A SMART STARTER GUIDE TO MACHINE LEARNING OPERATIONS

atrium

Table of Contents

If you've been in the software development industry or simply used software, you've no doubt heard of DevOps. But what about **machine learning operations**, or more commonly known as **MLOps**? In order to do a deep dive into MLOps — the potential, the benefits, and best practices — we have to look at its beginnings.

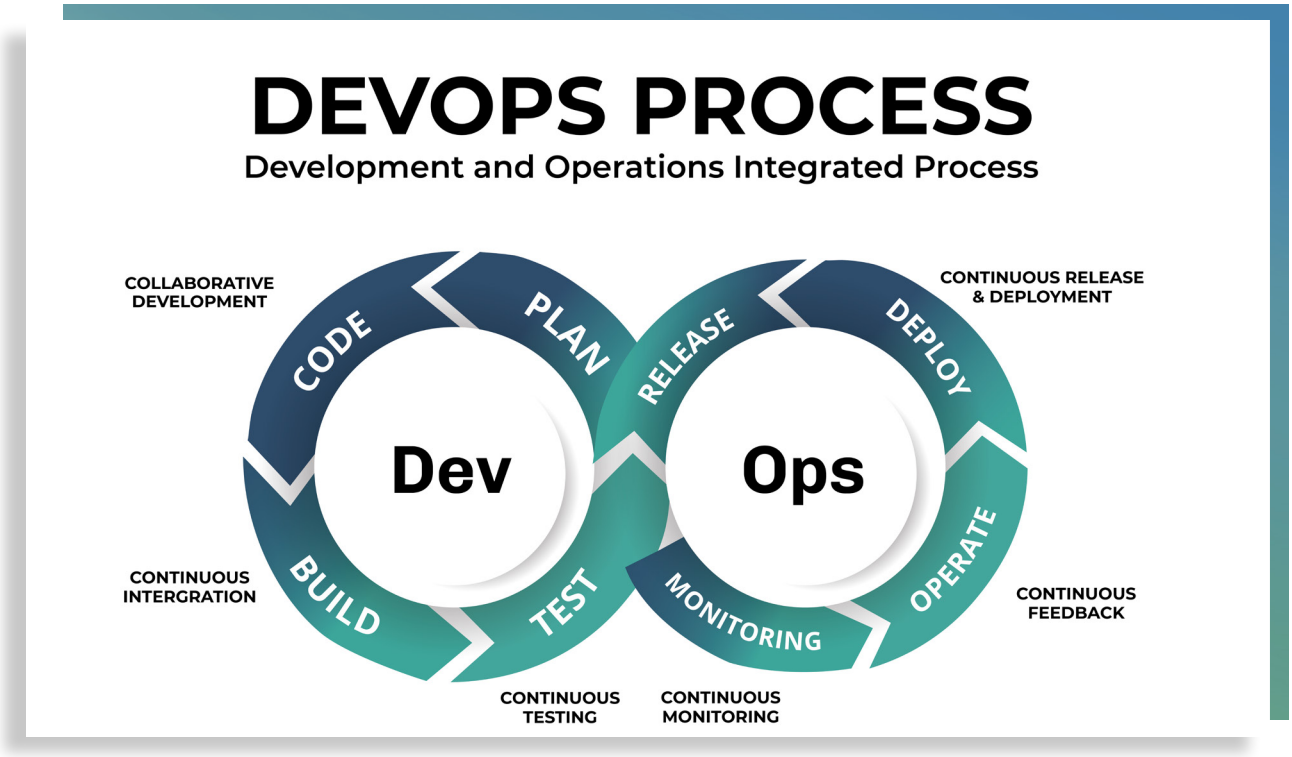
The origin story:
DevOps and MLOps

First, you can think of DevOps as the pursuit of faster and more reliable software delivery through communication, practices, measurement, and automation.

DevOps focuses on managing changes of code and configuration, but shipping machine learning models require more rigor.

MLOps adds the tracking of changes to models and data to the code lifecycle.

Models might have many different experiments in their construction, tuning of parameters, and each of these needs to be tracked/versioned and cataloged. Additionally, the data sets that are fed into the models may be part of what defines them, so tracking and versioning changes to data, its schema, and samples are required too.



