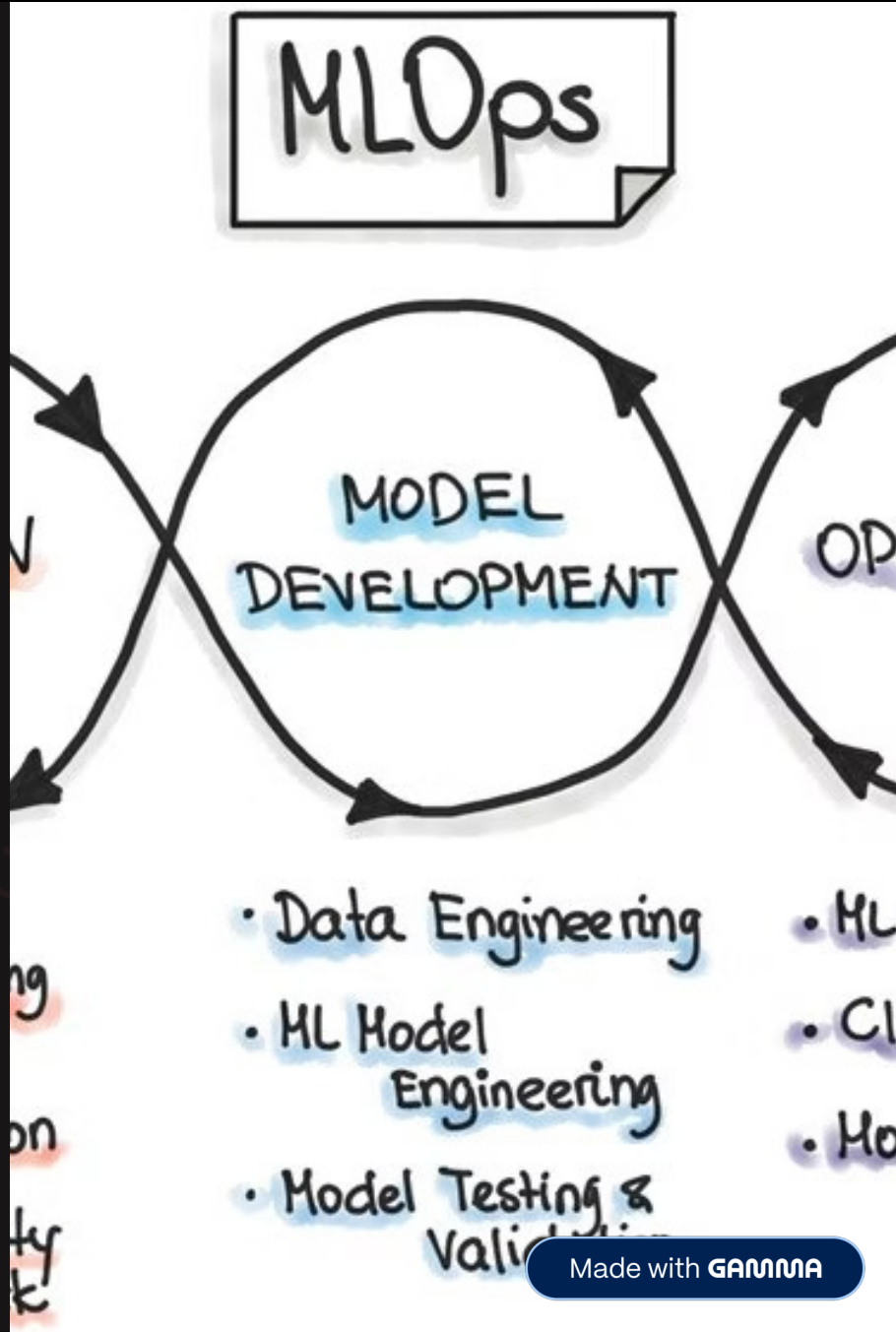
🔥Deliver a stable quality ML model that we will run in production



Operations Phase

🔥Deliver the previously developed ML model in production

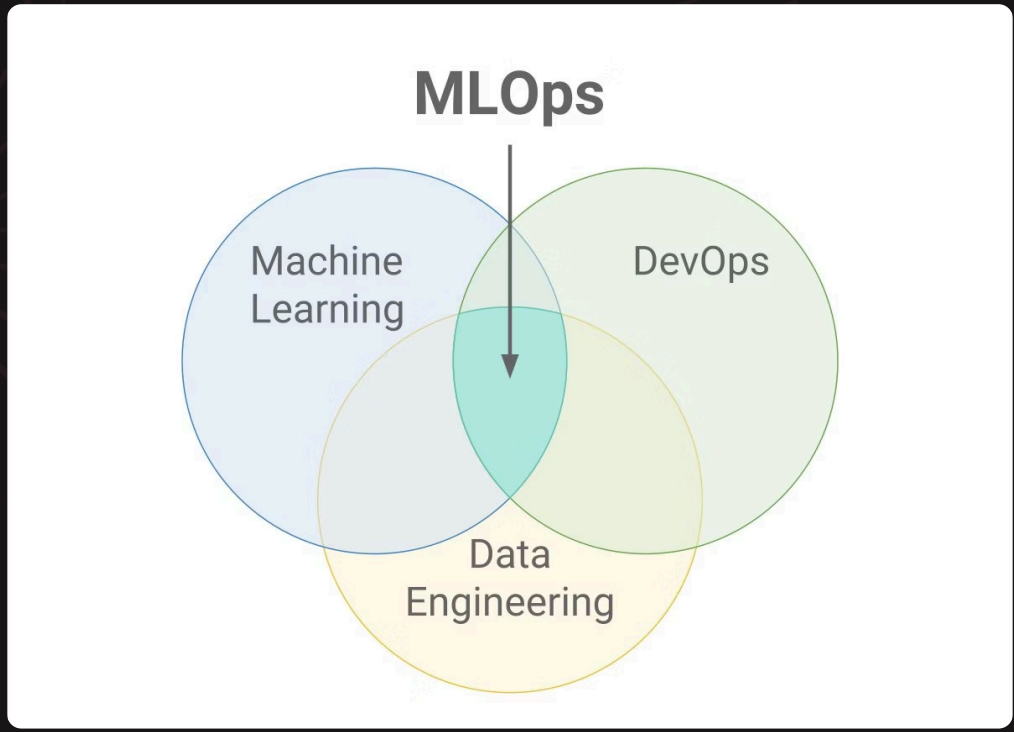
🔥Testing, versioning, continuous delivery, and monitoring



What makes an MLOps engineer?

A mix of

- Software developing
- Machine Learning
- Data engineering



The Hard Truth: Where the Effort Really Goes

50-80% of ML Project Effort is Data Cleaning.

