

02476 Machine Learning Operations

Nicki Skafte Detlefsen

What is Machine Learning Operations?

Let's start where it all began

Machine learning in production is fantastic

BUT

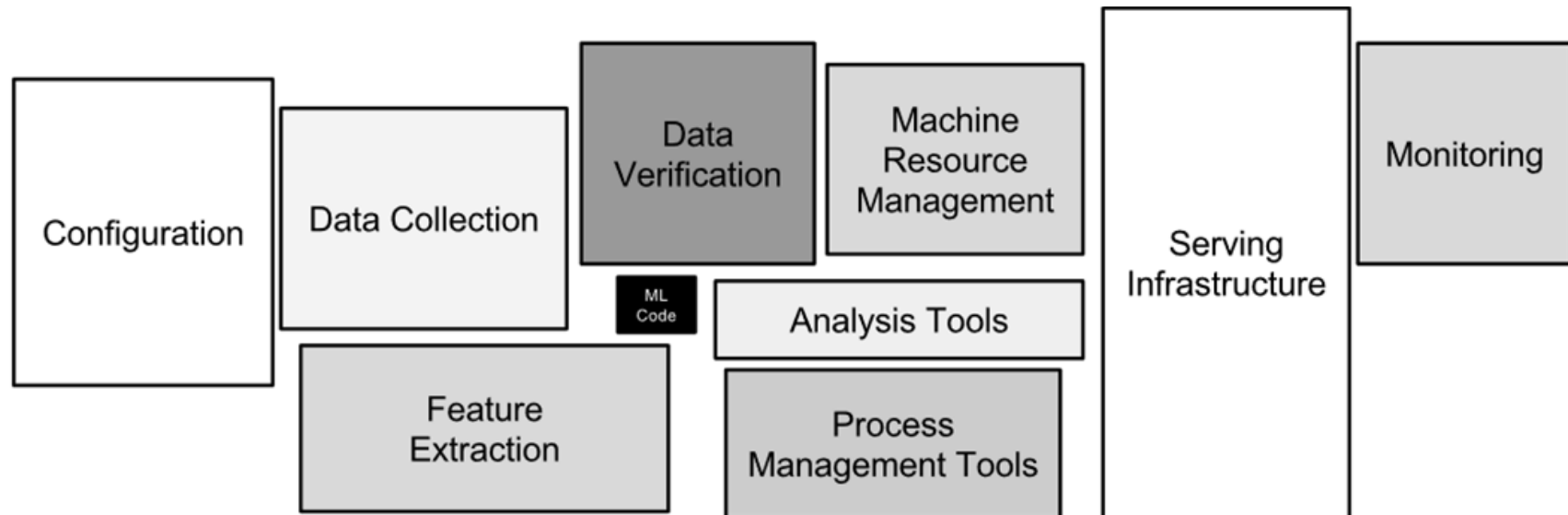


Massive technical depth is incurred if not careful

Hidden Technical Debt in Machine Learning Systems

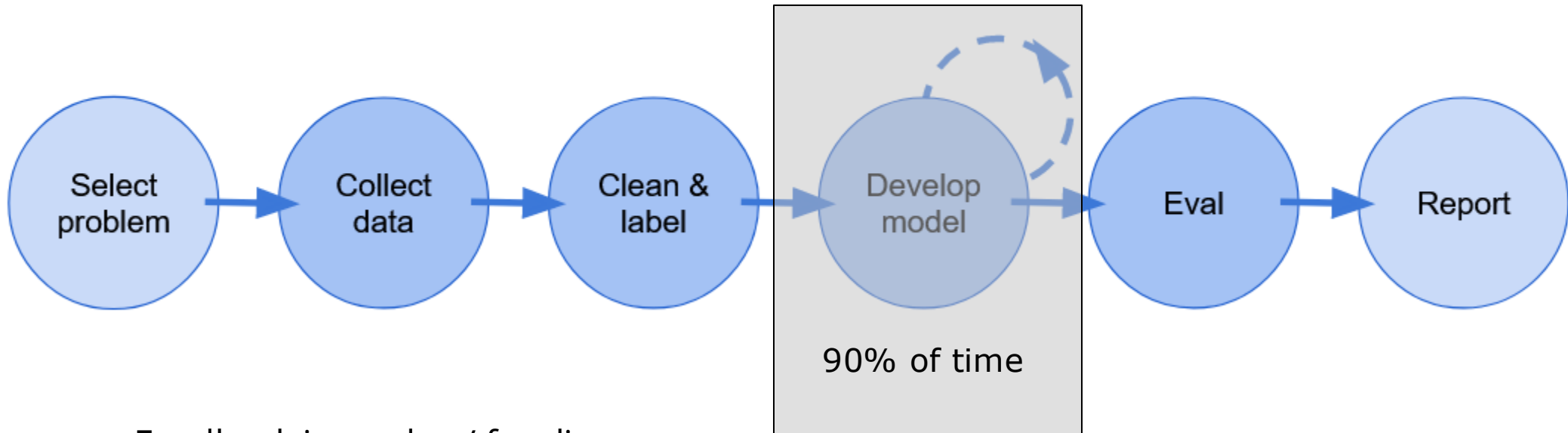
D. Sculley, Gary Holt, Daniel Golovin, Eugene Davydov, Todd Phillips
{dsculley, gholt, dgg, edavydov, toddphillips}@google.com
Google, Inc.

Dietmar Ebner, Vinay Chaudhary, Michael Young, Jean-François Crespo, Dan Dennison
{ebner, vchaudhary, mwyong, jfcrespo, dennison}@google.com
Google, Inc.



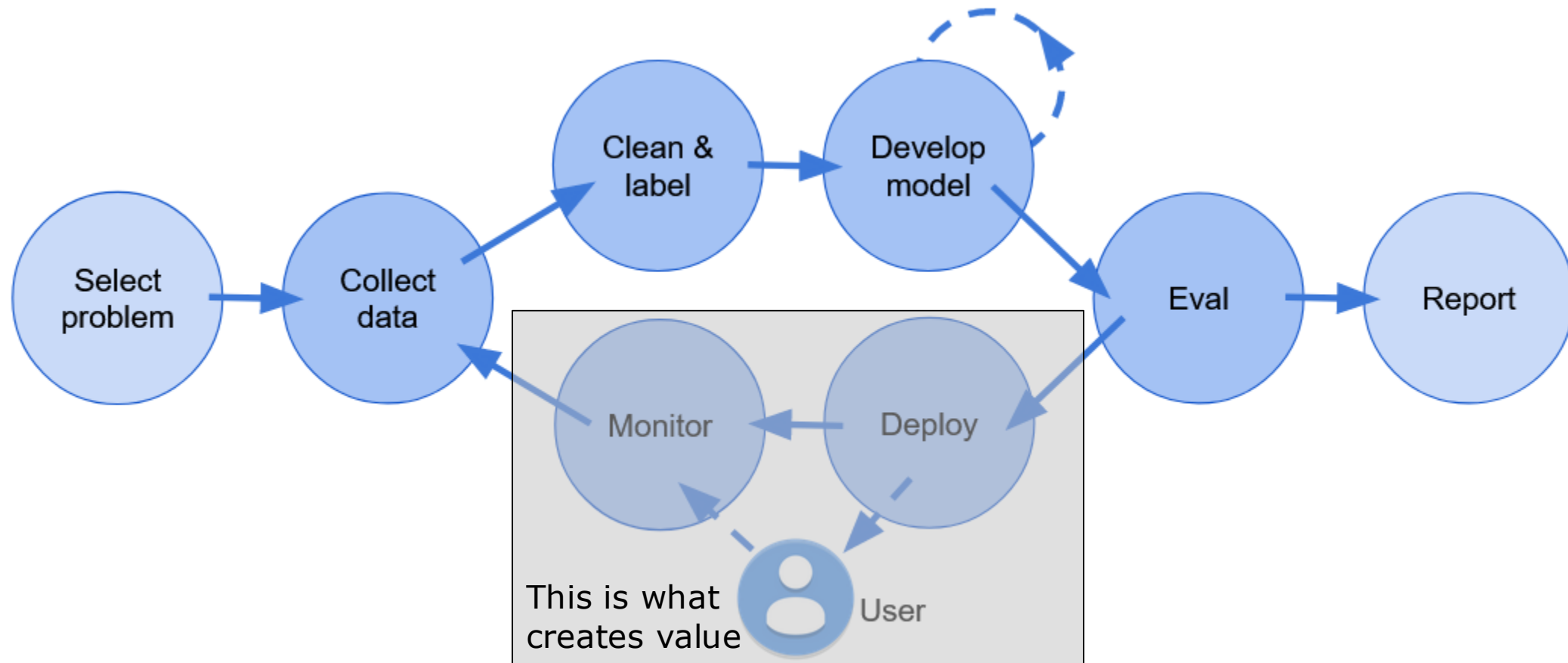
Why do we focus on modelling?

