ModelOps

The Key to Operationalizing AI at Enterprise Scale

ModelOp

Table of Contents

Executive Summary

In 2011, noted venture capitalist Marc Andreesen famously observed that "software is eating the world". History has indeed borne him out: low cost computing in the cloud and pervasive connectivity have enabled entire industries to be transformed by software.

Many of the world's most valuable companies generate value wholly or largely based on software, and without exception the most successful companies have adopted structures, processes and technologies that make them adept at creating and delivering software with speed and efficiency.

Today, an even greater transformation is underway driven by the pervasive collection of data and advances in AI. Some very special kinds of software – namely models – are poised to further transform companies and industries. In this new era, one can say that "models are eating the world". Those organizations that survive and thrive will be those that adapt to the unique technical and organizational challenges and opportunities that models present.

This paper explains what models are and details the core requirements for ModelOps, which is the technical and organizational capability essential to successfully deploy, manage and govern models at scale in large, complex enterprises.

> *"ModelOps is a superset of MLOps, and inversely, MLOps is a subset of ModelOps. It is to be noted that every good ModelOps capability will, by default, have a good MLOps capability, but the inverse may not hold true."*
>
> Gartner, "Innovation Insight for ModelOps", Soyeb Barot, 6 August 2020

What is a model?

Models are software artifacts that drive decisions based on data. In some cases models learn from data directly (machine learning) in other cases a human expresses the patterns in the data, such as in rules-based models.

They encode critical intellectual property and as such are highly valuable enterprise assets. As models have evolved and grown in power they have begun to take on an outsized role in how enterprises function and compete.

Models ingest data and produce inferences that are used in making decisions. Increasingly, the inferences produced by models are consumed by software applications and used to partially or fully automate business decisions.

Like software in 2011, models are not new. Many enterprises have employed models for decades, largely rule-based models or algorithmic models that perform mathematical optimization. In financial services for example, these types of models have traditionally been used to assist with myriad tasks such as processing loan applications, handling insurance claims, pricing products, or executing trades. These types of models are still in wide use and will likely remain so for many years to come.

In several key respects, ML models are not like conventional software or even like other types of models:

- ML models do not execute deterministic rules. They are statistical artifacts that are created by"training" with data. With appropriate design and training, ML models can make very effectiveinferences from complex data that would be impractical or impossible to code using explicit rules.

- Because ML models are trained using data, their effectiveness in making accurate inferences onlyholds for as long as the data they're being fed in production matches the conditions present when thetraining data was collected. As the world changes, and the "operating regime" reflected in the datapresented to the model in production diverges from what was present in the training data, theaccuracy of an ML model's predictions will degrade. Even under normal conditions, each model has a natural cadence at which it needs to be refreshed (retrained) in order to maintain its desired effectiveness, which can range from months or weeks to as short as a day or less.

- In extreme situations in which there's a significant, rapid change in environment (the Coronaviruspandemic being a prime example), ML models can lose their predictive efficacy very rapidly.

- Models have very exacting technical requirements. If the production environment departs even insmall ways from the environment used in development, the model may not operate properly.

What is a model?

- Model life cycles require participation from a large number of stakeholders, including the line ofbusiness (which is typically the sponsor for the model), the data science team that develops the model,the DevOps team that integrates the model into production applications, the DataOps team thatsupplies the data pipelines, the ITOps team that operates the production infrastructure, and thegovernance organization that ensures compliance with internal and external regulations. In several key respects, ML models are not like conventional software or even like other types of models:

- ML models do not execute deterministic rules. They are statistical artifacts that are created by"training" with data. With appropriate design and training, ML models can make very effectiveinferences from complex data that would be impractical or impossible to code using explicit rules.

- Because ML models are trained using data, their effectiveness in making accurate inferences onlyholds for as long as the data they're being fed in production matches the conditions present when thetraining data was collected. As the world changes, and the "operating regime" reflected in the datapresented to the model in production diverges from what was present in the training data, theaccuracy of an ML model's predictions will degrade. Even under normal conditions, each model has a natural cadence at which it needs to be refreshed (retrained) in order to maintain its desired effectiveness, which can range from months or weeks to as short as a day or less.

- In extreme situations in which there's a significant, rapid change in environment (the Coronaviruspandemic being a prime example), ML models can lose their predictive efficacy very rapidly.

- Models have very exacting technical requirements. If the production environment departs even insmall ways from the environment used in development, the model may not operate properly.