85%

ML models never reach production

(QCon SF 2024)

32%

ML deployments successfully move from pilot to production

(Rexer Analytics 2023)

$3.4B

MLOps market (2024)

Growing at 31.1% CAGR

> "Garbage in, garbage out. If the quality of the data is bad, the quality of the model is bad."
>
> — Wenjie Zi, Grammarly

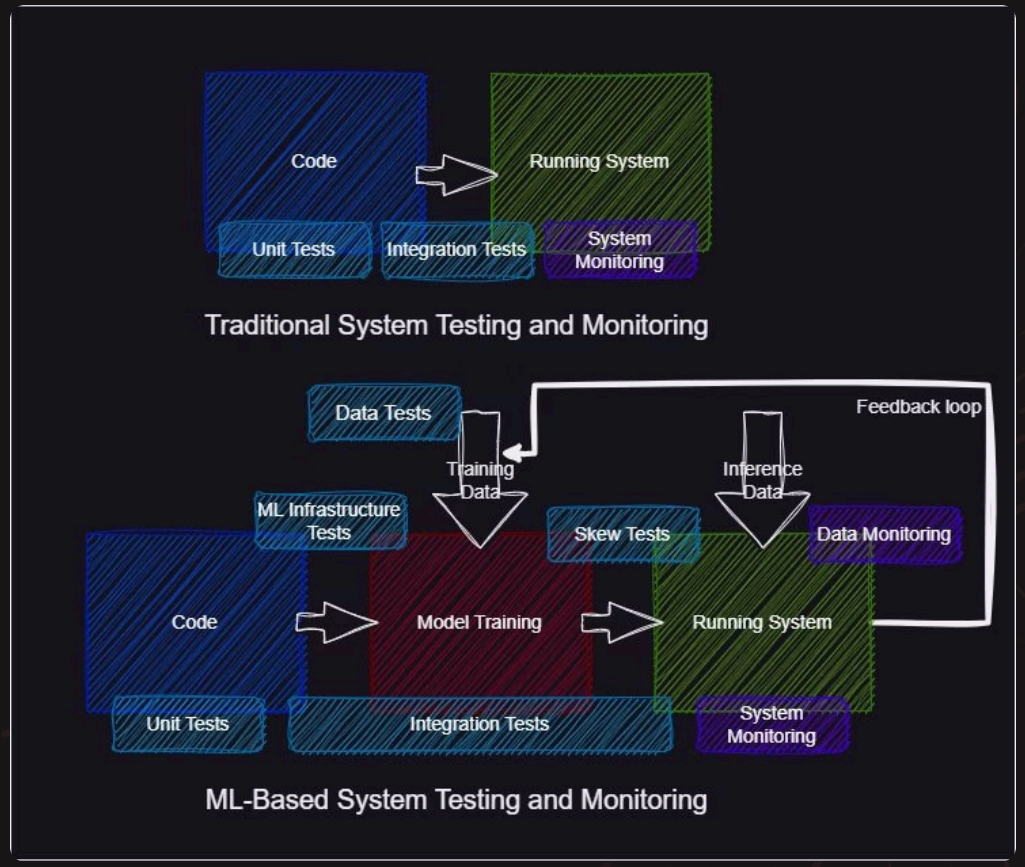
[1] https://www.interregeurope.eu/embraisme

[2] https://www.ml6.eu/blogpost/unlocking-the-full-potential-of-data

[3] QCon SF 2024 presentation by Wenjie Zi (Grammarly)

[4] Rexer Analytics 2023 Data Science Survey

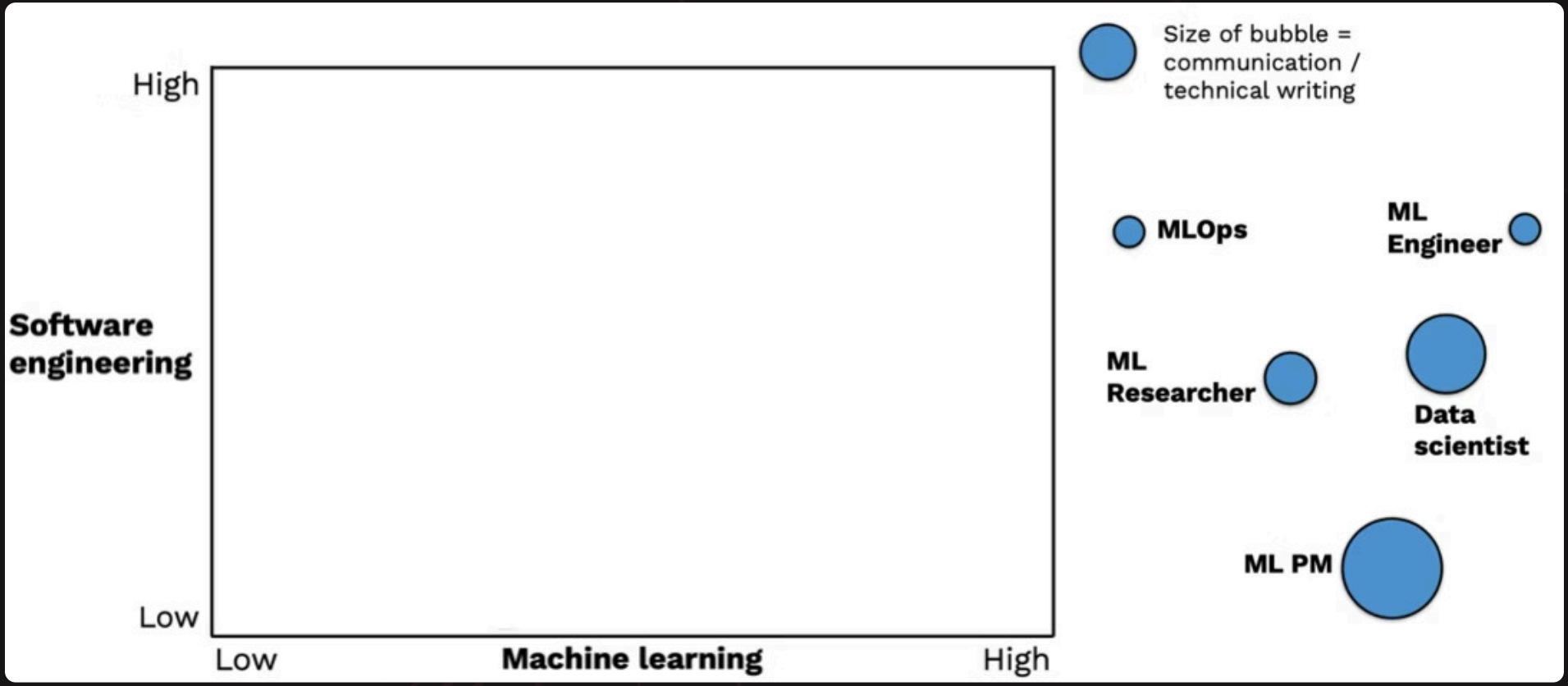
[5] P&S Intelligence MLOps Market Report 2024