Because we teach people it!
Courses / Projects are linear in nature



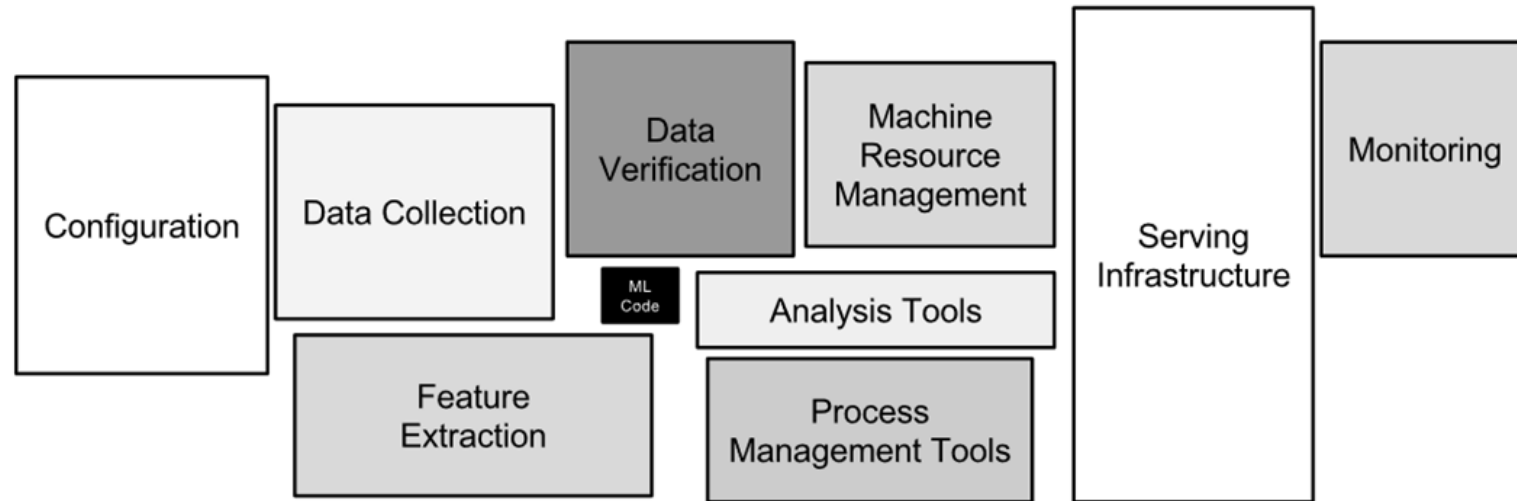
Feedback is grades / funding

Machine Learning in the real world

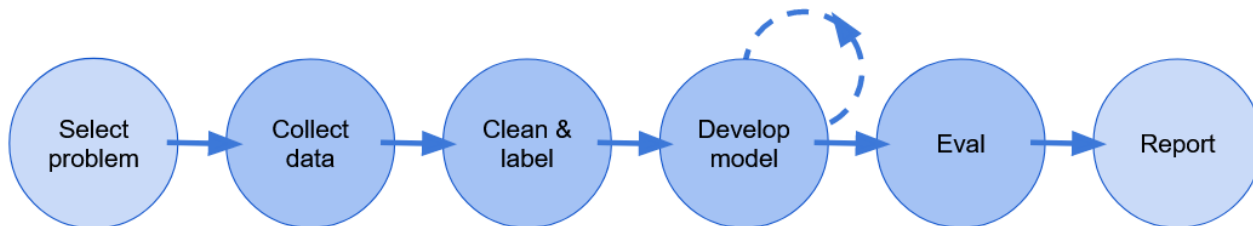


Key observations

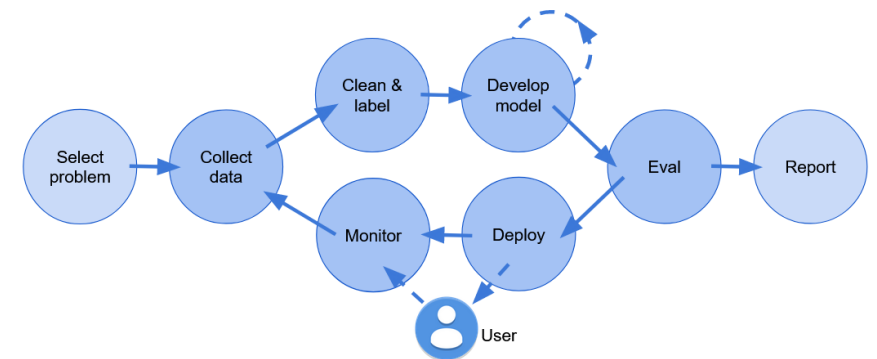
1. Machine Learning in production is much more than doing ML modelling



2. Machine Learning in production is a cycle



VS

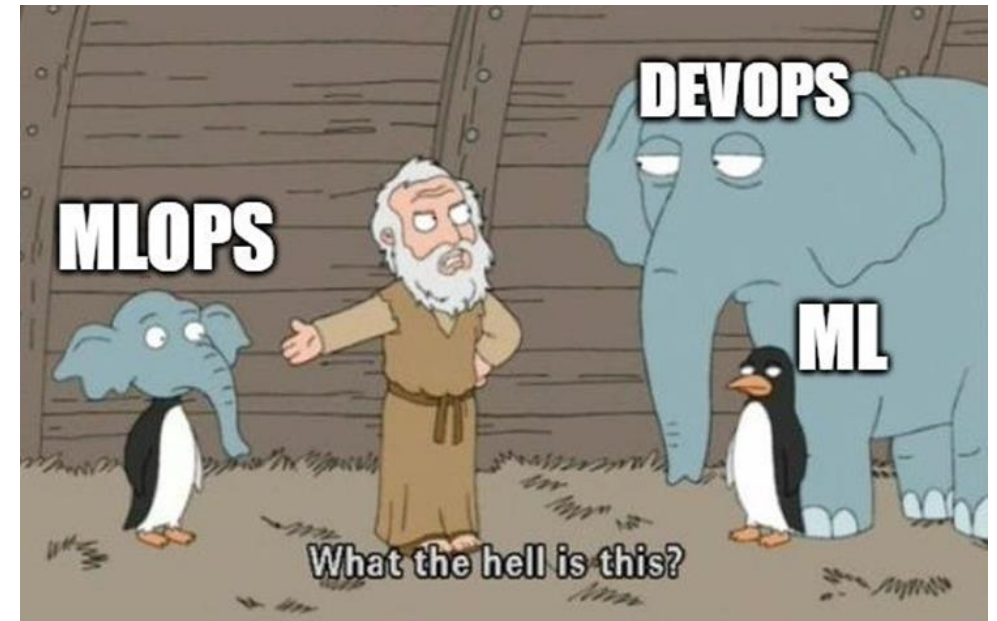


The other stuff is DevOps

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.

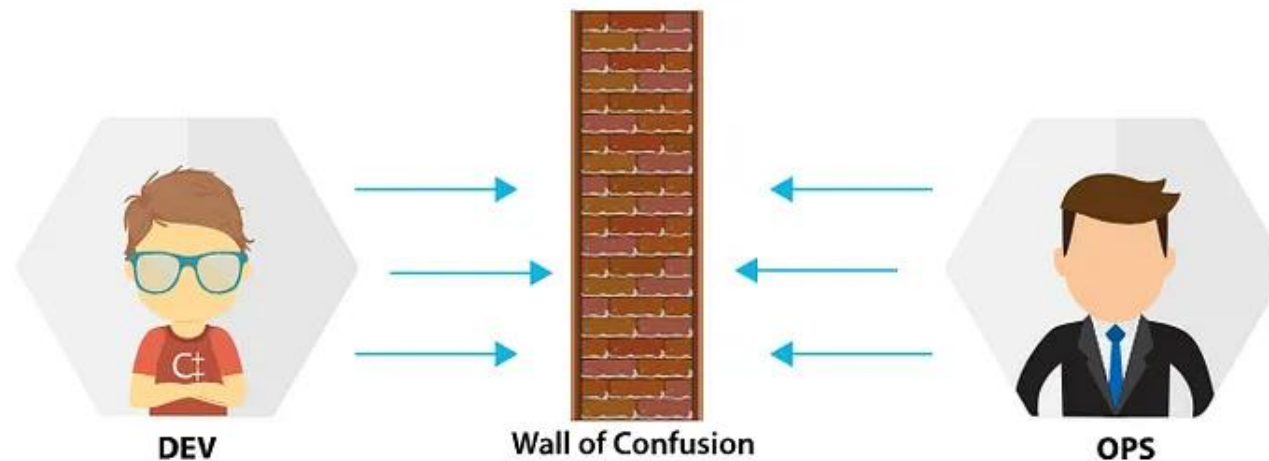


The core problem

There are in general two teams in software development

- 💡 Dev team = development and improvement of software
- 💡 Ops team = infrastructure and operations