So now you can hopefully envision a “pipeline” in your organization.

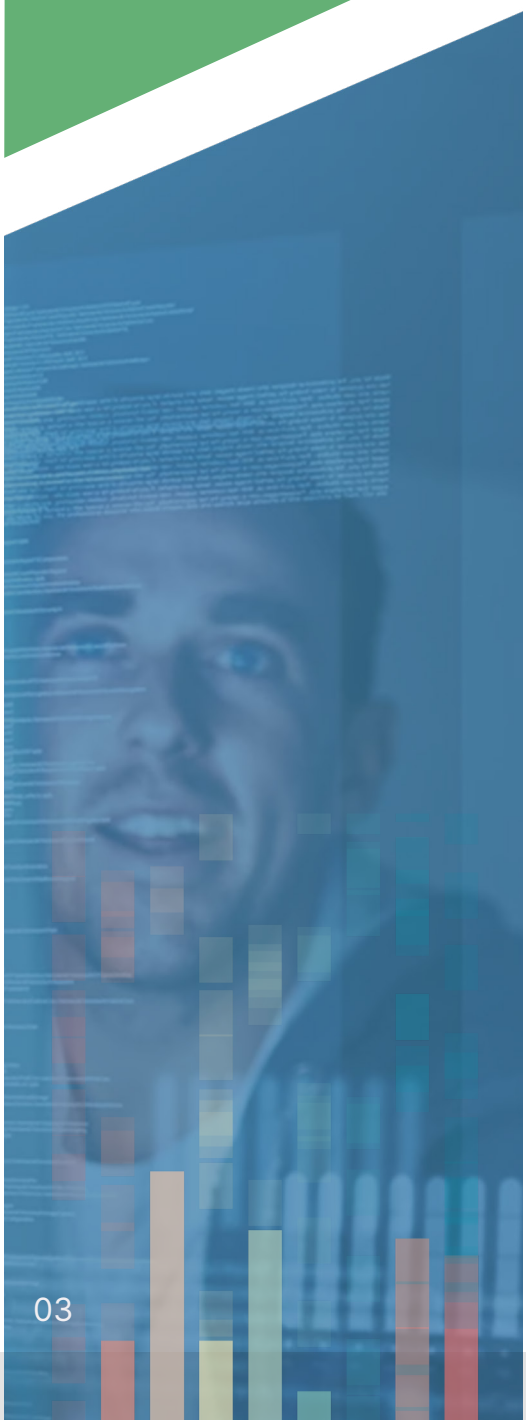
It starts with data sources coming in, followed by a series of steps to clean, refine, and prepare that data for analytics and data science teams to consume. In the case of the data science teams, a series of experiments and iterations are conducted to produce one or more models.

Finally, extra code may need to wrap these models to expose them via APIs or make them available to provide value.

MLOps CI/CD Pipeline

MLOps is the pursuit of faster and more reliable delivery through this pipeline of data to insight and value.

While DevOps was concerned with breaking down the silos of development and operations, MLOps builds upon DevOps, but adds the breakdown of the data engineer/ETL silo, the analytics silo, and the data science silo under its domain.



The essential roles of an effective MLOps team

Building machine learning solutions requires additional support staff to assist the researchers and data scientists building the models. This MLOps staff has many of the same responsibilities as a DevOps engineer would have in a traditional application development role. The significant difference is that the data itself must also be managed.

Data management includes extraction, validation, modification, versioning, and coalescing. These tasks are known collectively as data pipelines. Data pipelines are complicated enough, but the MLOps engineer also needs to track the data pipeline's relationship to the code (i.e., model) pipeline. In the classic DevOps world, the data's shape and extent are relevant, but the data values are not.

However, in the machine learning world, the code is a function of the data itself. The bad news is that this is much more complicated than just DevOps plus snapshotting data. The two pipelines are highly dependent but loosely coupled.

With the rise of full-stack development, the good news is that the varied skills needed for these tasks can be available from a minimal set of individuals. We have defined three high-level roles: digital plumber, data wrangler, and cloud architect. These roles should exhibit all the skills necessary for good MLOps support staff. Of course, these three roles don't necessarily translate to three people. Depending on your mileage, you may find one person that can handle all the tasks in question, or you may need multiple.