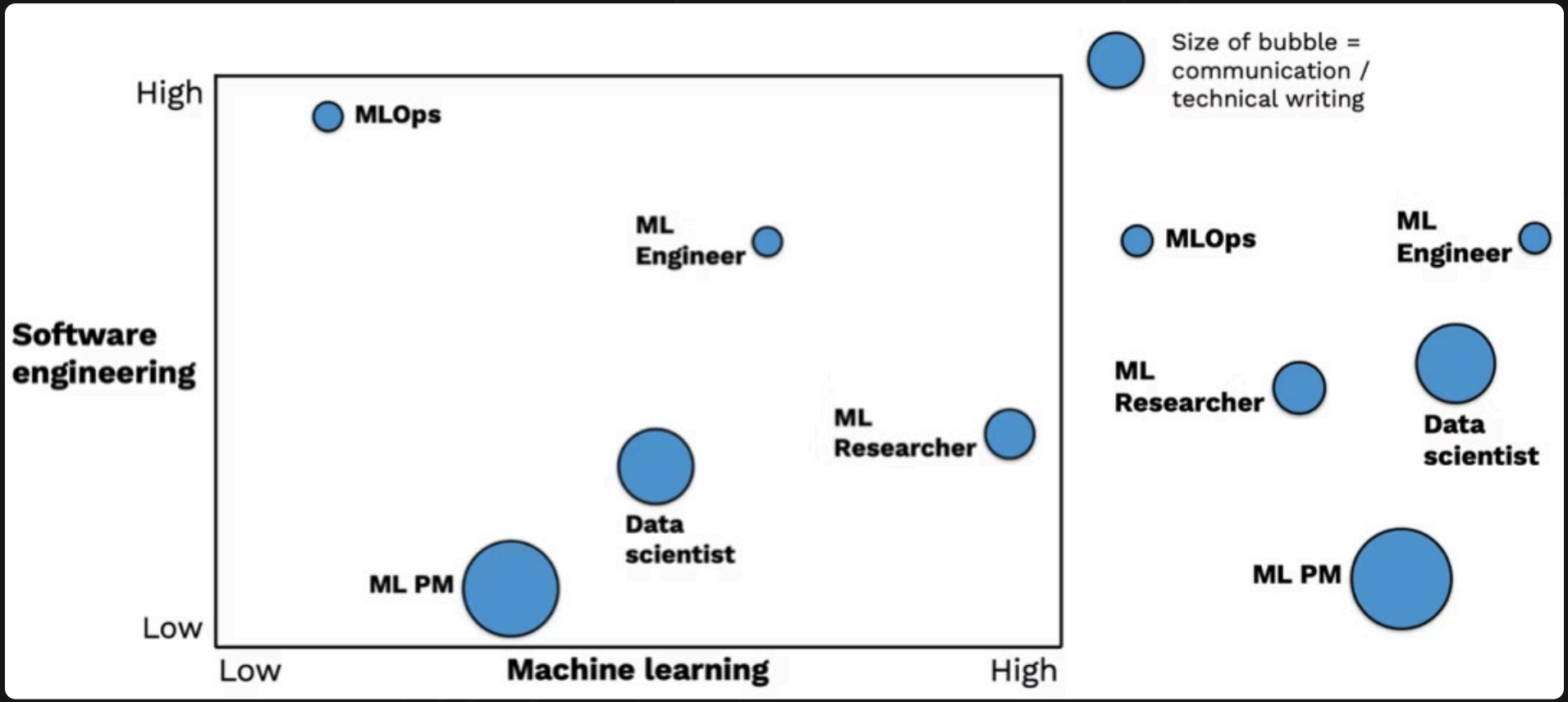
Why is DevOps not enough?

Because the devil is in the ~~details~~ data

Where's Waldo?



Where's Waldo?

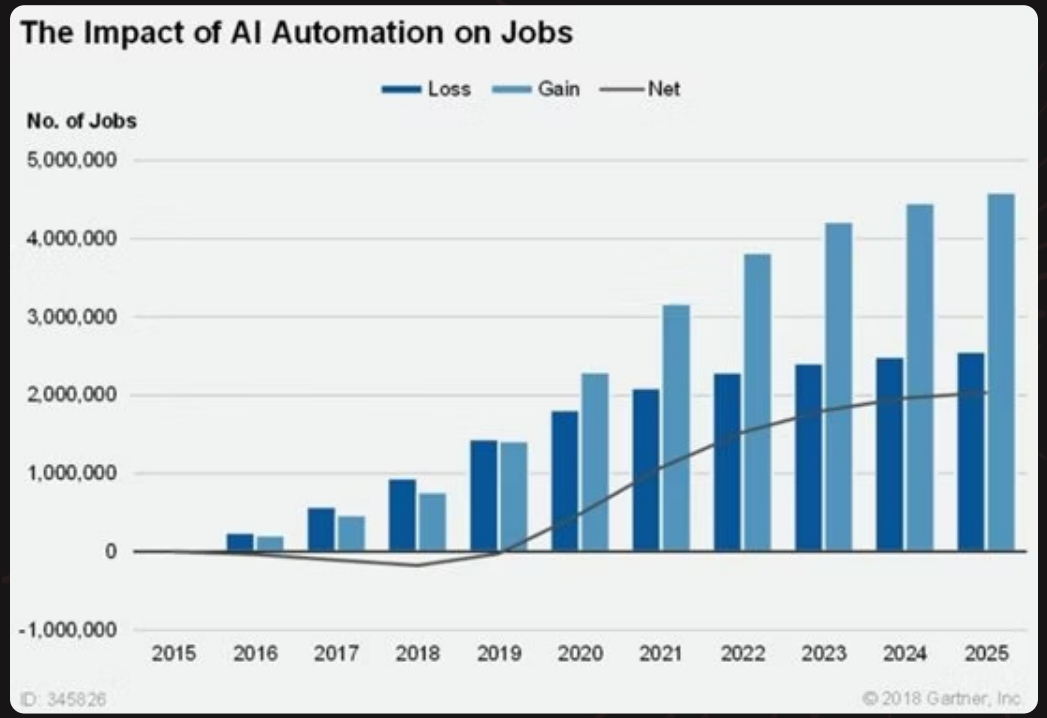


What does a MLOps engineer look like?

Prompt: “Machine Learning operations engineer”

🔥Buff 🔥Locked in 🔥Many screens

Why does companies care about MLOps



Having automated model deployed with errors can cost A LOT of money:

"A famous example of the dangers here was Knight Capital's system losing $465 millions in 45 minutes, apparently because of unexpected behavior from obsolete experimental codepaths"

– Hidden Technical depth in Machine Learning Systems

Knowing "only" machine learning is no longer enough

https://kyunghyuncho.me/i-sensed-anxiety-and-frustration-at-neurips24/

TL;DR:

The AI job market has shifted drastically since the early days of deep learning, causing anxiety among late-stage PhD students and postdocs. Initially, deep learning expertise was rare, and companies aggressively recruited PhD grads with high pay and research freedom. This led to a boom in AI PhD programs. However, with the rise of productized AI, such as large-scale language models, companies now prioritize practical skills over academic research, hiring more undergrad and master's grads. PhDs, trained for innovation, struggle to find the same opportunities. This shift has left many feeling frustrated, anxious, and uncertain about their future in the field.

The MLOps Advantage: Pillars & Payoffs

By implementing MLOps, organizations can deploy higher-quality models faster, with greater reliability and scale.

Core Pillars (The "How")

Automation

Use CI/CD/CT pipelines to automate the entire lifecycle from data ingestion to model deployment.

Reproducibility

Version everything—data, code, and models—to ensure any experiment or result can be reproduced.

Collaboration

Create a single, unified platform for data scientists, ML engineers, and operations to work together.

Governance

Implement robust monitoring, security, and compliance checks to manage risk and ensure fairness.

Business Payoffs (The "Why")

Velocity

Reduce the time to deploy new models from months to days, reacting faster to market changes.

Reliability & Quality

Deliver robust, thoroughly tested, and continuously monitored models that you can trust.

Scalability

Efficiently manage, serve, and monitor hundreds or thousands of models in production.

ROI

Maximize the return on investment from your AI/ML initiatives by actually getting them into production.

Why did OpenAI win to begin with?

