insight.tech POWERING IoT AND EMBEDDED INNOVATION

MLOps: The Path Forward for Scalable AI Workflows

Scalability is one of the most difficult parts of any AI project. First, it takes a huge amount of data to build and train an AI model. And there's an immense complexity in synchronizing different jobs, models, and data set requirements. Challenges like these were the primary drivers of IT and engineering organizations' adoption of DevOps design principles. But DevOps really can't solve these AI scalability issues.

DevOps, a set of software development lifecycle (SDLC) management practices, reduce development time while delivering continuous, high-quality software updates so organizations can scale their engineering output. Given that AI and machine learning rely on repeated model iteration and algorithm optimization, any organization managing multiple AI projects simultaneously should use DevOps to handle things like version control and automated testing.

The catch is that AI projects are fluid. Different AI use cases require different tools, so any infrastructure that supports more than one AI application should be elastic. And the same data can be both the input and the output in an AI system where data and workflows in an ML project can exist on multiple parallel tracks simultaneously.

The DevOps model wasn't designed to deal with these multi-tiered data and workflow hierarchies. Fortunately, veteran developers who created DevOps best practices decades ago are still tackling the problem today.

From DevOps to MLOps

To illustrate how unruly AI can become within an organization, let's start with a story.

Chia-Liang "CL" Kao, an open-source developer for more than two decades, built SVK, a software version control system early in his career. In 2018, a nonprofit AI training academy brought Kao in to develop a back-end management system that would allow different groups of people to collaborate on shared data sets and share development resources without overwriting one another's work or accidentally modifying someone else's data. At the time, there was no equivalent process or architecture for machine learning workflows.

Kao's decades of experience in DevOps development came in handy for this problem, which is larger than it might seem.

"They had 200 people in the first cohort and needed an automated way to manage all the resources for performing deep-learning tasks," Kao says. "Otherwise, you would need 10 people to manually manage the environment to make sure the 200 students' projects didn't interfere with each other.

"How do you get the data, clean the data, aggregate the data, and organize data?" he asks. "And how do you manage your training library and workloads? And once you have the models, how do you keep track of them?"

The process of answering these questions led Kao and other members of the DevOps community to begin creating what would become MLOps.

Prepackaged, Open-Source MLOps

MLOps is an approach to AI lifecycle management that tailors the SDLC practices of DevOps to AI projects and workflows that need to scale. For instance, Kao founded

InfuseAI, a company that builds PrimeHub AI, an open-source pluggable MLOps platform that supports a wide range of tools and software packages from within a single multi-tenant dashboard **(Figure 1)**.

The PrimeHub AI platform uses an API-centric architecture that helps users juggle multiple projects with different software requirements and data repositories without cross-connecting workflow silos. This starts with a prepackaged software stack comprising open-source and off-the-shelf tools that data scientists, AI developers, and IT professionals may already be familiar with:

- Jupyter Notebooks for interactive development
- Crane for building and pushing Docker containers
- PipeRider and ArtiV for data version management and data set metadata tracking
- Framework that enables 3rd-party apps to be seamlessly integrated into the ML workflow, such as model registry and data labeling tools
- Streamlit for data focus and visualization

In addition, the platform supports multiple cloud-based offerings from AWS, Google Cloud, and Microsoft Azure.

As Kao explains, MLOps has to "embrace the diversity of different projects but standardize at a very high level so your tooling can support that practice."

This is where APIs come into play that help integrate every part of an ML stack so different IDEs, libraries, data sources, and more can slide right into DevOps workflows built on top of the software infrastructure described above. They also enable the integration of other common workflow automation tools such as Jenkins servers, TensorFlow libraries, and even model optimizers like the Intel® OpenVINO™ Toolkit.

"It's like a central hub of all the tools that you need," Kao says. "We are not going to re-implement different stages. There are going to be multiple tools. We're not going to replace them all, but we are going to make them work seamlessly."

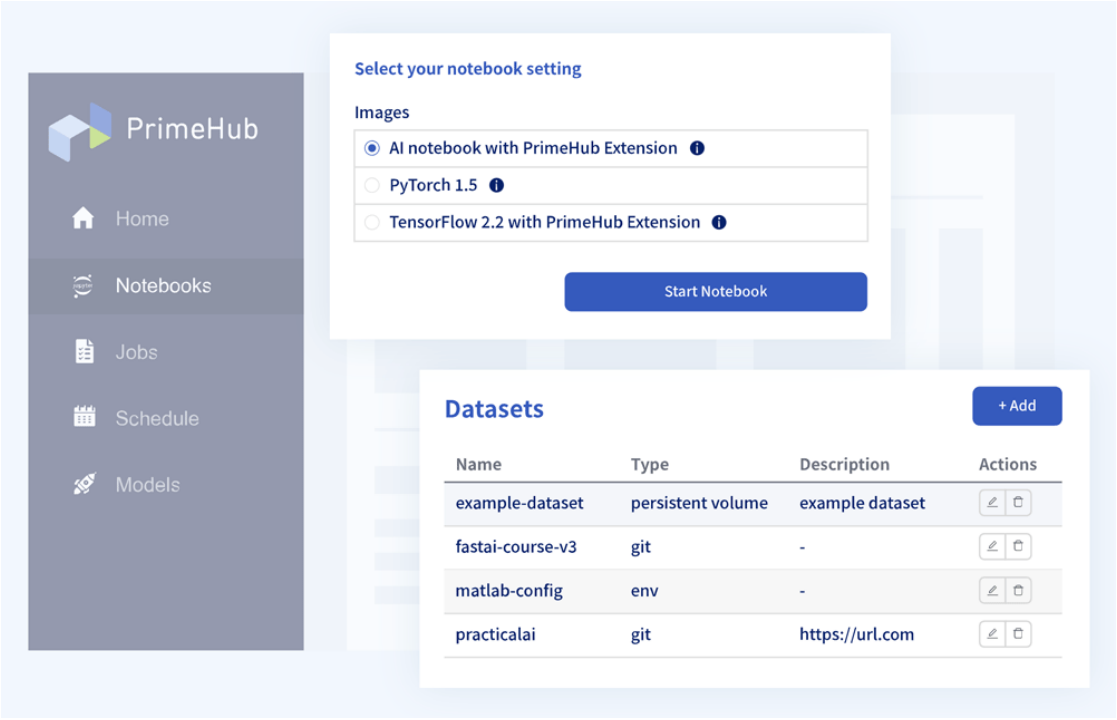

Figure 1. The PrimeHub AI platform allows organizations to manage and deploy multiple ML models, data sets, and tools from a central dashboard. (Source: InfuseAI)

A Matter of Scale

The goal of a successful MLOps deployment is to solve scalability issues so less time is spent configuring workloads and managing data sets and more time focused on AI outcomes.

For example, the PrimeHub AI platform was deployed in a large hospital where a small AI team used it to manage predictive ER capacity at the height of the COVID-19 pandemic. The application involved multiple ML models that tracked different patients' conditions, predicted their likelihood of improvement, and determined how many beds would be available the next day. In parallel, the models needed to be constantly retrained to adjust for seasonal changes, refreshed pandemic data, and other factors, then verified and deployed.

Previously, all these moving parts created a silo between the AI developers and operations engineers that made updating models a weeks-long process. But with PrimeHub MLOps, it became an hour-long workflow that encompassed updating the model, testing it, and deploying it into production.

Extending the classic DevOps model to automate workflows, scale deployments, and make back-end control as simple as possible empowers this kind of organization to capture AI value and ROI. Without the shorter turnaround time, Kao says, "you're just wasting your AI investment that's sitting there idle."