Digital Plumber

The digital plumber is responsible for getting, moving, and combining data from disparate sources, systems, and networks.

This person should be comfortable with APIs, the consumption, and the development of REST services. They should also be comfortable with networking and network security. The digital plumber's pedigree is a back-end developer, system administrator, or any other developer with a strong background in shell scripting, scheduling, and monitoring jobs. This resource is also comfortable in SQL and ETL (aka extract, transform, and load). If your team does not have a strong shell scripter, someone with a mastery of an ETL platform is critical for data, but often comes up short running CLI commands.

Once the data scientist develops the model, it is the digital plumber's responsibility to wrap the output into a service called by the client application. Plumbing the model output to the end-users is as critical as plumbing the data to the model, so the digital plumber needs to be a strong API developer.

All of the digital plumber's work products are created in containers, using a platform such as Docker.

This is a critical area of expertise for all MLOps support staff. Ideally, one resource has a mastery of Docker, and most likely it would be the digital plumber. The use of containers is a cornerstone of MLOps environments that must be ubiquitous and identical regardless if they are hosted on a workstation or the cloud.

Data Wrangler

The next role that we will dig into is that of the data wrangler, who is responsible for the pre-processing of all the data.

Often the responsibilities of the data wrangler can be performed by the digital plumber or even the data scientist developing the model, but it's important to define this role on its own merits.

Transforming raw data into model inputs requires data-munging techniques. Aggregation, deduplication, normalization, sorting, mapping, and anonymizing are a few of the data wrangler's exceptional skills. Aside from the physical transformation, they also analyze variances over time to identify events that cause change in the data's behavior or how it was entered.

For example, consider modeling voter demographics for congressional districts. The data wrangler would look for drastic changes in the demographic over time which may have been caused by redistricting. They would then determine the best segmentation strategy. This role will be responsible for developing the "pre-processing" data pipeline — not to be confused with the digital plumber pipeline. Yes, they may be connected in one super-pipeline, but they should be developed separately.

Both data pipelines should follow the basic principles of continuous delivery. The code that created these flows should be tracked in version control, and the data they produce also needs to be versioned. Ideally, the entire process is void of human interaction, totally automated. Even the environments running these jobs should be completely ephemeral and spun up as containers as needed.

This "phoenix environment" should be burned down when it has served its purpose, only to rise from the ashes when summoned.

True to the spirit of continuous delivery, the data pipelines should have built-in validation. This could be as simple as asserting record counts if the results are deterministic or perhaps more complex. The critical point of these validation checks is to halt the run and alert MLOps staff when they fail so the data wrangler or digital plumber can address them as quickly as possible.

Cloud Architect

The cloud architect's primary function is to identify the platforms and tools for the entire machine learning flow.

Much like the digital plumber, this role comes directly from the enterprise architecture team, and has mostly the same skill set with additional data pipeline tools and data versioning tools. The cloud architect will first look internally to determine if the existing platform's capabilities match the "productionized" pipelines and model service's needs. Often, this entails examining the current platform's features vs. additional licensing costs for new features. If the existing infrastructure is insufficient, the cloud architect will identify new platforms to fill the gap. The cloud architect will identify the open-source frameworks, languages, and databases directly related to platforms and infrastructure to cooperate with the data wrangler and digital plumber.

Of course, there are always multiple technologies that will meet the project goals, and often these selections fall back to personal preference or familiarity with one language over another. So the cloud architect should also be able to play mayor to Data Town and make sure that a compromise is made between stakeholders. This role should also have a strong background in security.

Depending on the data's nature, they may choose to enforce a zero trust architecture by assuming that no two systems may relax their security even if they share a common network perimeter.

The cloud architect often plays the technical lead role and becomes the ambassador to the MLOps mission, vision, and goals. This team member is typically more senior — someone who knows the business, infrastructure, and technology in equal parts.

Solving for pressing business needs with MLOps

Simply put, any company that has productionized machine learning models to drive immediate action needs to implement MLOps. For example, implementing next best actions requires a real-time response. If any part of the system becomes unavailable, it may lead to revenue loss or unhappy customers. However, it is essential to note that it is not only the system being unavailable that may lead to adverse conditions, but also the model's gradual inefficiency. This decay is why we must have staff dedicated to this process. You can't set and forget machine learning models — or MLOps as a practice.