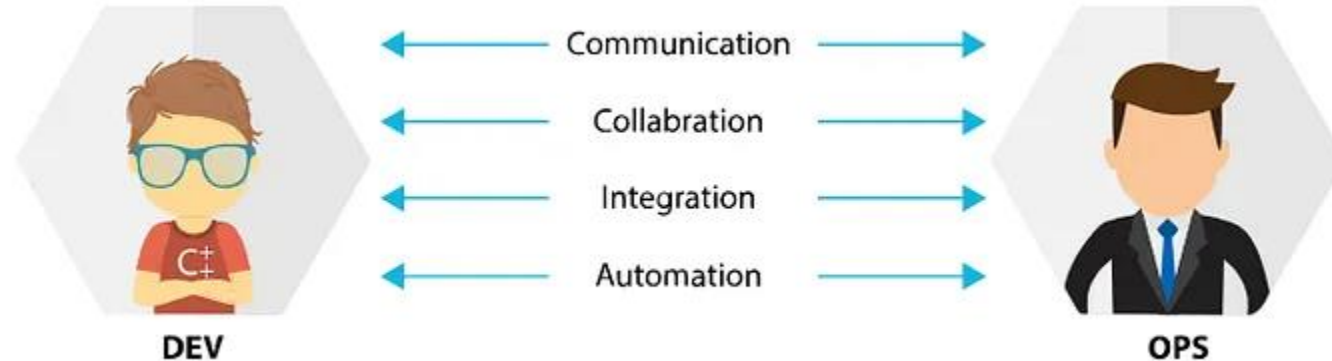
If these two teams does not communicate, then Dev may develop software that Ops cannot operationalize or Ops may setup the wrong infrastructure in relation to what Dev is creating



So, what is DevOps?

This is the closest to a definition that I could find:

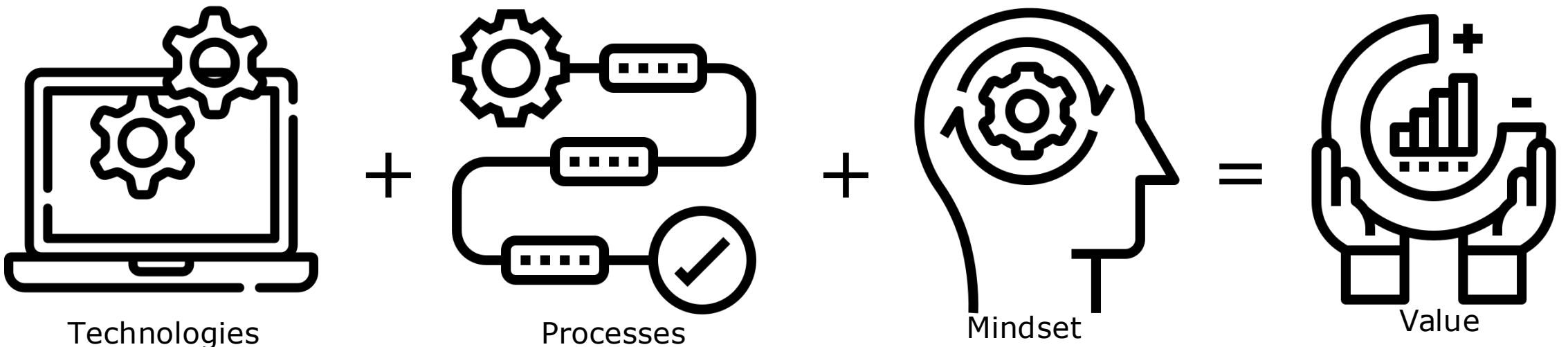
DevOps is a set of practices that combines software development (*Dev*) and IT operations (*Ops*). It aims to shorten the systems development life cycle and provide continuous delivery with high software quality. It's an combination of human mindset, processes and technologies that continuously creates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



So, what is DevOps?

This is the closest to a definition that I could find:

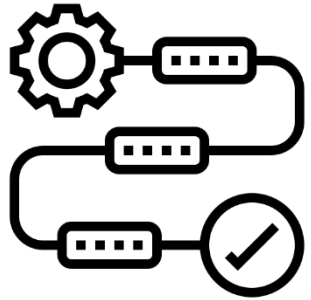
DevOps is a set of practices that combines software development (*Dev*) and IT operations (*Ops*). It aims to shorten the systems development **life cycle** and provide continuous delivery with high software quality. It's an combination of **human mindset**, **processes** and **technologies** that continuously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



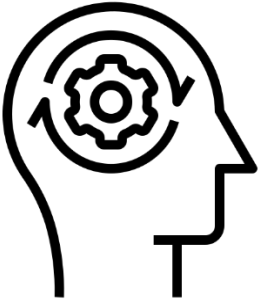
Technology, Processes, Mindset



Use technologies that support the different parts of the lifecycle



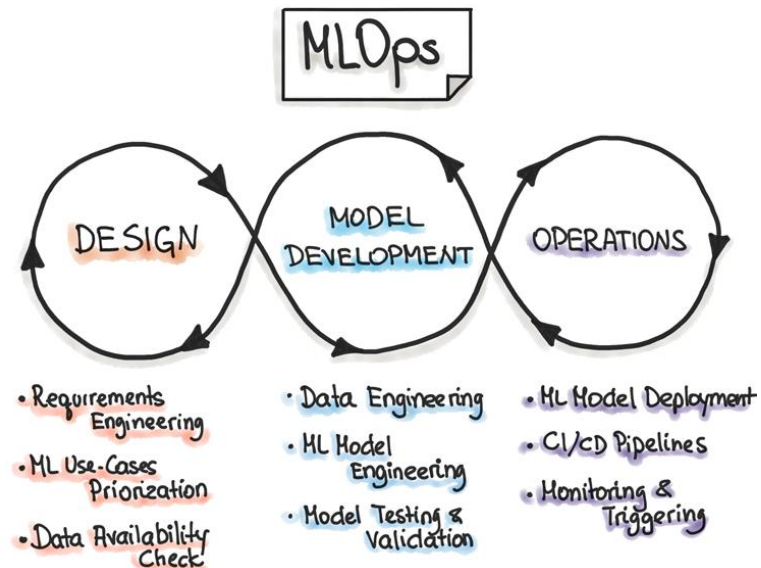
Implement processes to make sure everyone is in sync about the lifecycle



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** reproducible, trackable, testable and maintainable to continuously create value.

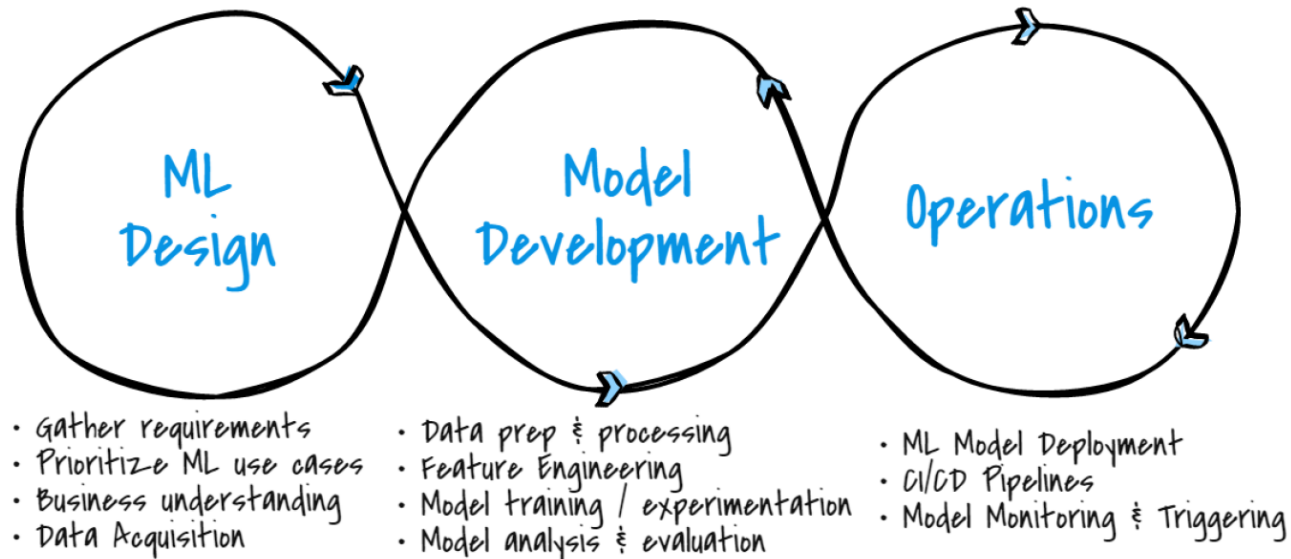


Let's look at the different phases of MLOps

Data phase

- 🔥 Business understanding
- 🔥 Data understanding
- 🔥 Designing the ML-powered software