🔎What is the contributes to the initial success of OpenAI?

💡First mover advantage

💡Funding

💡Data

💡People

💡Compute

💡Service contract with Microsoft

Why is Google winning right now?

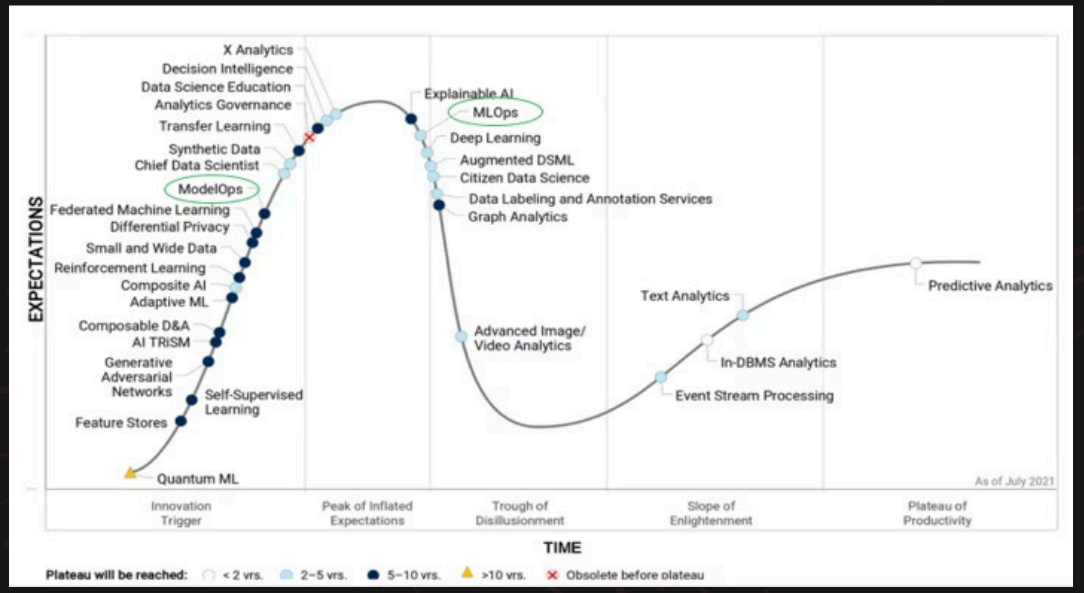
> "Who could have foreseen that the company which have all possible advantages would win in the end?"

💡 Financial stable

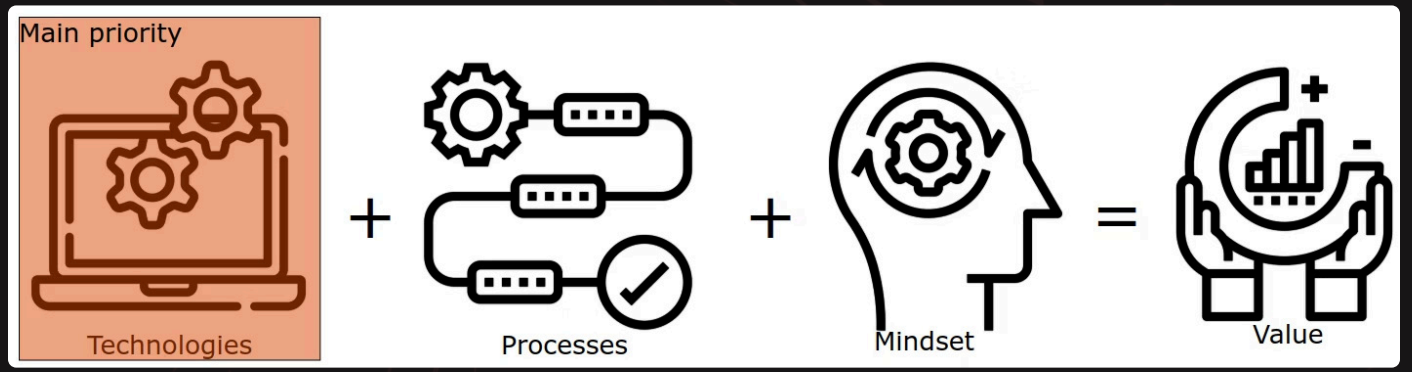
💡 In-house hardware (TPUs)

💡 Scalability

Trends in MLOps



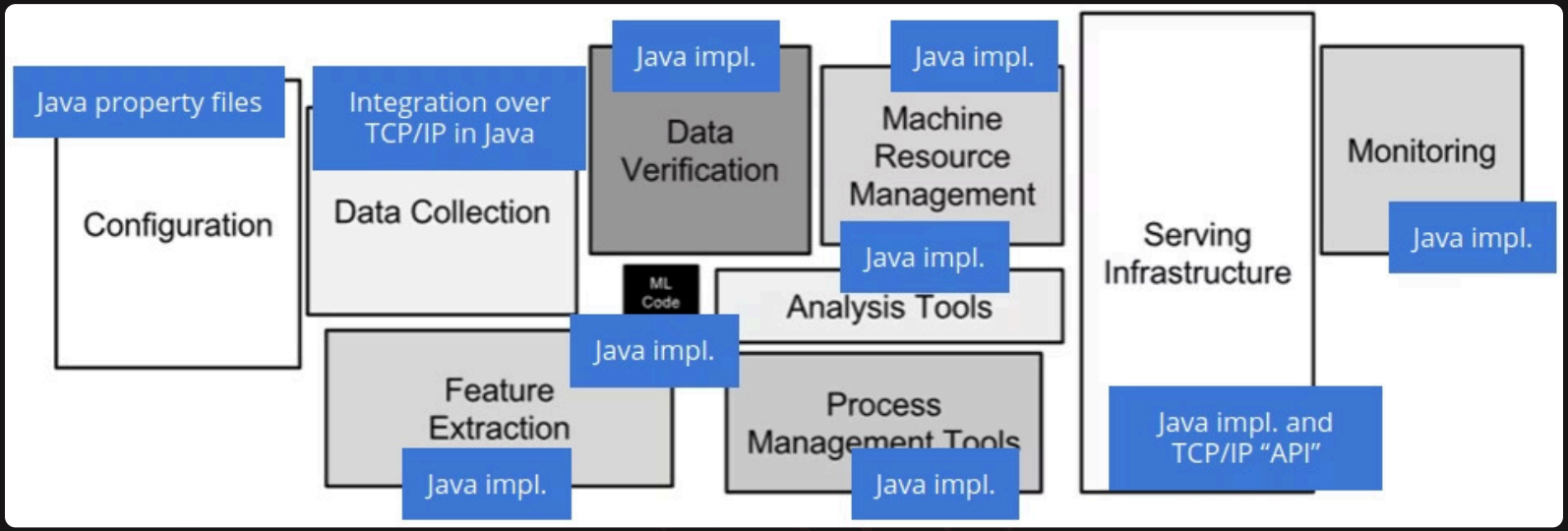
MLOps has been trending for a couple of years. Tools have been the main priority



Choosing the right tool for the job

MLOps has changed a lot over the last couple of years

MLOps back then (ca. 2006)



Pros:

➕ Full control

Cons:

➖ Slow to iterate

➖ Hard to maintain

➖ Lot of manpower per project

Today we have options

Pros:

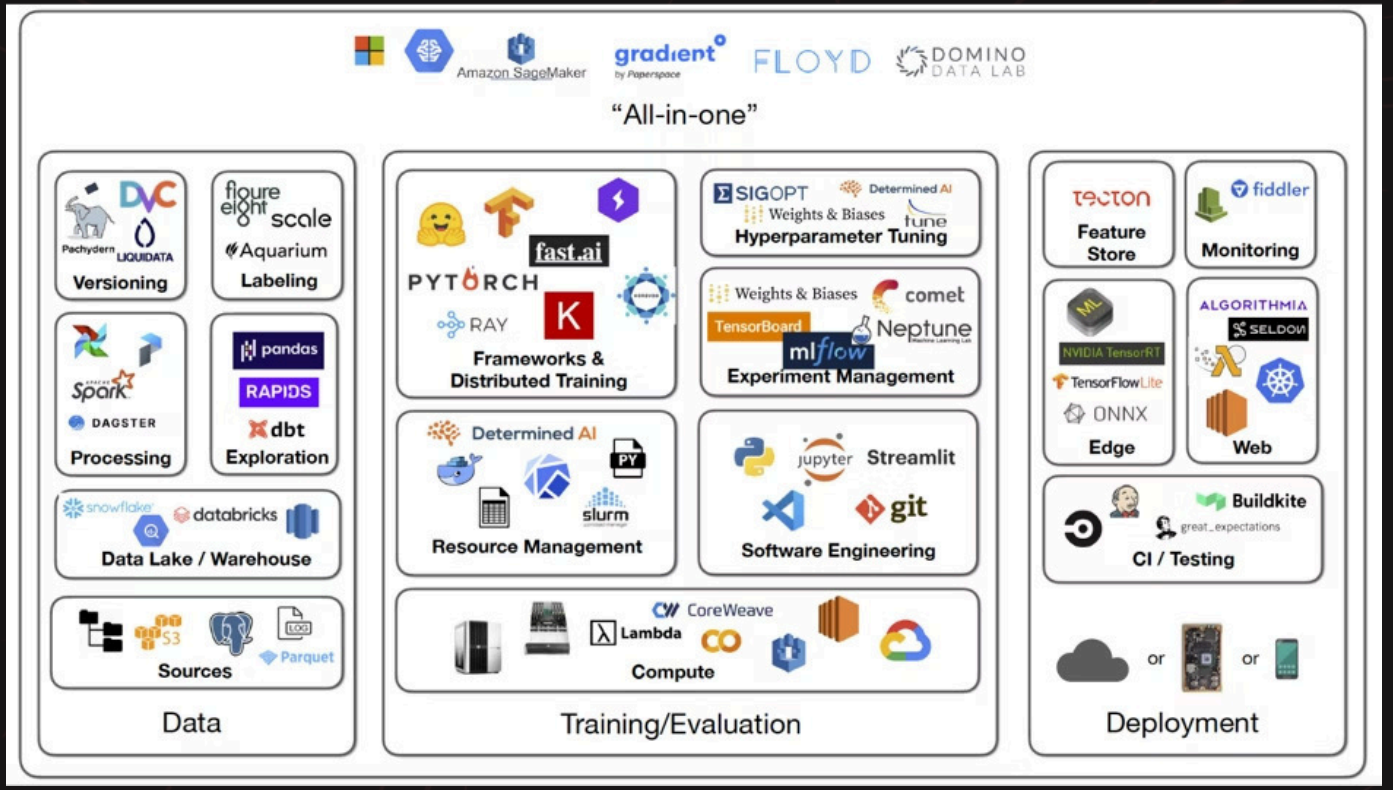
➕ Easy to get started

➕ Easy to iterate

Cons:

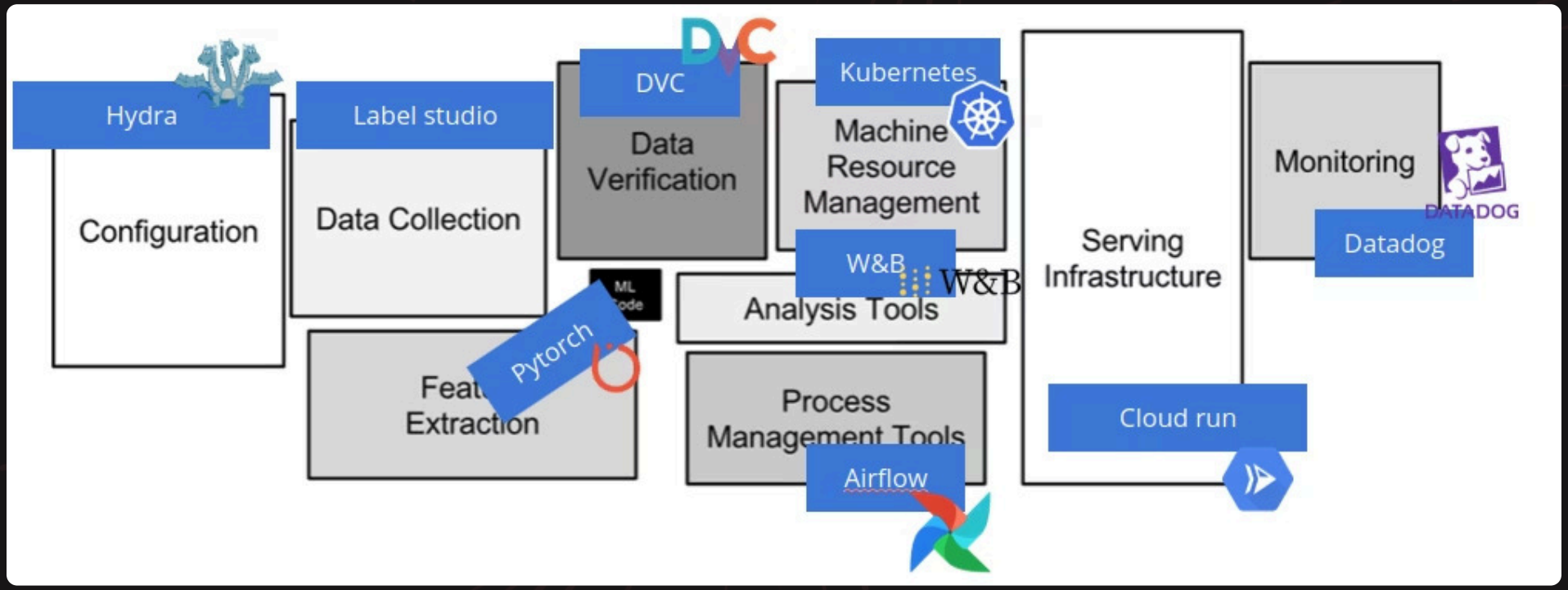
➖ Framework integration can be really hard

➖ Hard to compare frameworks



MLOps now

Pick a *stack* of tools



MLOps is full stack

In MLOps we embrace the full stack of problems that comes from the full lifecycle. Especially integration problems.