In contrast, a large team of dedicated data scientists may work for months to generate a model to lead to a single discrete action that does not require an MLOps staff. A one-time report generated for the board of directors to solidify a strategic direction does not need a data pipeline. This happens on a smaller scale too. Dedicated data science resources often look for signals in large data sets' noise in hopes of finding correlation. These models may become productionized at some point, but at this stage, they are still science experiments.

Other essential skills by area of expertise

System admin

The system admin maintains and manages hosts (typically Linux), including software and os patches, and databases. Although on-premises servers are not as common as they were a decade ago, the jack-of-all-trades knowledge of the system admin is well-suited for cloud platforms, containers, and hosting tools. Other aspects of the system admin's function include:

Shell scripting

Most modern platforms have a command-line interface (CLI) to access their API and its corresponding data. A good shell scripter can use primitive scripts on the host to automate tasks via a platform's CLI or API.

Database administrator

Data is easier and quicker to access and manipulate if stored in a database, and the DBA plays a vital role in defining the data's schema and access methods.

Docker/Kubernetes

Data is easier and quicker to access and manipulate if stored in a database, and the DBA plays a vital role in defining the data's schema and access methods.

Job Monitoring

Pipelines can be complex or straightforward; however, they often require an external system to determine if they are completing their tasks and to notify the authorities if they are not performing as expected.

Developer

The heart of the MLOps system is made up of many tiny programs strung together. A strong development background is a key area of expertise.

Python

Python is a programming language that is popular with data scientists for model development. It's also a powerful utility languaage that has many third-party modules.

Code Manager

A team member who approves and merges in "pull requests" to the Git repository is vital when multiple developers are working on the same code base. This person also ensures that coding standards are followed.

Jupyter

Jupyter Notebooks/Labs is a browser/web system to host multiple data science languages like Python, R, and NodeJS. It allows developers to work with small blocks of code in cells and also integrated documents and discussions.

CI/CD

Continuous Integration and Continuous Delivery are DevOps principles where developers' code immediately becomes live in a test environment as soon as a feature is complete. This is a valuable approach to identify bugs before they become problems.

REST Services/API Development and Consumption

Data is easier and quicker to access and manipulate if stored in a database, and the DBA plays a vital role in defining the data's schema and access methods.

Testing

Developers concentrate on creating code that does what it is supposed to, but a tester is dedicated to breaking the code. Splitting testing from development is an excellent approach to provide better features faster.

ETL

Extraction Transformation and Loading (aka ETL) is the primary focus of the data wrangler, which involves:

SQL

All MLOps team members should have strong SQL skills, since most data is stored in a SQL or SQL-like database.

NoSQL

Although less-common in the data science world than the application development world, there is also a need to extract data from a schema-less datastore.

Pandas

Pandas is a Python data framework that allows a table-like concept to be stored in memory. It has all the modern SQL features but also provides data science and statistical functions.

Data Pipelines (Airflow)

Airflow is an open-source manager to tie together small data ETL tasks into larger workflow flows called DAGs (i.e., directed acyclic graph).

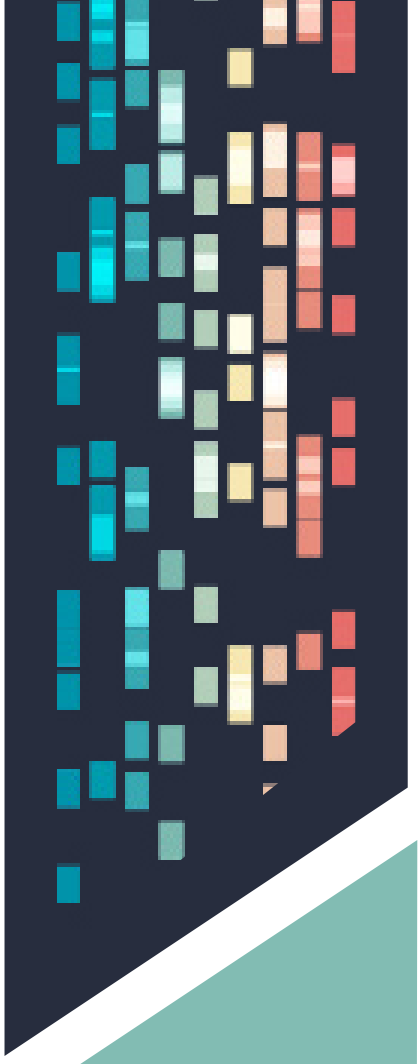
Today, MLOps has a broad set of use cases

The machine learning branch of AI has now transitioned beyond the peak hype phase, and organizations are rightfully expecting a return on their investments in the space.