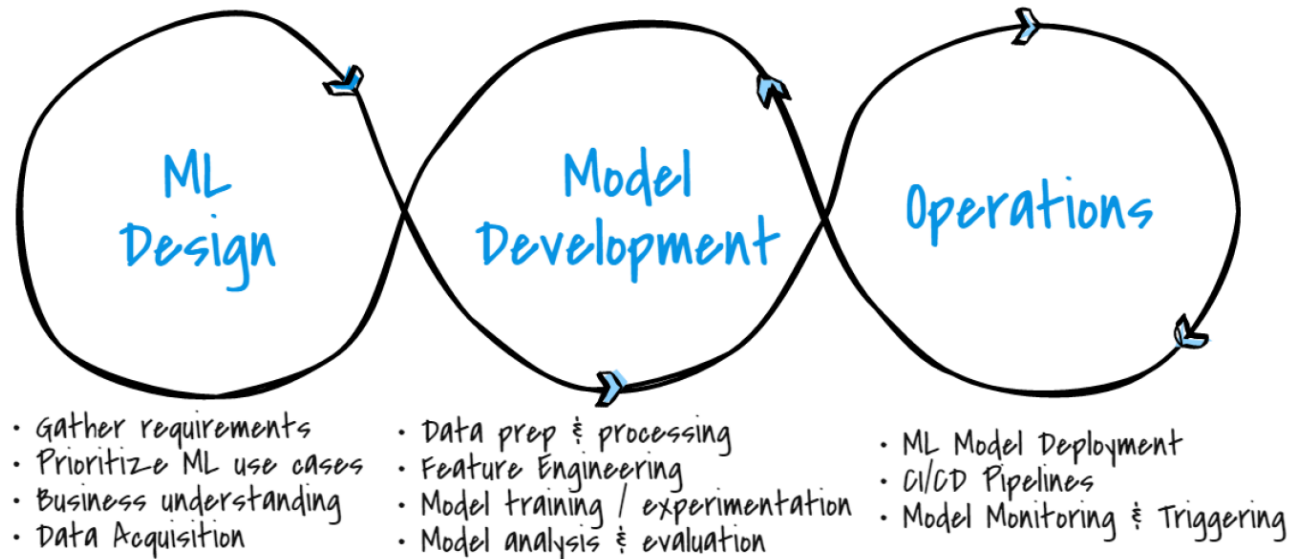
Machine Learning Operations (MLOps)



Model phase

- 🔥 Model engineering
- 🔥 Data engineering
- 🔥 Deliver a stable quality ML model that we will run in production

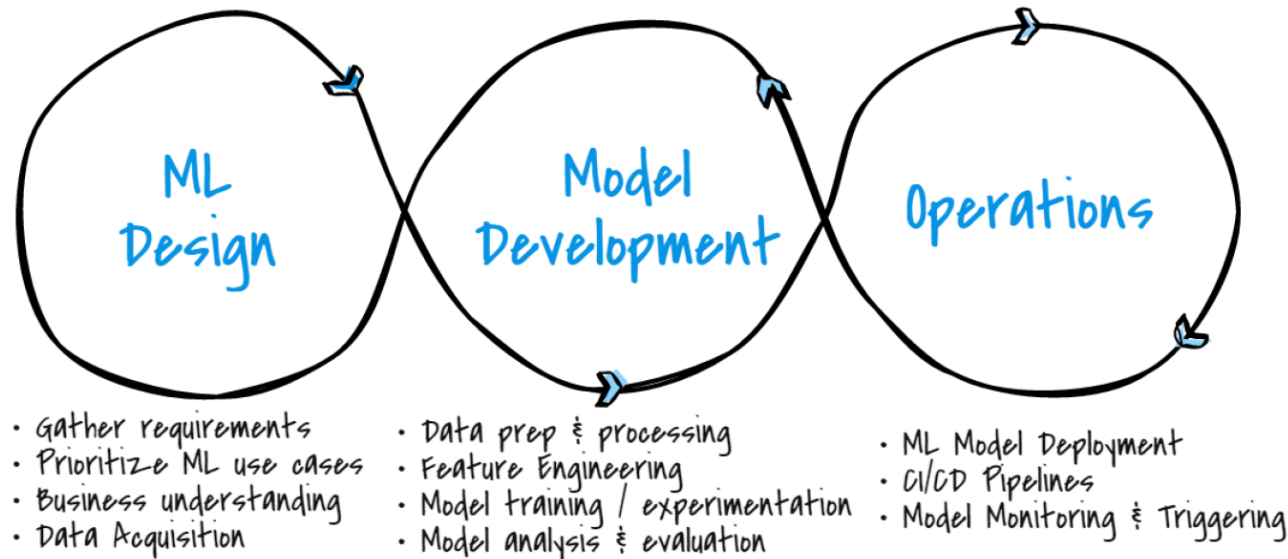
Machine Learning Operations (MLOps)



Operations phase

- 🔥 Deliver the previously developed ML model in production
- 🔥 Testing, versioning, continuous delivery, and monitoring

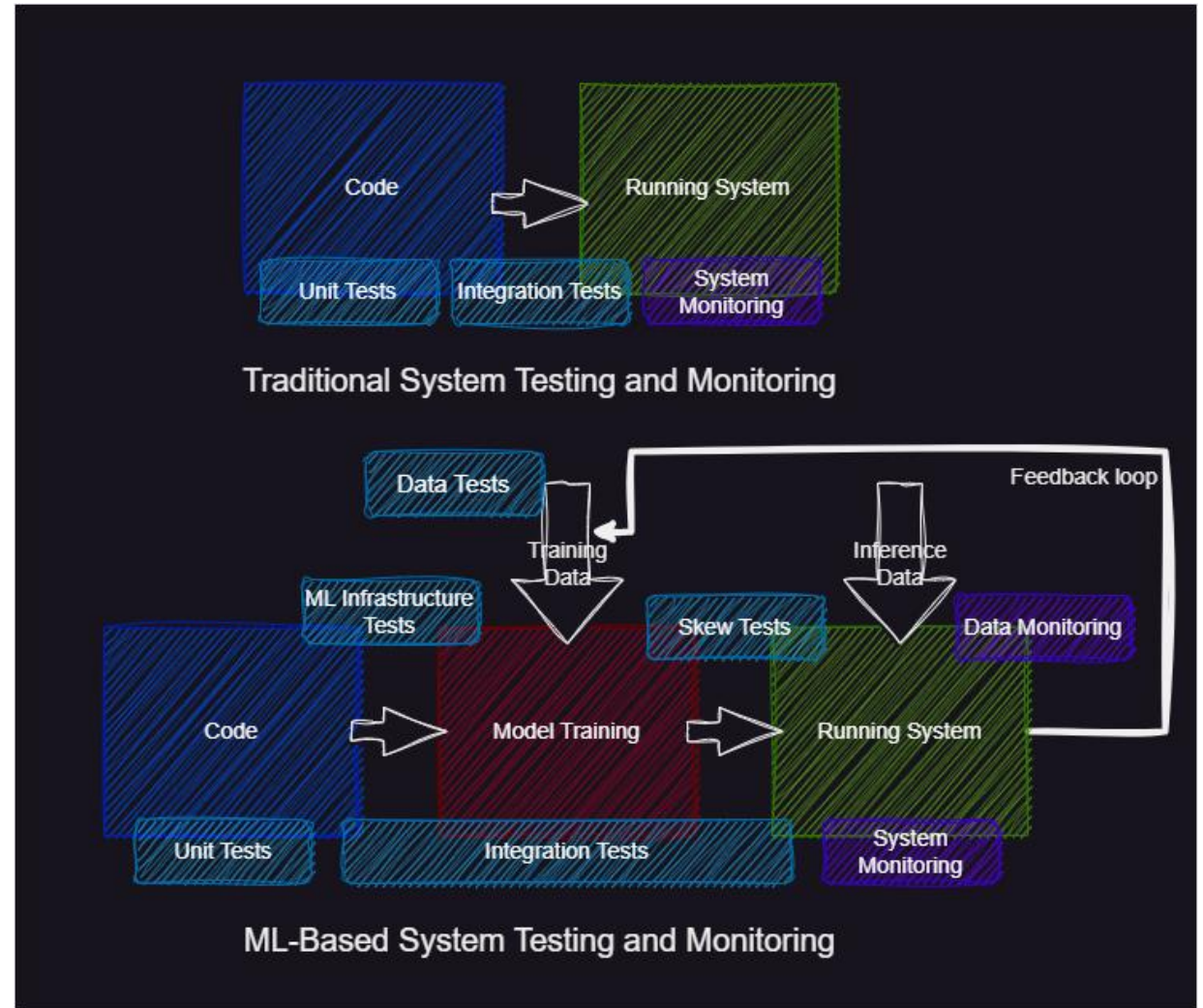
Machine Learning Operations (MLOps)



If DevOps exist, then why do we need MLOps?



