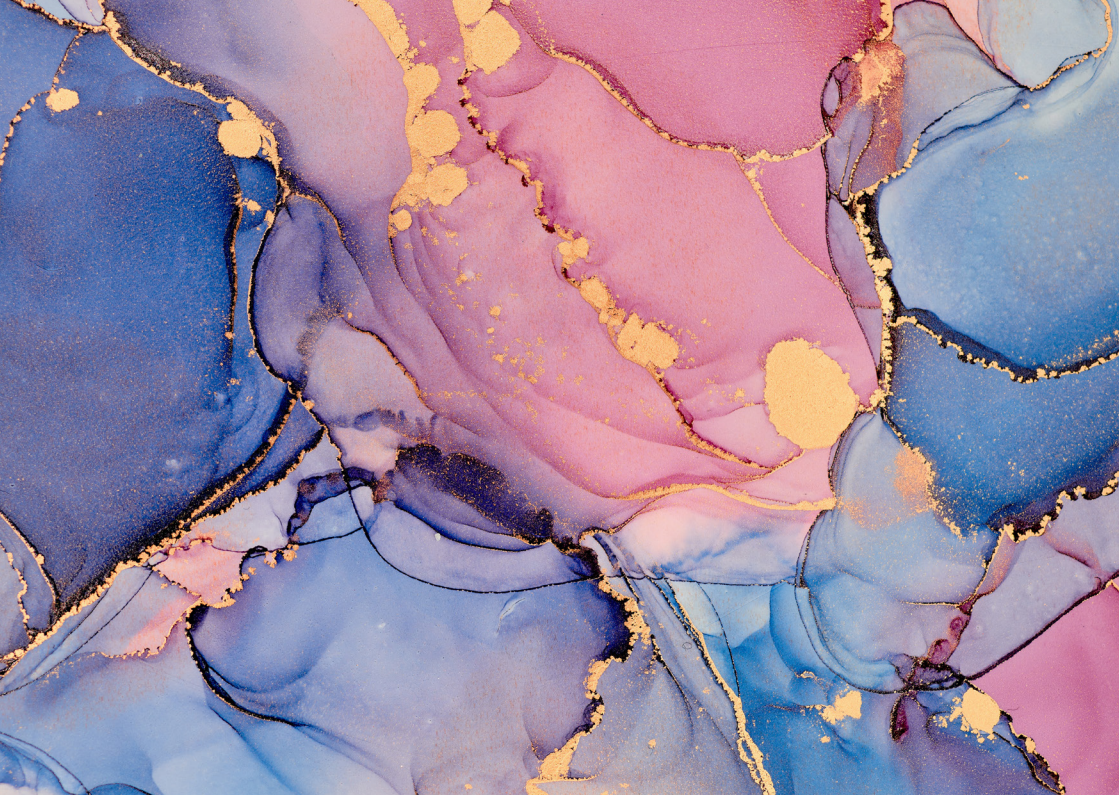


How to get MLOps right

Tackling the complexity of building and deploying machine learning in your organization

thoughtworks | aws

There isn't a single way to do MLOps right, but there are many ways to do it wrong

About this ebook: This ebook walks through the concept of MLOps along with the challenges and opportunities it presents. It also describes an approach developed by Thoughtworks to successfully implement MLOps using Continuous Delivery for Machine Learning (CD4ML). Through a variety of tools and vendor solutions, Amazon Web Services (AWS) can help you succeed with MLOps.

It's a long road to ML models in production

Data science teams across organizations have shown that machine learning (ML) can offer many ways to improve efficiency, automate processes, reduce costs, and augment customer experience. But realizing these benefits in day-to-day operations by deploying and integrating ML models into IT infrastructure is a very different story. Real-world data, on which ML models are trained, changes rapidly. And in most

87%

of data science projects never make it into production

—VentureBeat[1]

[1]VentureBeat, Why do 87% of data science projects never make it into production?

organizations, it takes between four months to a year to launch their first ML minimum viable product (MVP), according to the Harvard Business Review[2]. Bringing ML models into production is different and more complex than deploying software. But tools and processes in professional software development can certainly help.

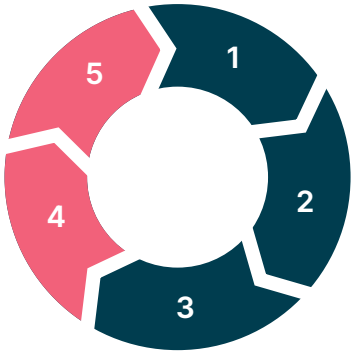
> **"The big story in infrastructure and operations will be learning to put machine learning products into production. ML and AI raise challenges that few ops teams have faced."**
>
> **—Mike Loukides, O'Reilly[3]**

[2]Harvard Business Review, How to Choose Your First AI Project
[3]O'Reilly, Radar trends to watch: January 2020

Deploying ML models can take longer than you think

ML and artificial intelligence (AI) have tremendous power to transform organizations—beyond helping humans make better decisions and processes more efficient. Those possibilities can only be realized after ML models are deployed into production. To bring them into production, datasets must first be captured, stored, cleaned, and curated to extract insights and train ML models. And once in production, new data can be used to re-train and improve models. This process is known as the cycle of intelligence.