Similar to what DevOps did for software application development, MLOps encapsulates the range of work, tools, and infrastructure associated with deploying, versioning, integrating, monitoring, and governing machine learning artifacts in a reliable, scalable, and repeatable way. It is the *sine qua non* of machine learning; it's where the rubber meets the road.

Undeployed machine learning tends to quickly fade into science project obscurity, while by contrast, we see the consumption of model outcomes driving measurable and persistent value. Consequently, MLOps needs to extend beyond the technical deployment of models for operations to the integration in day-to-day user experiences, business processes, and workflows.

We see customers implementing process automation around alerting of outliers and threshold violations (i.e., approvals, rejections), automatic rerouting of items in the workflow (i.e., queue assignment), actioning data rows at scale (i.e., bulk assignment/mass handling), adding business logic to predictions (i.e., next best action) and many other exposure mechanisms for machine learning outputs. Successful, modern MLOps ultimately occupies this entire space as a frictionless pipe running from model management to outcome consumption.



Some industries leverage MLOps better than others

Industries and companies more advanced in machine learning are, almost by definition, more effective in MLOps.

Organizations in telecommunications, high tech, and financial services particularly have invested in the foundations needed to build, deploy, integrate, operate, and govern their machine learning solutions end to end.

MLOps is both a set of processes and the infrastructure and technologies that organizations who are leading with AI have come to recognize as a critical piece of the puzzle.

We've seen firsthand that stand-out Salesforce customers have established or are investing in:

- Automating most aspects of model training.
- Model governance frameworks, including release-to-production processes.
- Comprehensive model-performance tracking.
- Automated model refresh (i.e., retraining) and rescoring based on drift or business changes.
- Procedures to mitigate the introduction of harmful bias in the model training process and subsequent monitoring of predictions for disparate impact.

Salesforce technologies delivering on the promise of MLOps

Any solution that applies machine-learning models to a business process or decision-making requires some level of MLOps.

As a result, the scope across Salesforce applications and technology is extremely broad, ranging from embedded, out-of-the-box machine learning-based solutions like Product Recommendations in Commerce Cloud and Marketing Cloud's Send Time Optimization, through configurable Service bots, to fully custom machine learning solutions implemented in Einstein Discovery.

These solutions offer very different levels of control and configuration, and that directly influences the scope of MLOps applicable. For instance, Lead Scoring in Sales Cloud Einstein is intended for easy setup by Salesforce administrators, is optimized for automation, and is intentionally constrained in scope. Training data is sourced from the Lead object, models are automatically retrained monthly and predictions are shown on open lead records only. By contrast, a pricing guidance model built with Einstein Discovery for a quoting workflow in Salesforce CPQ is likely to have multiple and heterogeneous sources.

It must be seamlessly integrated into the operational workflow and carefully monitored for accuracy, effect on the business KPI, as well as user acceptance of recommended prices and their feedback.

Real-world examples of effective MLOps via Salesforce technologies

Model training and deployment

Einstein Discovery and/or Einstein Platform to train, deploy and manage models.

Analytics

Tableau CRM to capture, analyze, and monitor model performance and track the ultimate business KPIs.

Automation

a range of process automation tools in the Salesforce platform including Process Builder, Flow, Validation Rules, Next Best Action, Approval Processes etc. to blend model outputs with business rules and to embed in the user's workflow.

Asset lifecycle management

tools including changesets and packaging allow controlled migration of models and related metadata from pre-production environments to production as well as integration with source control.

Insight: What does it take to be successful with MLOps on Salesforce?

First and foremost, it's critical to think ahead to what production deployment looks like at the outset of your machine learning initiatives and avoid falling into the trap of training nominally accurate models in the lab which are undeployable into the real world.