Because data changes everything

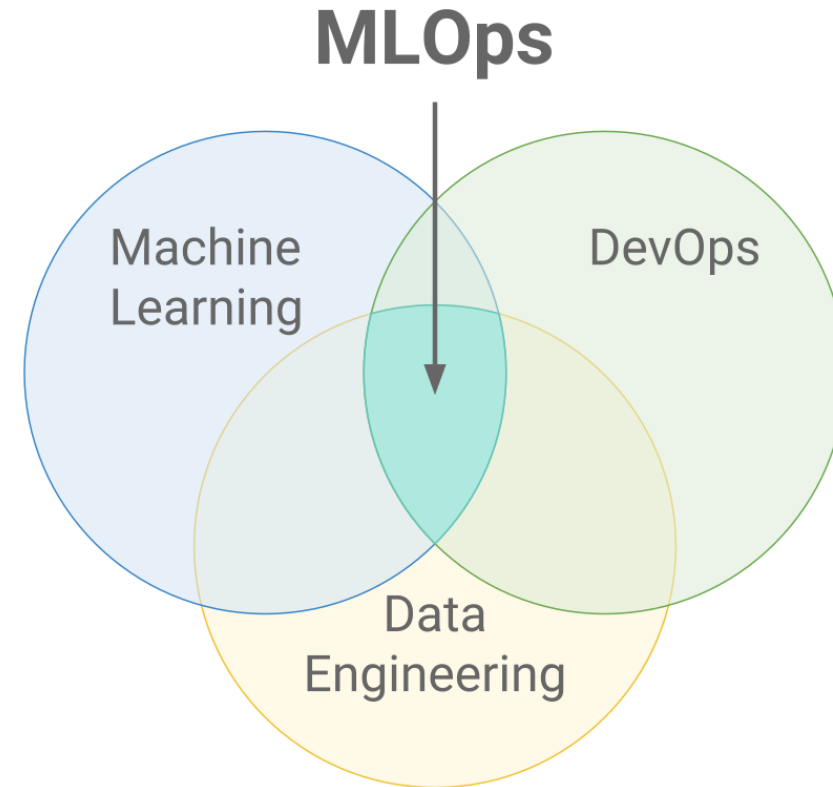


What is an MLOps engineer?

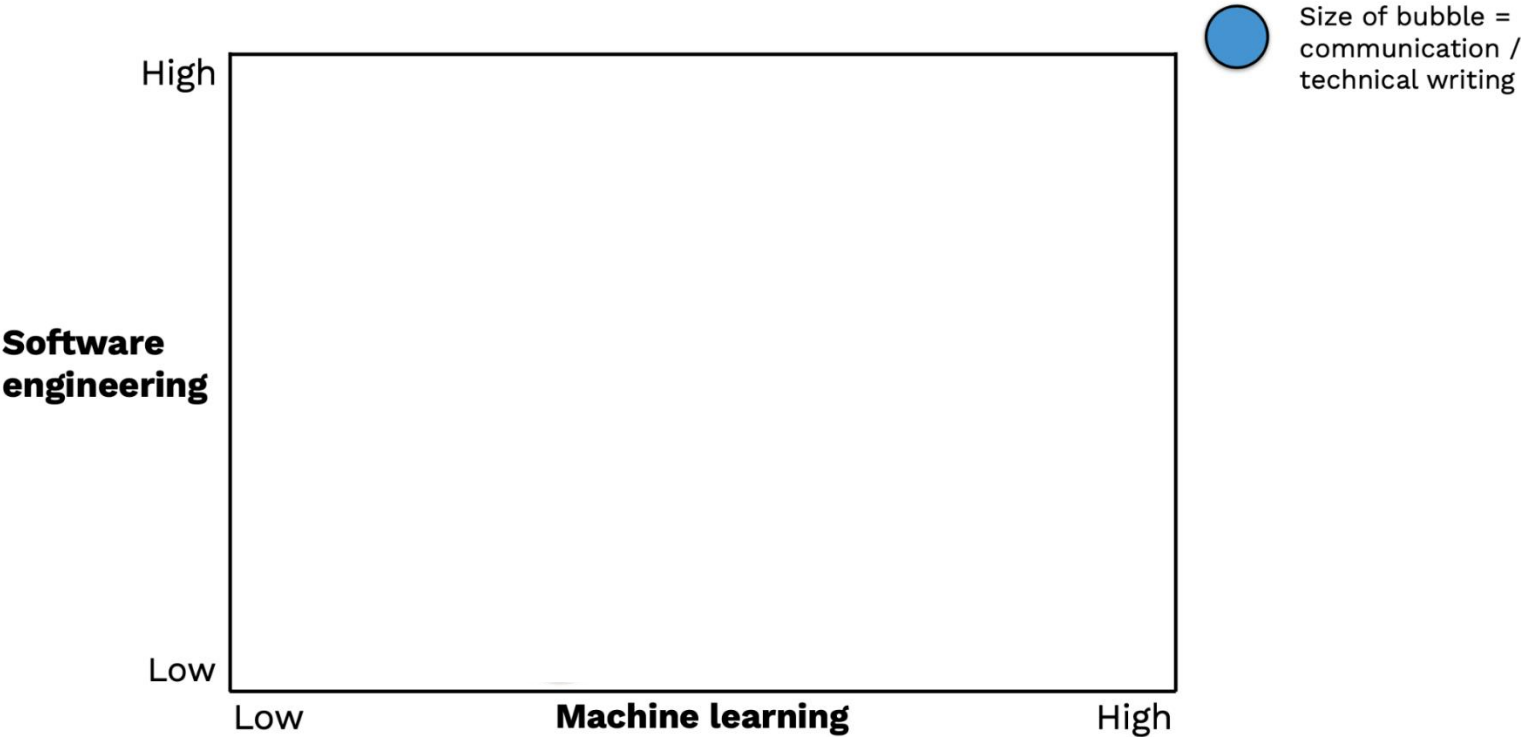
What makes an MLOps engineer?

A mix of

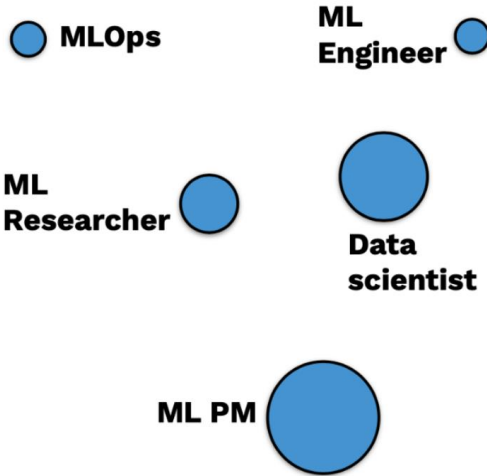
- Software developing
- Machine Learning
- Data engineering



Where's waldo?



Where should the different positions be?



According to stable diffusion

<https://stablediffusionweb.com/>

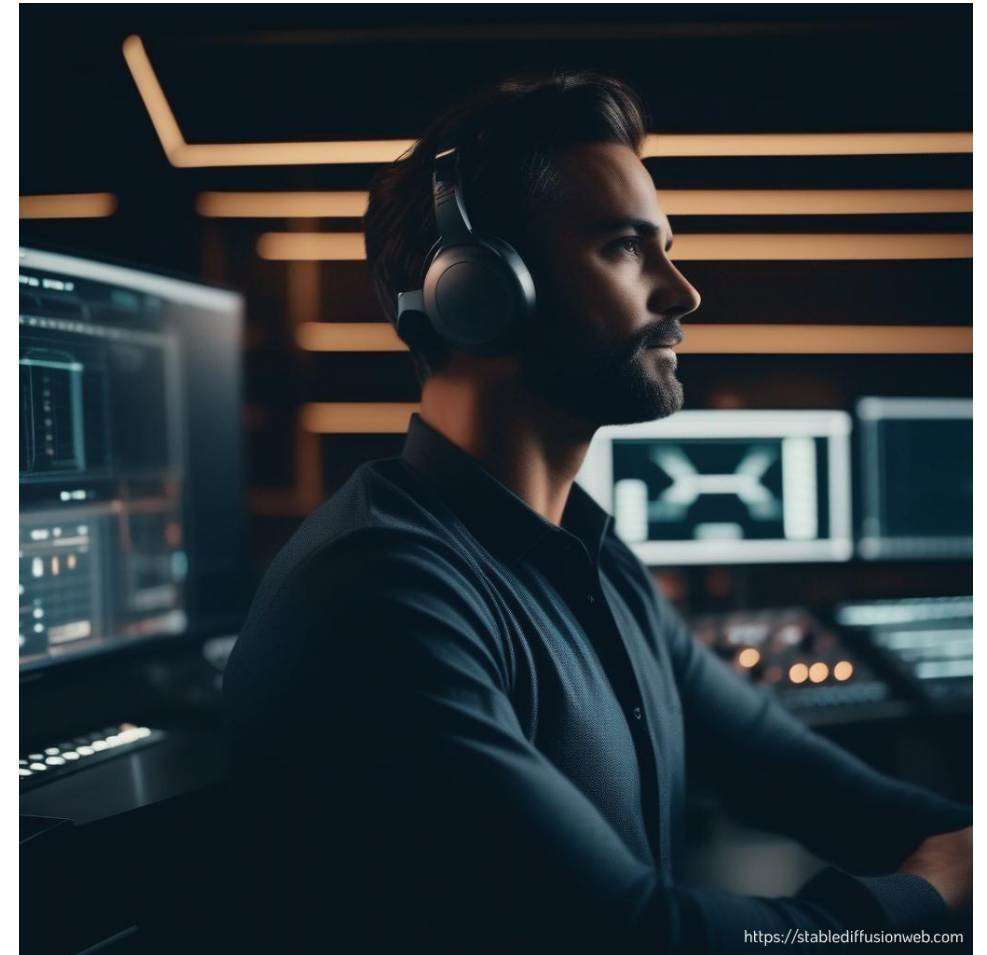
Prompt:

“Machine Learning operations engineer”

A MLOps engineer is

- 🔥 Buff
- 🔥 Locked in
- 🔥 Many screens

Its not completely wrong



<https://stablediffusionweb.com>

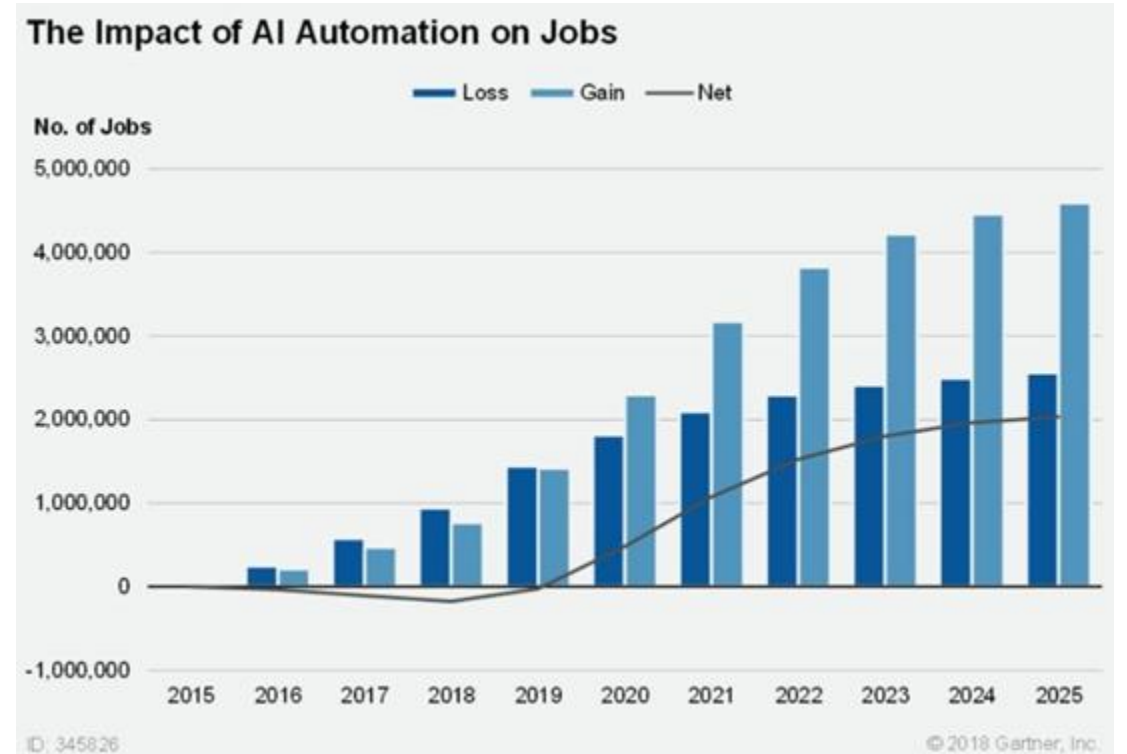
Why is MLOps hot?

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/>

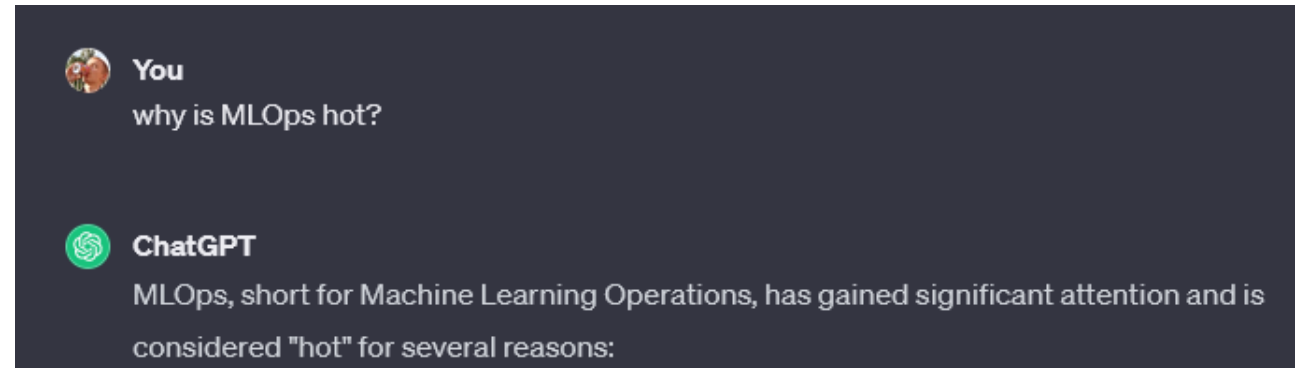
TL;DR:

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.



Kyunghyun cho

Let's ask ChatGPT



1. Growing Adoption of Machine Learning (ML)
2. Complexity of ML Workflow
3. Bridge between Development and Operations
4. Need for Collaboration
5. Ensuring Model Governance and Compliance
6. Automation and Scalability
7. Continuous Integration and Continuous Deployment (CI/CD)
8. Infrastructure Orchestration
9. Adoption of Cloud Services
10. Business Impact

Open AI study

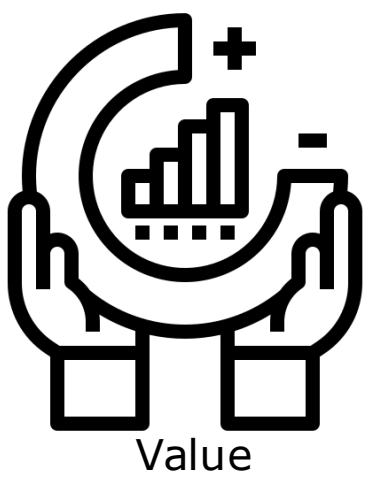
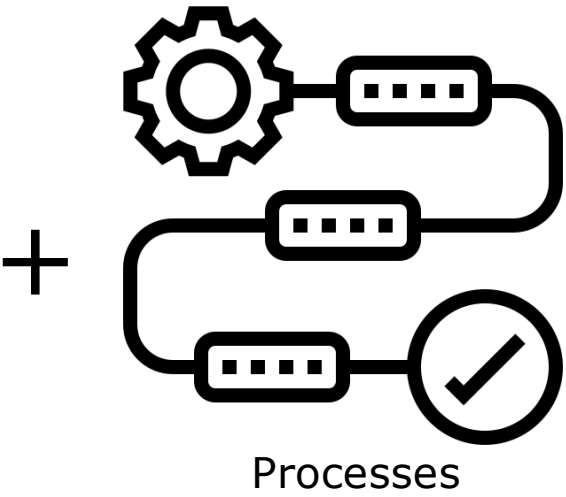
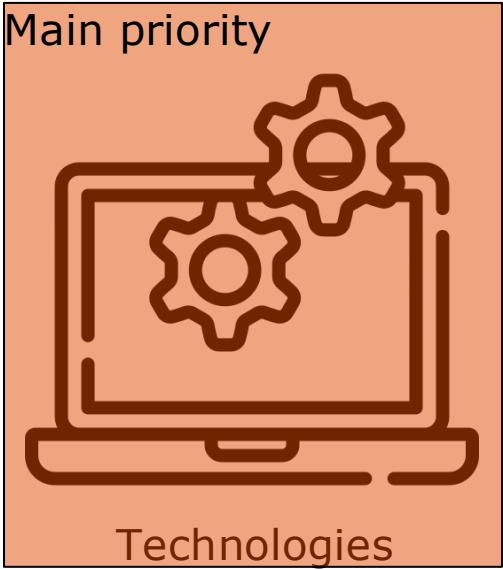
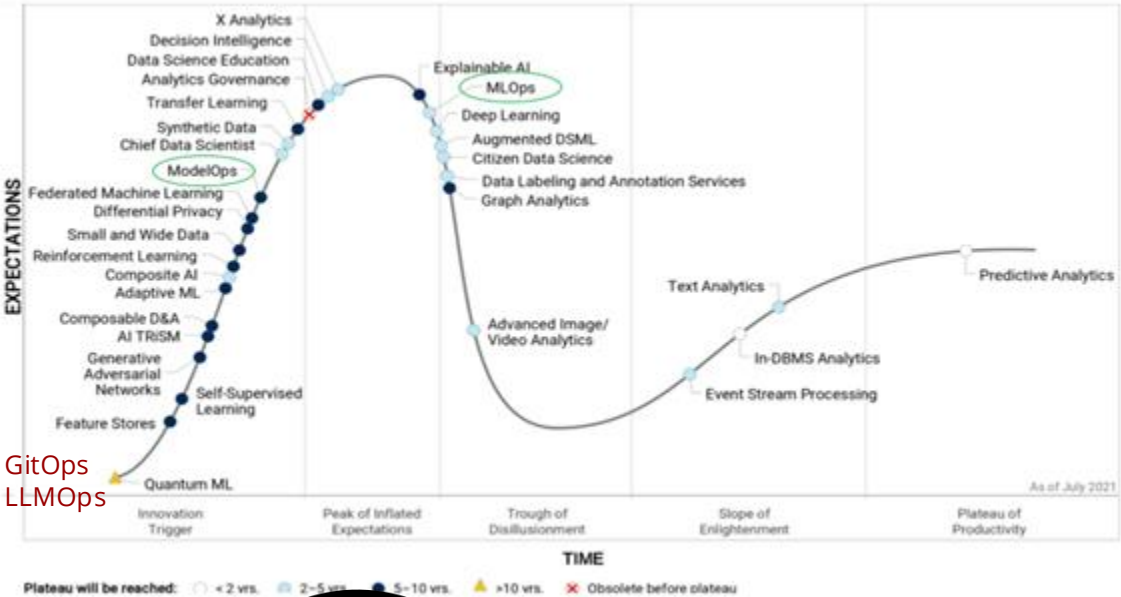
🔍 What is the contributes to the success of OpenAI?