Cycle of intelligence for ML models



Data Science

1. **Data:** Acquire
   Values attributed to parameters.
2. **Information:** Store, clean, curate, featurize
   Data that has meaning and is fit for consumption and analysis.
3. **Insight:** Model
   Understanding, predicting, classifying and detecting.

Data Engineering

4. **Decision:** Productionalize
   Planning and prioritizing actions. Hypothesis testing.
5. **Action:** Execute
   Changing the real world!

According to a 2021 study from Algorithmia, the time required to deploy a model is increasing. In 2020, 64 percent of organizations took one month or longer to deploy[4]. This means that for many organizations, doing one iteration around the cycle of intelligence can take a very long time.

MLOps can help you shorten that feedback cycle. Whether you need to rollback a problematic model or release a new model quickly within days or hours, MLOps can help because it removes the technical constraints of the process, turning it into a business decision.

[4]Algorithmia, **2021 enterprise trends in machine learning**

The awkward handoff from data scientists to engineering teams

Data scientists often work in isolated, local environments on their personal notebooks. This is ideal for their R&D focused work but does not reflect some realities of the production context in which their ML models are meant to operate.

Real production situations involve multiple environments, with various sources of data, and interactions with other live systems. Production systems must exhibit high quality and be reliable, scalable, well tested, maintainable, and auditable. This often results in data scientists having to throw their work over the wall for an engineering team to integrate—or sometimes completely rewrite—their models.

This awkward handoff is often exacerbated by having these two disciplines sitting in completely different parts of the organization. Separate teams with different incentives makes the process of getting ML into production difficult and error prone.

Bring the best of DevOps to your ML models

MLOps is an extension of DevOps into the ML space, improving collaboration and integration so that data scientists can do more.

Inspired by Thoughtworks Principal Technologist Ken Mugrage's definition of DevOps, we can define MLOps as "**a culture where people, regardless of title or background, work together to imagine, develop, deploy, operate, monitor, and improve machine learning systems in a continuous way.**"

Let's have a look at all components of this definition:

Culture: MLOps is not just a set of tools or practices, above all, it is a culture. A culture is defined by the way people work together and the values they share. For an organization to adopt MLOps, it must undergo a cultural shift.

People, regardless of title or background, work together: A lot of skills are needed to bring ML models into production, including data science, data engineering, ML engineering, software engineering, build and release, infrastructure, and operations. The important thing is that people in these roles should work together seamlessly, collaborating without creating silos.

To imagine: Designing ML is a mixture of creativity and scientific rigor. To be able to design the right model, the data—the basis for every ML model—must be freely discoverable. Data in silos, proprietary databases, or

datasets locked behind department boundaries are not discoverable and make it impossible to imagine future models.

Develop, deploy, operate: This is the main pipeline of ML development. For reproducible and reliable results, the code, the model, and the data must be kept in sync.

Monitor: As data can change continuously, the model should be monitored in production. Drift and bias in the data presented in production can deteriorate the model's performance with possible catastrophic consequences.

Improve in a continuous way: As real-world data presented to the productive model will change continuously, MLOps is not a one-way process but a continuous circle. Based on changes in the data, the model performance has to be improved by retraining or adapting the model and running through the whole MLOps circle again. The faster and smoother this circle is executed, the faster an organization is able to adapt ML-based processes to changes in the outer world.

> **"DevOps is a culture where people, regardless of title or background, work together to imagine, develop, deploy and operate a system."**
>
> **—Ken Mugrage, Thoughtworks**

MLOps streamlines processes for reliable production

MLOps is designed to combine the creative scientific process of data scientists with the professional software engineering process of releasing software into production safely, quickly, and in a sustainable way.

- To prevent error prone manual steps, almost **everything is automated**
- **Quality is built into the process** and no longer dependent on human testing alone
- **Deployment is done in frequent small batches** to minimize risk and to improve continuously
- Everything is versioned to get the work out of people's heads and into **a repeatable and auditable process**

The value of MLOps

- A faster and more reliable way to deploy and improve models in production
- Higher productivity and impact for data scientists
- Deploying a new model becomes a business decision, rather than a technical one

How continuous delivery for ML (CD4ML) works

Thoughtworks has developed continuous delivery for machine learning (CD4ML)[5], an approach to implement MLOps that adapts the principles, practices, and tools from continuous delivery. In one of its first ML projects, Thoughtworks built a price recommendation engine on AWS with CD4ML for AutoScout24, the largest online car marketplace in Europe[6]. Today, CD4ML is the standard at Thoughtworks for ML projects.