- Model life cycles require participation from a large number of stakeholders, including the line ofbusiness (which is typically the sponsor for the model), the data science team that develops the model,the DevOps team that integrates the model into production applications, the DataOps team thatsupplies the data pipelines, the ITOps team that operates the production infrastructure, and thegovernance organization that ensures compliance with internal and external regulations.

What is a model?

- A large complex enterprise with multiple business units can have hundreds or thousands of models,each with a unique life cycle in terms of its business KPIs, development platform, training andretraining criteria, production environment, reporting and alerting thresholds, approvals regime and compliance requirements.

- Because models automate decisions that have a direct impact on the top and bottom lines theyrequire direct accountability to the business organizations that employ them. Line of business managers need real-time visibility and control over models in production, to a much greater degreethan is typical with conventional software applications.

- Because the decisions that models automate can have a direct and significant impact on people's lives-e.g. whether or not they get a loan, or a job interview, or a raise - models require significant governance oversight to limit organizational exposure and risks.

- Perhaps most importantly, models encode an organizations' most valuable intellectual property, i.e. the behaviors of customers, employees, suppliers, partners, and markets. While much of the enterprise software that was previously proprietary has been commoditized, many of the mostvaluable models do not readily commoditize. This makes models, even more than conventional software, critical corporate assets that deserve and require extremely effective curation and management.

What is ModelOps?

In view of the unique nature of models and their life cycles, many large enterprises that were not "born digital" have experienced significant difficulty moving models from development into production, ensuring ongoing monitoring and governance. The response to this challenge is the development of a new discipline known as ModelOps.

ModelOps is an organizational capability that enables large enterprises to scale and govern their AI initiatives. While ModelOps is required in some form by any organization, Enterprise ModelOps is proving to be essential for large, complex enterprises that intend to broadly leverage models across their operations.

Modelops: A Definition

"ModelOps (AI model operationalization) is primarily focused on the governance and life cycle management of AI and decision models (including machine learning, knowledge graphs, rules, optimization, linguistic and agent-based models). Core capabilities include the management of model development environments, model repository, champion-challenger testing, model rollout/rollback, and CI/CD integration. ModelOps enables the retuning, retraining or rebuilding of AI models, providing an uninterrupted flow between the development, operationalization and maintenance of models within AI-based systems. ModelOps provides business domain experts autonomy to assess the quality (interpret the outcomes and validate KPIs) of AI models in production and facilitates the ability to promote or demote AI models for inferencing without a full dependency on data scientists or ML engineers."

Gartner, "A Guidance Framework for Operationalizing Machine Learning", Soyeb Barot, 14 May 2020

Subscribers can read the entire note at Gartner.com

What is ModelOps?

In large enterprises, an effective ModelOps capability accelerates AI initiatives across the company. ModelOps eliminates waste, friction and excess cost, and unleashes the creativity of the business - including professional and citizen data scientists – while protecting the enterprise from potentially unbounded risks.

Critical ModelOps capabilities include:

- A CAD system for defining the end-to-end life cycle for each model.
- A business process execution engine for automating and visualizing the full model life cycle.
- A model catalog for all models that captures all of their metadata and artifacts (technical requirements, training data, business KPIs, approval requirements, etc.).
- Mechanisms for encapsulating models for deployment in different types of infrastructure (on prem, private cloud, public cloud(s), hybrid).
- Mechanisms for creating model pipelines that interconnect a series of models and data sources.
- Mechanisms for instrumenting and monitoring each model's technical and business performance in production.
- Facilities for A/B and champion/challenger testing on models in production.
- Tools for defining and performing compliance tests and generating reports used for for internal and external audits (explainability, bias, etc.).

What is ModelOps?

- Interfaces to business intelligence (BI) and other analytic and operational systems to connect model performance to business KPIs and enable line of business managers to manage their investments in AI and their impacts on their businesses.
  - Alerting and reporting to identify and address:
  - Technical performance issues
  - Business performance issues
  - Operational bottlenecks that impede model deployment or continued operation
- Connections to the enterprise operational stack (code development, ticketing, performance monitoring, security, authorization, etc.) to avoid duplication of centralized IT infrastructure.It's important to note that for large, non-digital native organizations that may be using algorithmic, rule-based and other types of models in addition to AI models, the ModelOps capability needs to encompass all types of models and should be able to create and manage model pipelines that include a multiplicity of different types of models.

How does ModelOps Relate to MLOps?

MLOps arose as a response by data science workbench tool vendors (and some open source projects) to the challenges organizations face when trying to deploy machine learning (ML) models.