Part of this is a fundamental recognition that machine learning is intrinsically different from traditional (deterministic) software solutions and requires care and feeding post-deployment. More specifically:

Think about closing the loop.

Invest in tracking predictions and their relevance to the consumer or consuming process. Where you're delivering for Salesforce users, native integration provides a unique opportunity to do this and eliminates many barriers to measuring and acting on drift.

Select tightly integrated technologies.

This will mitigate the substantial risk of getting bogged down in glue code and costly data integration work.

Plan for resources and budget.

To operate your machine learning solutions post-deployment.

Learn from others.

Leverage partners with a track record of success in delivering production machine learning solutions on Salesforce.

Build for trust and governance.

Document your model design and solution fit in shareable model cards. This supports a model governance strategy and helps avoid misuse and the potential for reputational damage or other undesirable consequences of biased or misconceived models operating at scale.

Predictions: What's next for MLOps?

A "bang for the buck" approach

Companies will increasingly focus on "bang for the buck" in selecting and implementing AI and will prioritize the application of proven machine learning techniques to measurable business problems. Those with deep pockets and substantial resources will bifurcate their AI strategy with smaller but important investments dedicated to nascent AI techniques and applications.

A shift from nice-to-have to a must-have

Ethical issues related to biased machine learning applications will increase. Legislation governing AI in the European Union will up the ante for organizations applying machine learning in production; explainable AI will shift from a nice-to-have to a must-have and companies will expand the remit of CISO/CDAO/CAIO roles to oversee governance.

More successful operationalization of use cases

Tighter integration and overlap of (descriptive) analytics and machine learning products will provide customers with new technology options, the opportunity to deliver value from existing technical architecture, and will enable smaller organizations who lack the resources and data science expertise to successfully operationalize their use cases.

Growth, growth, and more growth

We'll see relentless growth in machine learning adoption on Main Street with an emphasis on building out and operating numerous use cases. Additionally, leaders will progressively industrialize their machine learning initiatives by leveraging MLOps techniques, practices, and integrated technology. Scale, efficiency, and cost benefits will ensue.

Common MLOps trends to pay attention to now

Fundamentally, building and improving machine learning models requires many iterations and small improvements over time that lead to big improvements in overall performance. As such, we see many enterprises adopting the fundamentals of agile methodologies to their data science practice, applying principles of CI/CD to model development, testing, and deployment.

However, beware of trying to leverage traditional software practices directly to your data science programs! More on that below...

Another trend we're seeing in the marketplace surrounds testing and monitoring of machine learning/AI models. It can be difficult on the surface to determine if a model is performing well, with erroneous results potentially even *improving* production. As such, we are seeing a drive toward building out techniques for creating non-machine learning baselines, using visual debugging tools, and leveraging model performance predictors. Facebook, Google, and other organizations experienced with large-scale production machine learning repeatedly emphasize the importance of machine learning-specific production metrics that range from health checks to machine learning-specific resource utilization metrics.tasks

A key factor in the acceleration of many organizations' data science programs is leveraging the open source ecosystem for model development using tools such as TensorFlow, Spark, Pytorch, R, etc. This ecosystem is growing rapidly, with tooling used by enterprises of all sizes and scales such as Apache Atlas for governance and compliance, Kubeflow for MLOps on Kubernetes, MLFlow for lifecycle management and Tensorflow tracing for monitoring.

Finally, we are seeing a rapid adoption of cloud-based services to make production machine learning easier. Initially setting up machine learning/AI models and getting them deployed into production workflows can be daunting, even with open source tools available for each stage of the process.

The cloud offers an attractive alternative, since the resource management aspects of the process are handled by the cloud backend. Furthermore, it can be highly cost-effective to leverage accelerators (e.g., GPUs, TPUs, etc.) from cloud-based services, as the initial infrastructure investment to get these up and running can be substantial. Avoiding an initial large in-house infrastructure rollout can greatly improve speed to market for machine learning/AI models, as well as control initial costs as processes mature.