Criteria for what goes into the stack (4Cs):

💡 Cost

💡 Coverage

💡 Complexity

💡 Community

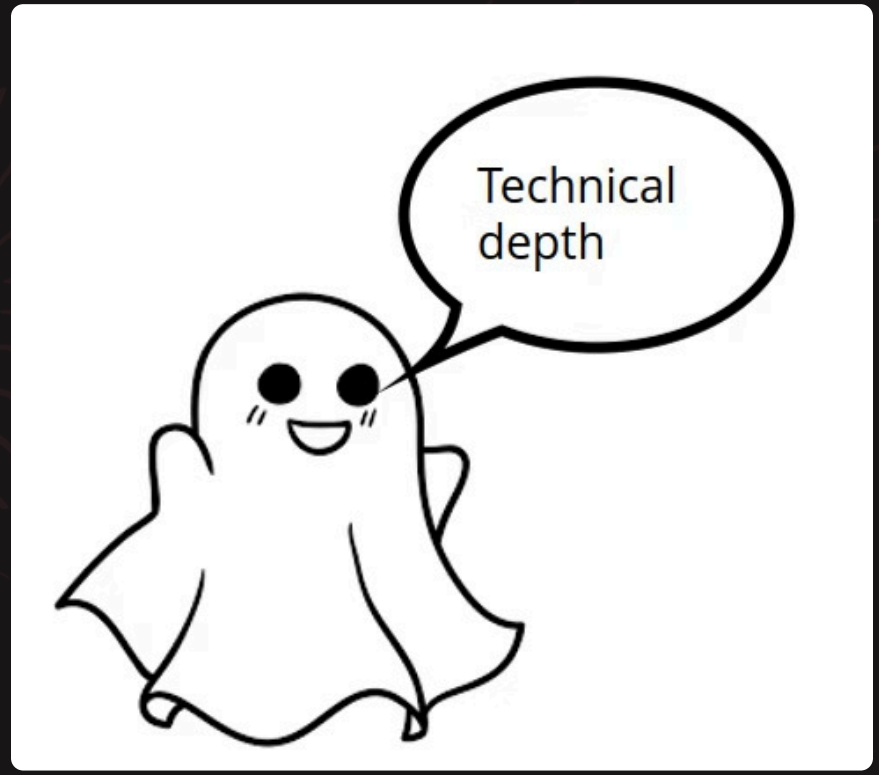
Whenever we need to pick one tool over the other, we need to consider these 4 criteria.

And most time this is not possible without actually trying to use both.

Not picking the right tool leads to Technical Dept

In a nutshell MLOps is about dealing reducing technical debt

⚠️Technical debt is the implied cost of future reworking required when choosing an easy but limited solution instead of a better approach that could take more time



MLOps has and is too tool centric

Why the Focus on Tools?

💡 **Tech-Driven Hype** – MLOps emerged alongside cloud AI services, containerization, and orchestration tools, making it feel like a tooling problem rather than a process and culture problem.

💡 **Vendor Influence** – Companies push their own MLOps stacks, leading to fragmented ecosystems that emphasize tool adoption rather than best practices.

💡 **ML Engineers' Backgrounds** – Many ML practitioners come from research backgrounds and are more familiar with coding than system design

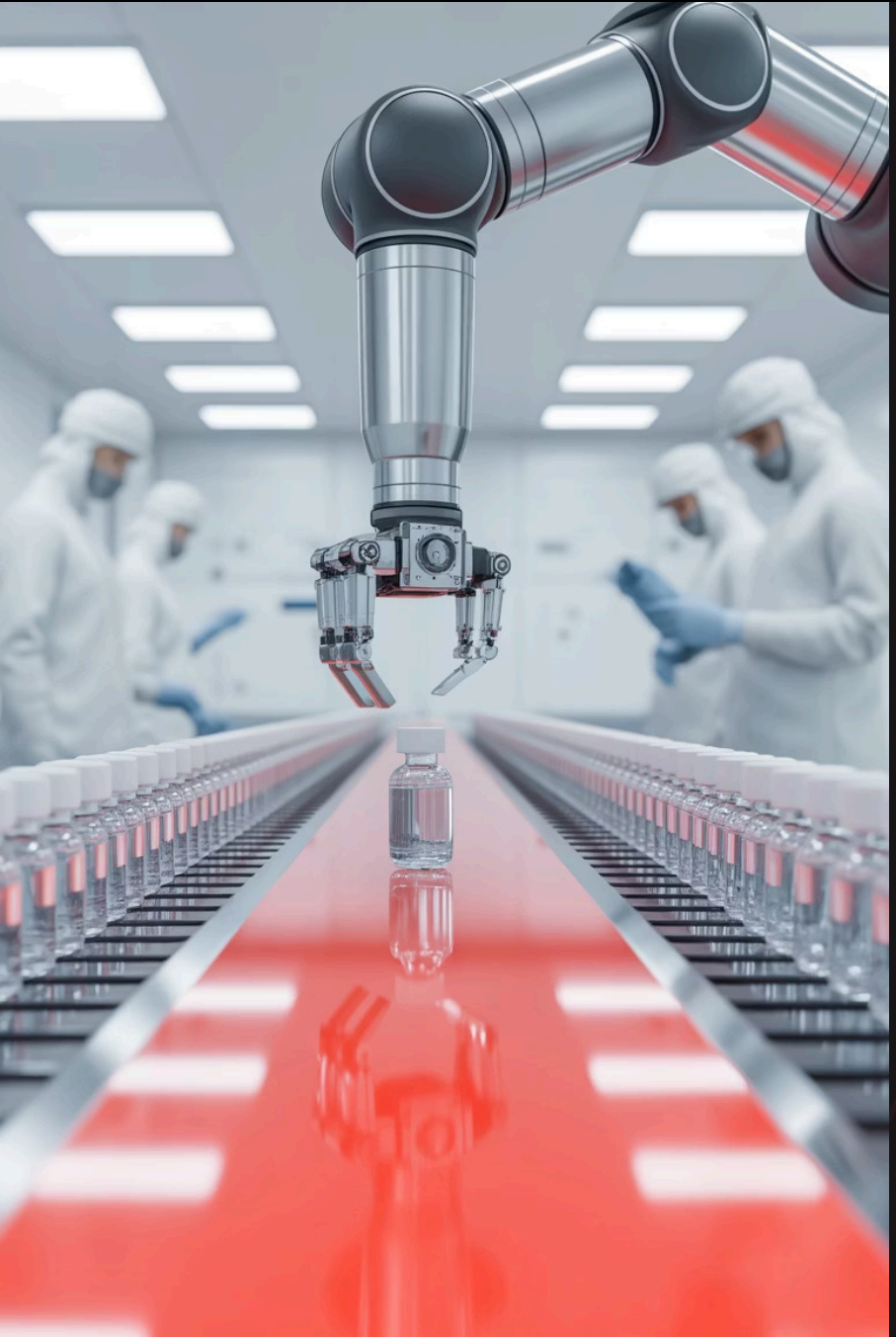
Why Process Matters More

If we **strip away the tools**, the core of MLOps is about establishing **robust workflows** that ensure:

💡 Models are reproducible and traceable (versioning, experiment tracking).

💡 CI/CD practices extend beyond software to **model development** (automated validation, deployment gates).

💡 Models are **continuously monitored and retrained** as data distributions change.