While continuous delivery has been the approach to bring automation, quality, and discipline to the process of releasing software into production, ML systems introduce new challenges beyond just the software release and deployment process:

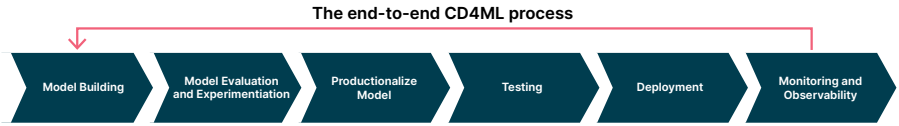
- **Constantly shifting data:** If you train an ML model on stale data, you will get suboptimal results.
- **Models under constant change:** Data scientists are running multiple experiments and researching methods to improve the model's results.
- **Promotion to production:** Once new models are proven useful on training datasets, they need to be promoted to production. This promotion process will require a level of governance to assess potential bias, fairness, privacy, ethical, and other relevant quality considerations before they can be deployed in production.

[5]Danilo Sato, Arif Wider, Christoph Windheuser, Continuous Delivery for Machine Learning
[6]Thoughtworks, Getting Smart: Applying Continuous Delivery to Data Science to Drive Car Sales

- **Automated deployment:** The deployment process itself needs to be robust, automated, and allow for quick rollbacks in case any issues are found.
- **Monitoring in production:** Once in production, you need to monitor how the model is behaving against real world data, to avoid a drift in performance.

CD4ML addresses all these challenges by considering the entire end-to-end process.

CD4ML: the proven process for MLOps at Thoughtworks



Model building: Data scientists start their research phase by exploring the available datasets, understanding the problem, and training initial models.

Model evaluation and experimentation: Multiple candidate models are trained to experiment with different approaches, then evaluated against a test dataset.

Productionalize model: A chosen model is selected and put into production. While this isn't always a step, some cases require a translation to use appropriate technologies for production.

Testing: On the path to production, the chosen model needs to be tested against several aspects. Translated models should still perform similarly against the same test data. The model can also be assessed against potential bias, fairness, or ethical concerns, as well as other non-functional requirements like security or scalability.

Deployment: There are multiple strategies to choose how to release the new model. It can run in parallel to existing models, or gradually replace them to reduce risk and increase

confidence. It can also be rolled back in case any issues are found in production.

Monitoring and observability: Once models are live, performance is monitored against real production data to detect potential drift. This will generate new data that can be used to start the cycle again.

Choose the right solution to implement CD4ML

Implementing and fully automating the end-to-end CD4ML process can be difficult, requiring lots of tools, technologies, and architecture decisions. Many products and solutions are emerging to solve these problems, but you should first understand their strengths and limitations. The ideal technology is one that solves 80 percent of the challenge but allows you to customize and extend it to cover the remaining 20 percent that will be unique to your organization.

Look for the following technical components when deciding on a solution to implement CD4ML:

Discoverable and accessible data: Enable data scientists to find and use the data they need, as well as enhance it as they see fit.

Reproducible model training: Aid automation and reduce the risk of relying on tacit knowledge and local manually crafted environments.

Experiments tracking: Track different experiments and their results for auditing purposes and comparison of multiple variations.

Elastic infrastructure: Leverage the cloud to quickly provision ML training infrastructure on demand, as well as other environments on the path to production.

Version control and artifact repositories: See who changed what, when, and why for auditing and reproducibility purposes.

Testing and quality: Assess different quality aspects of the ML system and automate it as much as possible.

Model serving: Choose the best way to host and serve the models in production, meeting the desired non-functional requirements.

Model deployment: Automate the release and rollback of new models, and how they compete or replace existing models in production.

Monitoring and observability: Track how models behave in production.

Continuous delivery orchestration: Automate the end-to-end process from code and data to production, including manual approvals required for governance purposes.

Bringing ML models into production is only the beginning

To close the feedback loop, it is important to continuously gather and monitor new production data. It can then be curated and labeled into new training datasets that can be used to improve future ML models. This enables models to adapt and creates a process of continuous improvement.

CD4ML is the recommended approach to get MLOps right. **Read the associated technical whitepaper** to learn more about the various ways you can succeed with MLOps.

How will you succeed with MLOps?

About the authors

Christoph Windheuser
Global Head of Artificial Intelligence, Thoughtworks

Before joining Thoughtworks, Christoph gained more than 20 years of experience in the industry in several positions at SAP and Capgemini. Prior to that, he completed his Ph.D. in Neural Networks with a focus on Speech Recognition at the University of Bonn, Germany, Carnegie Mellon University in Pittsburgh, USA, Waseda University in Tokyo, Japan, and France Telekom (E.N.S.T.) in Paris, France.

Danilo Sato
Head of Data and AI Services UK, Thoughtworks

Throughout his 20-year career, Danilo has combined his experience leading accounts and teams with a breadth of technical expertise across architecture and engineering, software, data, infrastructure, and machine learning. Danilo is the author of DevOps in Practice: Reliable and Automated Software Delivery, a member of Thoughtworks Technology Advisory Board and Office of the CTO, and an experienced international conference speaker.