MLOps addresses the subset of ModelOps that deals with technical requirements for deploying ML models such as cataloguing, encapsulating, and monitoring them. Some MLOps systems support functions such as retraining, however model creation tools that support retraining are more geared towards development environments than full-scale production environments operated by central IT organizations. MLOps does not address non-ML models (algorithmic, rule-based, optimization, etc.) and does not define or automate the full model life cycle from business KPIs through governance and compliance. MLOps also does not support rule-based, algorithmic or other types of non-ML models.

One good way to distinguish a ModelOps capability from an MLOps capability is in terms of the constituencies that it supports and the questions that it can address. An effective ModelOps solution fully automates responses in these and many other scenarios:

- Show an internal auditor or external regulator how a particular model running in production was trained and tested, including the data sets used.
- Show a business analyst if a new model is causing an observed dip in a key business KPI.
- Enable an enterprise AI architect and business analyst to define and automate set the rules and thresholds that will trigger a model retraining or refresh.

How does ModelOps Relate to MLOps?

MLOps typically cannot on its own address these questions without days or weeks of manual effort, if at all.

To provide enterprise-wide management of all model life cycles, a ModelOps solution needs to support models of all types (ML, rule-based, algorithmic, optimization etc.) created by any type of model creation tool. For ML models, a ModelOps solution can leverage the MLOps capabilities that the creation tools provide (if available), and must also provide a full suite of MLOps capabilities to support those model creation tools, such as some open source packages, that don't include MLOps capabilities. Per the diagram below, an effective ModelOps solution (which must be cloud agnostic) also complements and extends MLOps capabilities in Cloud/Hyper-scale platforms, and interfaces with DataOps, Compliance and Risk Management tools, BI tools and business applications..

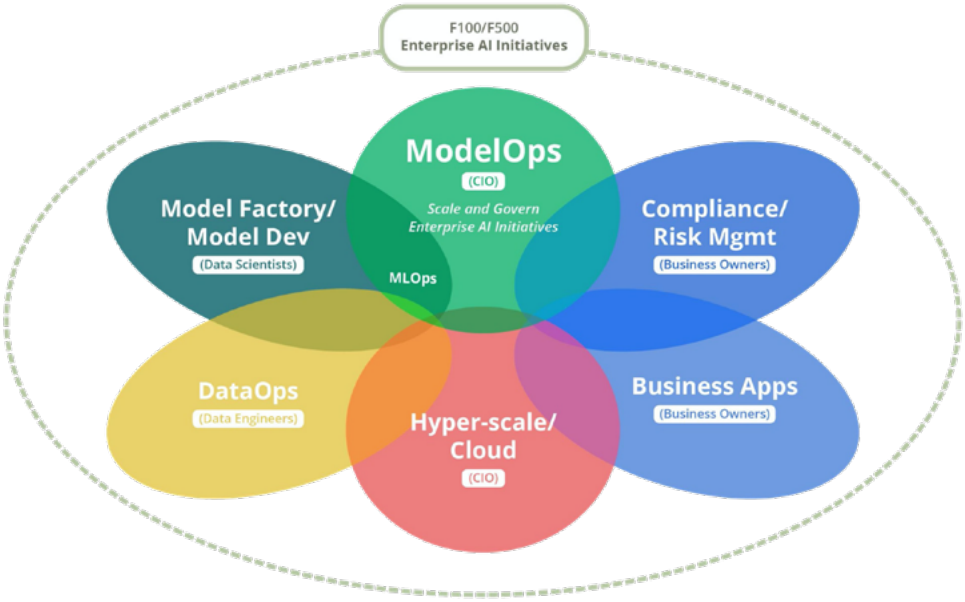

Figure 1. ModelOps: a key capability for Enterprise AI initiatives, that includes MLOps features

Who ”Owns” ModelOps?

The appropriate organizational owner for ModelOps can vary for different types of organizations. For large enterprises that have a central IT organization responsible for providing IT operations across the company, ModelOps is necessarily the purview of the CIO, ofter provided as a Shared Service. There are several reasons for this:

- The CIO’s organization operates the company’s production IT infrastructure and the ITOps team is the only group authorized to manipulate production systems. There’s no provision for a data scientist to release, say, a retrained model into a production IT system.
- The ITOps team is the only group within the company with the organizational resources and skills required to operate 24x7 business-accountable systems and applications.
- The CIO organization includes enterprise AI architects that can design and implement end-to-end model life cycles.
- The CIO organization operates the systems that provide visibility and control for the line of business and compliance organizations.
- The CIO organization is responsible for implementing the organization’s digital transformation and cloud journeys.