- 💡 Funding
- 💡 Data
- 💡 People
- 💡 Compute
- 💡 Service contract with Microsoft

Microsoft fired 10,000 workers in the same breath as the invested \$10B in OpenAI

Trends in MLOps

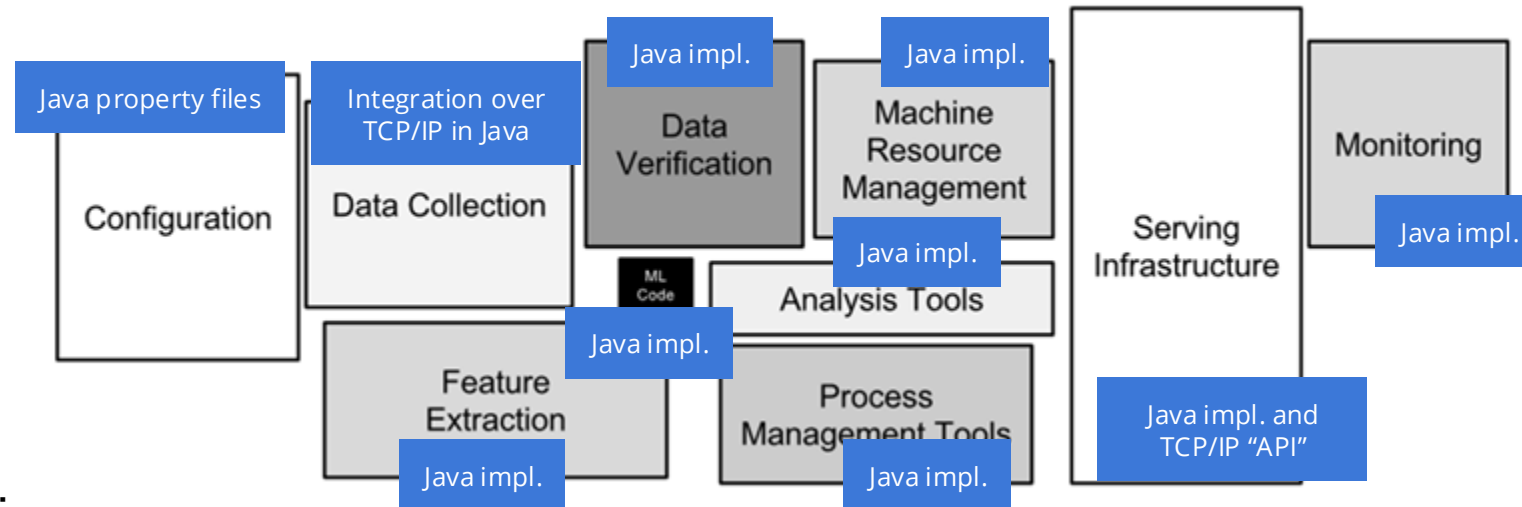
MLOps has been trending for a couple of
Tools have been the main priority



Choosing the right tool for the job

Looking back

MLOps around 2006 = write everything from scratch



Pros:

+ Full control

Cons:

- Slow to iterate
- Hard to maintain
- Lot of manpower per project

Today we have options

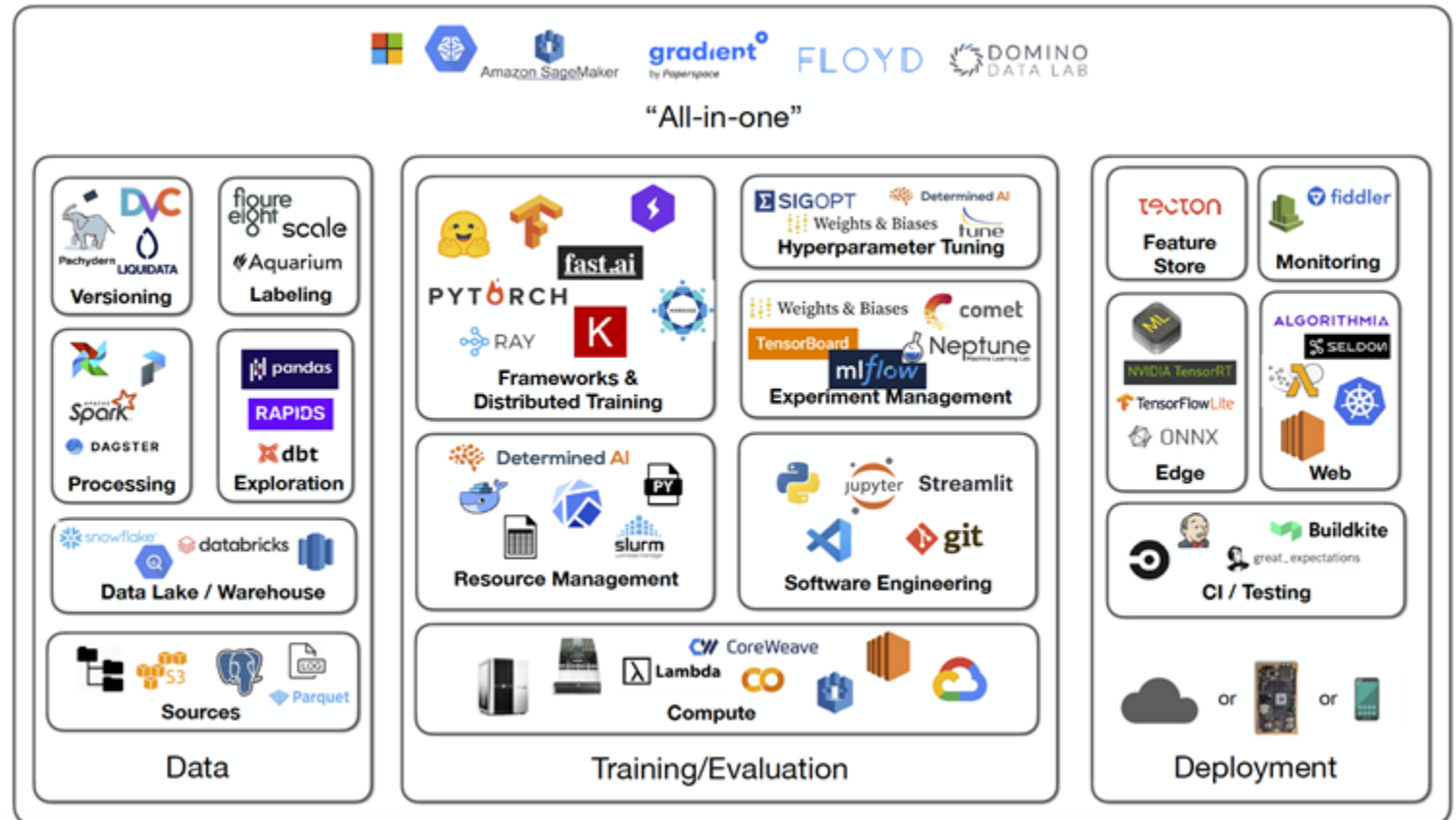
There is a tool for everything you need

Pros:

- + Easy to get started
- + Easy to iterate

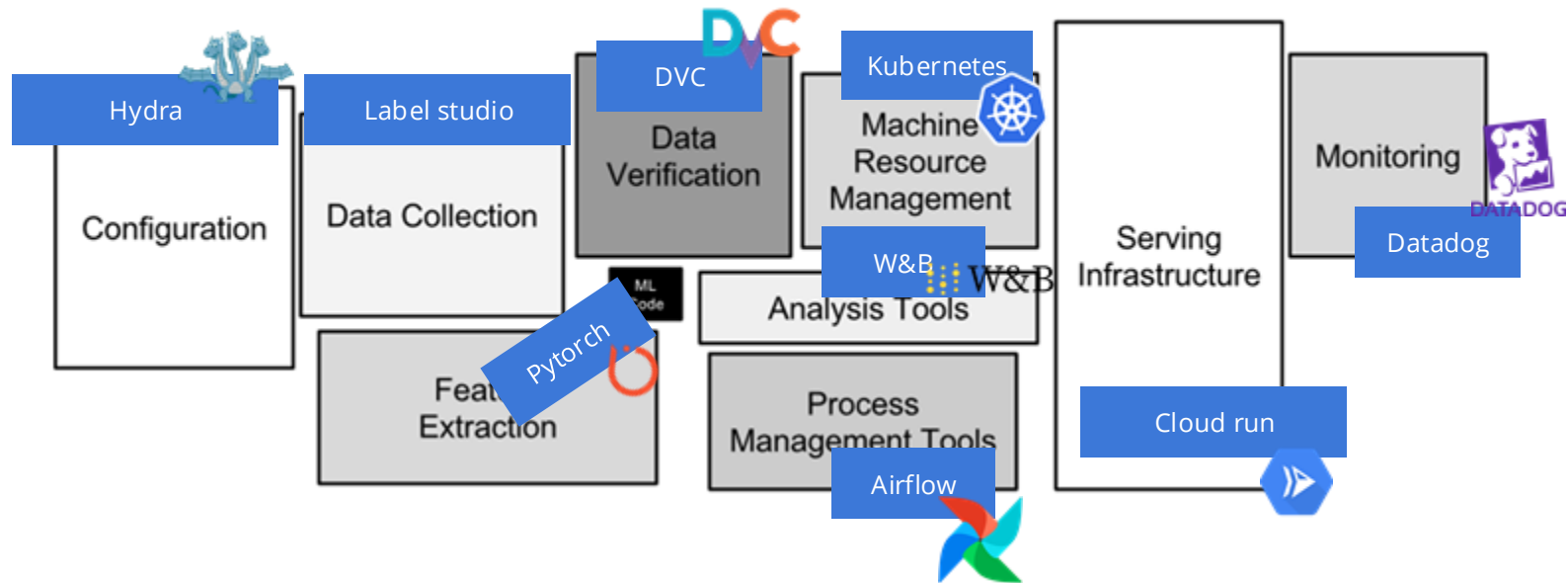
Cons:

- Framework integration can be really hard
- Hard to compare frameworks



MLOps now

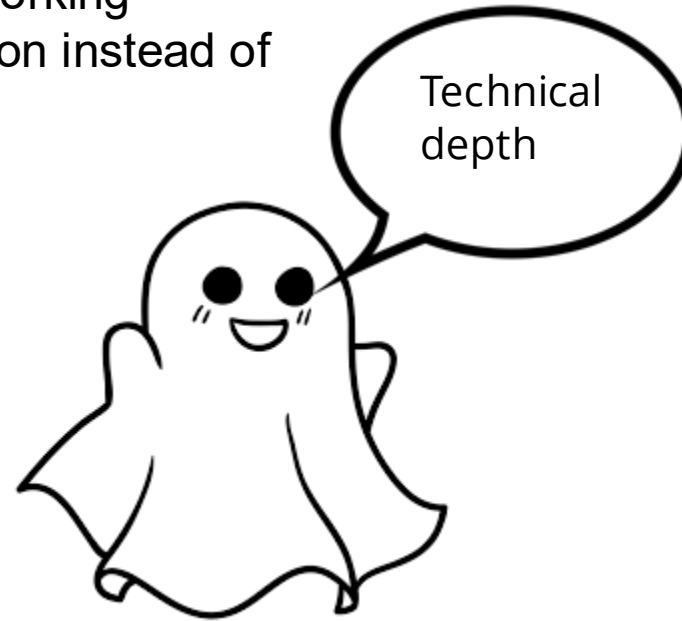
Pick a *stack* of tools



Not picking the right tool leads to TD

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 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.

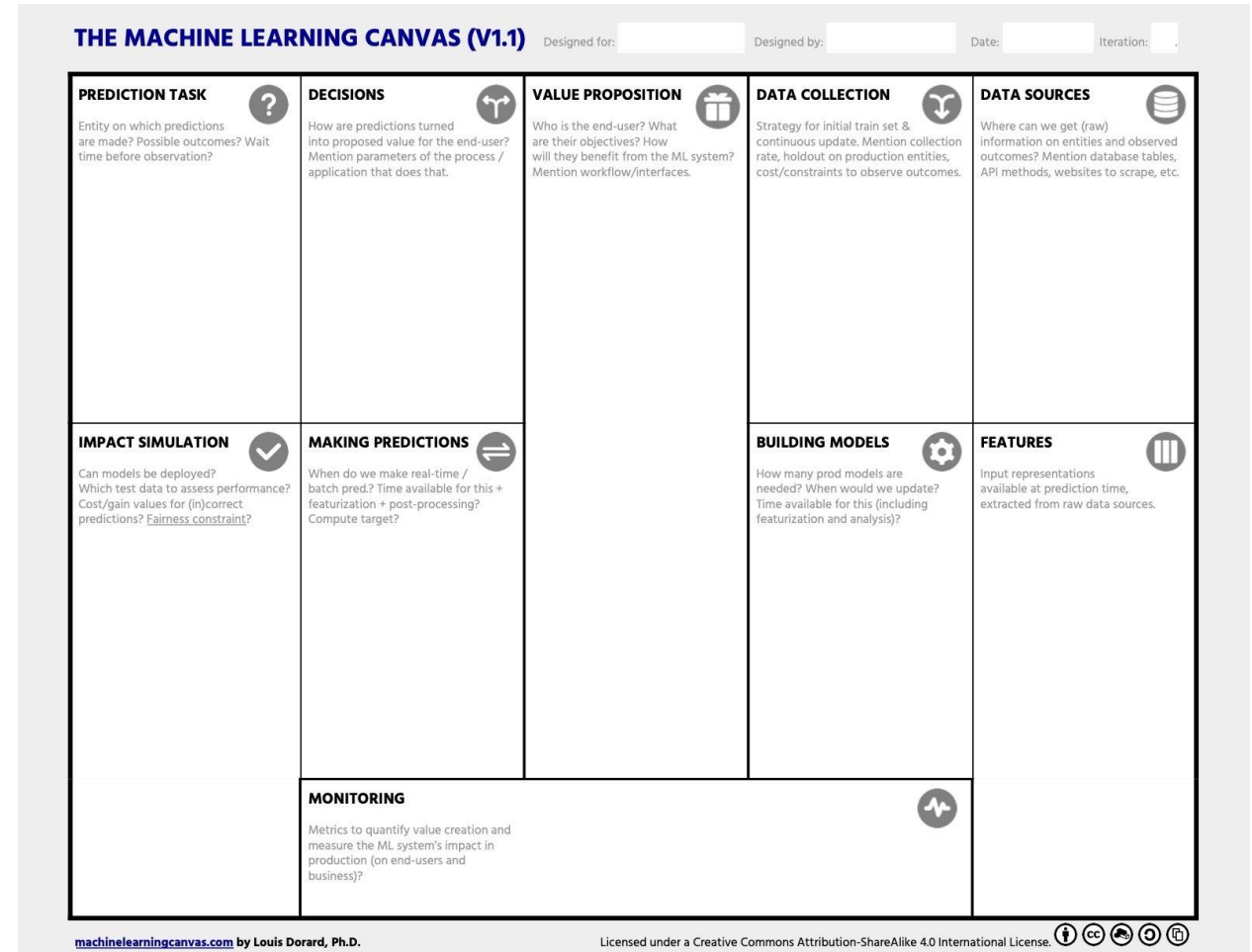


MLOps at its core is...

...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