Case Study

Medical Vial Quality Detection

Danish medical company needs automated error detection in pharmaceutical vials

Patient Safety

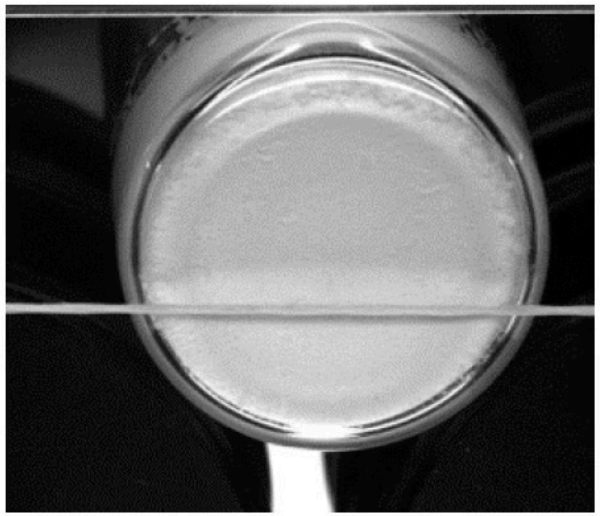
Zero tolerance for defective products reaching patients

Production Efficiency

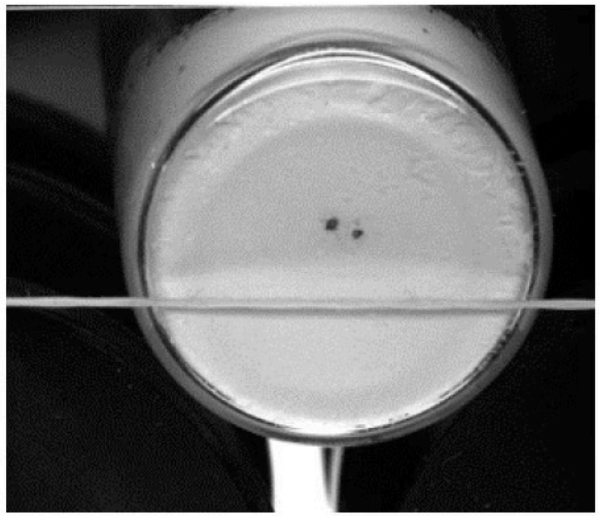
Minimize false rejections to reduce waste

Quality Assurance

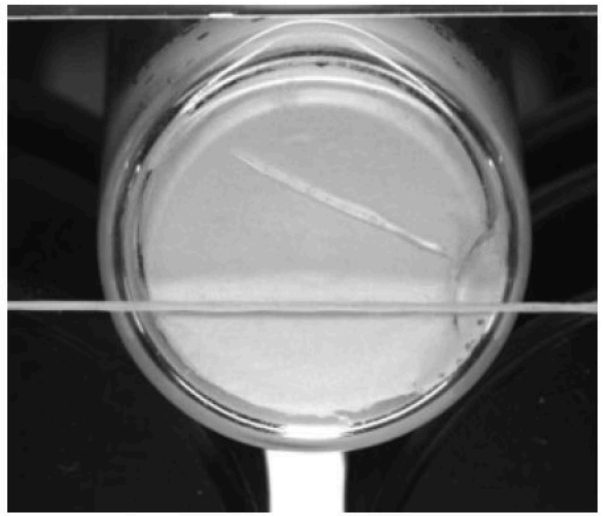
Outperform human inspection capabilities



(a) Good Vial - No Defect



(b) Particle Defect



(c) Chips & Cracks Defect

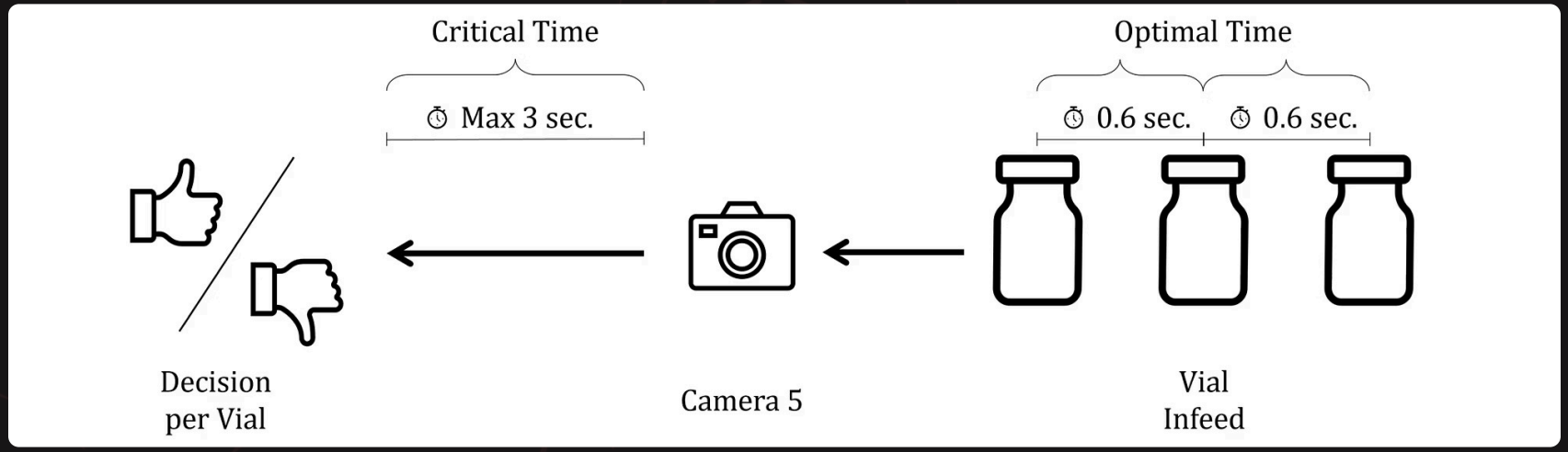
Requirements Specification

Model Requirements

- **Must** outperform human baseline
- **Must** not approve flawed vials
- *Should* minimize false rejections
- *Should* outperform existing model
- *Should* give reason for prediction

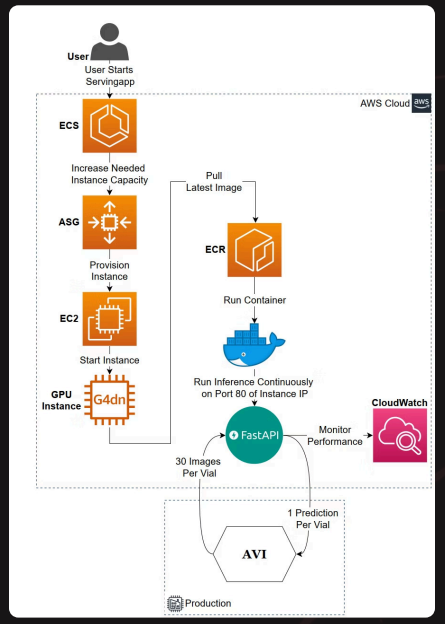
Serving Requirements

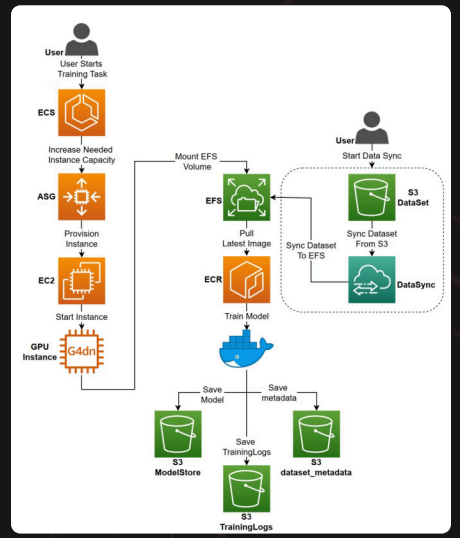
- **Must:** < 3 seconds per vial
- *Should:* < 0.6 seconds per vial