Per the diagram below, the Enterprise AI Architect and the ModelOps team are the primary operators of the ModelOps solution, but all other stakeholders in the Model Life Cycle interact with the system, even if “invisibly” via interfaces to their primary tools (i.e. the data science workbenches, BI tools, DevOps tools, risk management systems, etc.).

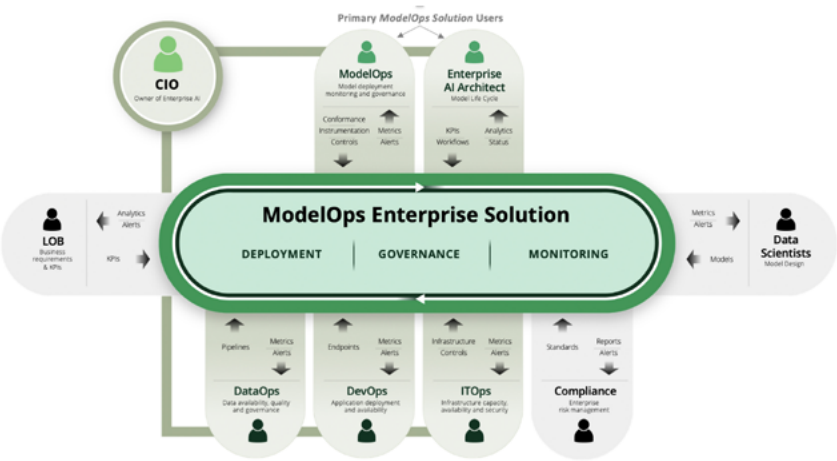

Figure 2. A ModelOps solution resolves the organizational challenges in the Enterprise AI journey

What's the Best Way to Implement ModelOps?

Implementing a highly performance, enterprise-grade ModelOps capability is a transformational process that requires significant planning and participation among disparate organizations across the company. The most critical factor for success is to develop and communicate a common vision across the company of models as critical, first-class enterprise assets that are fundamental to the organization's ability to compete and drive profitable growth.

Although implementing an effective ModelOps program is a project with significant depth and scope, it is neither necessary nor desirable to address it as a monolithic project. The best way is to start with a few models, or even a single model, and implement, test and revise in stages. Some steps that are especially important at the outset include:

- Designate the appropriate member(s) of the CIO's organization to lead the ModelOps effort.
- Designate a lead Enterprise AI Architect [Note – qualifications and job responsibilities for this and other roles can be found in the Additional Resources below].
- Identify candidate model(s) for initial implementation and identify all of the key stakeholders from all parts of the organization.
- Identify the key business KPIs, operational requirements, the regulatory and compliance requirements and the reporting requirements.
- With guidance and support from the Enterprise AI Architect, design end-to-end life cycle(s) for the models.
- Select and Enterprise ModelOps solution to provide the framework for the ModelOps program.
- Integrate the ModelOps solution with the data science tools, the DataOps, DevOps and ITOps systems, the business applications and analytics tools, risk management systems and the enterprise IT shared services stack.
- Create the reporting dashboards and implement the alerting systems and triggers.
- Run the model through the life cycle, monitor and measure, and track key parameters (such as ModelDebt) against targets.
- Identify learnings, adjust the processes as necessary, and bring on more models.

Additional Resources

- **ModelOps Essentials Guide –** Includes descriptions and qualifications for key roles, including the Enterprise AI Architect and ModelOps Engineer
- **ModelOps Masterclass Series** – An introduction to ModelOps solutions and ModelOp Center
- **Forbes –** An introduction to Model Debt
- **Gartner –** "A Guidance Framework for Operationalizing Machine Learning" (For Gartner subscribers only)
- **Forrester –** "No Deploy, No Joy: Leverage ModelOps To Operationalize AI And Machine Learning"

Learn more about us and our leading ModelOps solution: ModelOp Center.

About ModelOp

ModelOp, the pioneer of ModelOps software, enables large enterprises to address the critical governance and scale challenges necessary to fully unlock the transformational value of enterprise AI and machine learning investments. Core to any AI orchestration platform, G2000 companies use ModelOp Center to govern, monitor and orchestrate models across the enterprise and deliver reliable, compliant and scalable AI initiatives.

Contact us to make Enterprise AI real with ModelOps

ModelOp