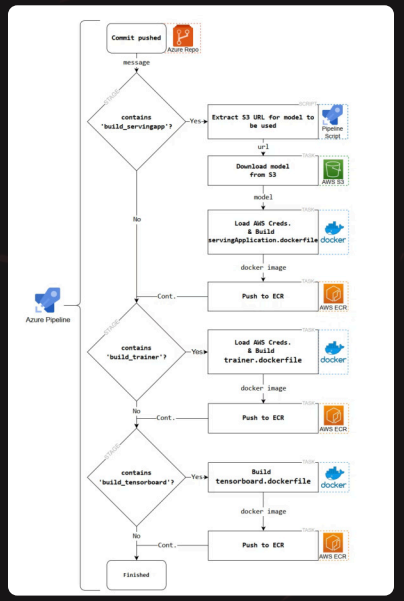


MLOps Pipeline Implementation

1	2	3
Containerization	**Training Pipeline**	**Inference Service**
Reproducible environments across development and production	Automated model training with version control	Scalable model serving infrastructure







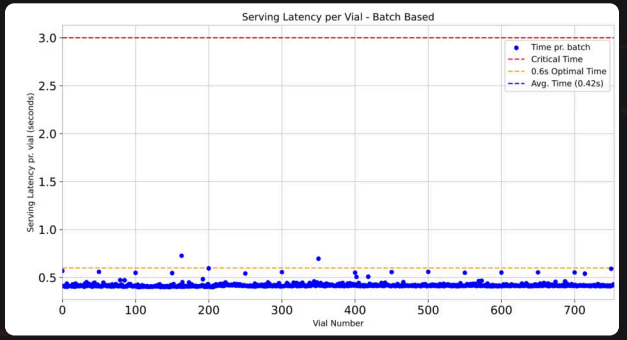
Results: Success with Trade-offs

Model Requirements

- **Must** outperform human baseline ✅
- **Must** not approve flawed vials ✅
- *Should* minimize false rejections ✅
- *Should* outperform existing model ✅
- *Should* give reason for prediction (✅)

Serving Requirements

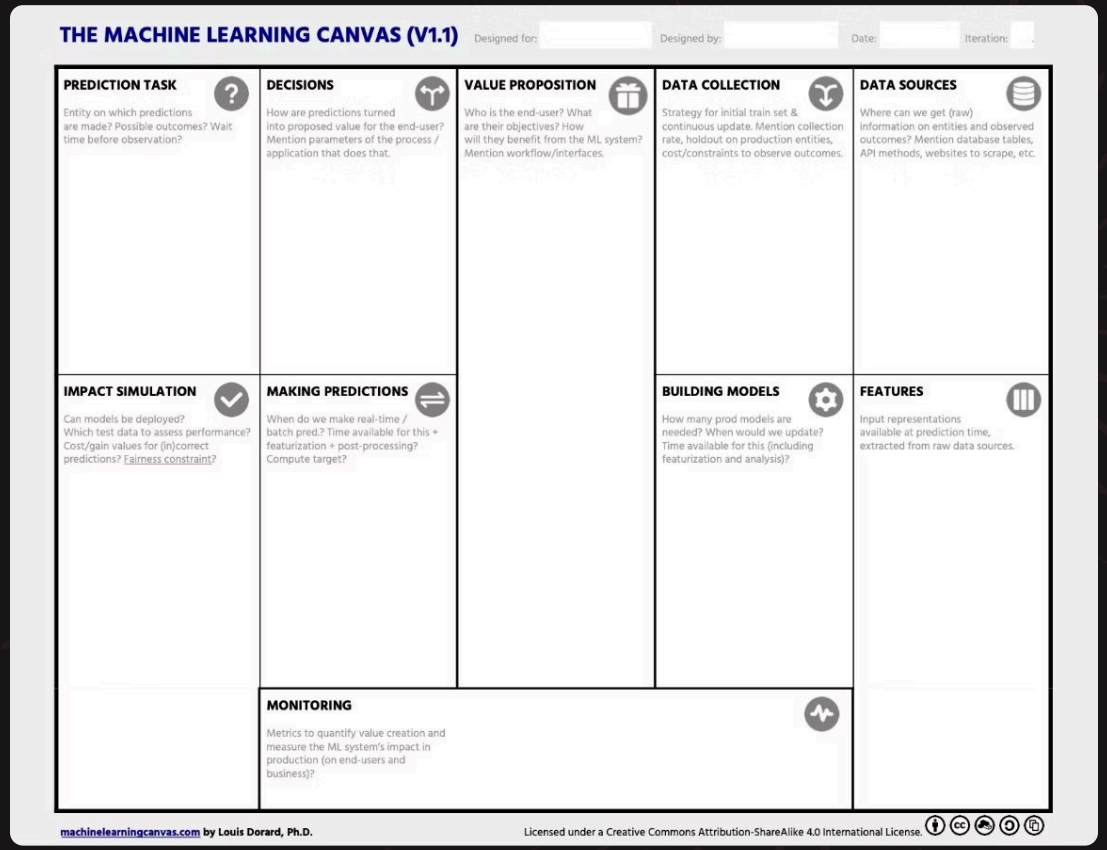
- < 3 seconds per vial ✅
- < 0.6 seconds per vial ❌



Predictions

Ground Truth	NN Model			Efficientnet_b0			
	Good	Particle	CC	Good	Particle	CC	Total
Good	648	13	22	682	0	1	683
Particle	0	17	8	0	25	0	25
CC	10	0	36	0	0	46	46

Summing up, MLOps at its core is...



THE MACHINE LEARNING CANVAS (V1.1)

Designed for: | Designed by: | Date: | Iteration:

PREDICTION TASK
Entity on which predictions are made? Possible outcomes? Wait time before observation?

DECISIONS
How are predictions turned into proposed value for the end-user? Mention parameters of the process / application that does that.

VALUE PROPOSITION
Who is the end-user? What are their objectives? How will they benefit from the ML system? Mention workflow/interfaces.

DATA COLLECTION
Strategy for initial train set & continuous update. Mention collection rate, holdout on production entities, cost/constraints to observe outcomes.

DATA SOURCES
Where can we get (raw) information on entities and observed outcomes? Mention database tables, API methods, websites to scrape, etc.

IMPACT SIMULATION
Can models be deployed? Which test data to assess performance? Cost/gain values for (in)correct predictions? Fairness constraint?

MAKING PREDICTIONS
When do we make real-time / batch pred.? Time available for this + featurization + post-processing? Compute target?

BUILDING MODELS
How many prod models are needed? When would we update? Time available for this (including featurization and analysis)?

FEATURES
Input representations available at prediction time, extracted from raw data sources.

MONITORING
Metrics to quantify value creation and measure the ML system's impact in production (on end-users and business)?

machinelearningcanvas.com by Louis Dorard, Ph.D.

...delivering value for business 💸

...thinking about the hole pipeline, not just data and model 🤔

...accounting for long term goals from the start🥅

Meme of the day

https://skaftenicki.github.io/dtu_mlops/s2_organisation_and_version_control/

1. Git + Github
2. Code structure
3. Data Version Control
4. Command Line Interfaces