#1: A Lack of an accountable and empowered owner for the enterprise MLOps process

Problem:

With many of our customers today, we do not see a dedicated owner with operational responsibility for MLOps (or the necessary budget to implement it). Even within organizations that have started down this path, there is often confusion regarding where ultimate accountability will land. We see MLOps sometimes landing with the Chief Data Officer's (CDO) organization, sometimes the Chief Information Officer's (CIO), and in other cases it is treated as a more tactical and technical initiative being operated out of a Chief Technology Officer's (CTO) office or even an executive in charge of analytics or business intelligence (BI). With this confusion around ownership, mandate, and budget, it is a challenge for companies to create scalable, repeatable processes which can hold contributors such as data scientists, DataOps, DevOps, and security specialists accountable to various aspects of critical operational considerations.

Recommendation:

Early on in the process, choose an owner for MLOps within your organization, and set a budget for them to establish Key Performance Indicators (KPIs) that can be measured, both as a baseline and ongoing, to gauge effectiveness of the MLOps processes being implemented. Typical measures include factors such model readiness and successful deployment, as well as other factors such as the amount of time since the model was last trained. Ultimately, your goal is to measure business metric improvement and overall ROI from the data science investment.

#2: Lack of understanding of the immediate importance of full machine learning/AI capabilities to the overall business

Problem:

All too often, enterprises view and invest in data science or machine learning/AI as a speculative experiment or strategic talking point. These investments can be financially significant, and we see large data sets being collected and potentially valuable models generated, but this occurs without the requisite feedback of realized value coming from the main line of the business. As a result, machine learning/AI programs experience costly churn, leading to missed-opportunity cost and reduced competitiveness in the main line of the business.

Recommendation:

Your data science work should not be viewed as a cosmetic or speculative investment, and it is a mistake to under-invest in a full model lifecycle capability that will enable your machine learning/AI models to have a business impact as quickly as possible. Management must be clear that machine learning/AI is not only well-funded, but is also being held accountable to deliver tangible business results quickly with clear KPIs involving model development, model deployment, model monitoring, model governance, and financial impact. Ultimately, leadership must spend significant time empowering machine learning/AI teams to get the information they need from business units while giving them the ability to drive models into business through the existing DataOps, DevOps, security, compliance, and other operational functions.

#3: Deciding on (and implementing) a comprehensive data strategy program *prior to* building an enterprise MLOps capability

Problem:

Machine learning/AI models are only as good as the data and simulations that are used to develop them. This has driven many enterprises to assume that a comprehensive data and DataOps strategy must be developed and implemented prior to serious investments in machine learning/AI and MLOps capabilities. However, with the involvement of skilled data scientists, and solely on the merit of feature generation, algorithmic and learning capabilities, modern machine learning/AI models can produce significant business impact from existing datasets found throughout the enterprise. Choosing to serialize investments in data prior to developing MLOps capabilities will likely lead to losing ground to competitors.

Recommendation:

Machine learning/AI processes are likely to become one of the most critical consumers (and producers) of data in the long run. Building out machine learning/AI models in conjunction with a data/DataOps strategy allows for continuous feedback and improvement to the DataOps capabilities, including requests for specific data sources that meet critical modeling needs which can drive business value. This guidance on structuring, curating, and delivering data to aid in the overall machine learning/AI lifecycle from development through operational relevance provides key requirements for your enterprise data strategy, and should be done in lockstep with any DataOps program.

#4: Trying to run machine learning/ AI models through an existing DevOps workflow

Problem:

Models are not software. Agile and DevOps delivery capabilities, however mature, are unlikely to work well with the life cycle of a machine learning/AI model as a result. Indeed, the majority of a model's life cycle has little relationship to the typical life cycle of enterprise software assets. Data Science is experimental in nature, which is fundamentally different from software development, and the testing of machine learning models requires rigor and processes which are often not applicable in typical software development. Trying to fit machine learning/AI models into your existing DevOps workflow can be a costly and time-consuming mistake, negatively impacting business metrics depending on those models.

Recommendation:

Treat your data science program, and the models it delivers, as first class and unique enterprise assets. Ensure that the enterprise does not view the machine learning/AI model lifecycle as a special case of your existing SDLC processes, and demonstrate to stakeholders the differences within the various stages. Develop an independent and complete MLOps capability with clear relationships and accountability to existing agile and DevOps capabilities, while accounting for the unique security, operational, and governance needs of an machine learning/AI model throughout its lifecycle.

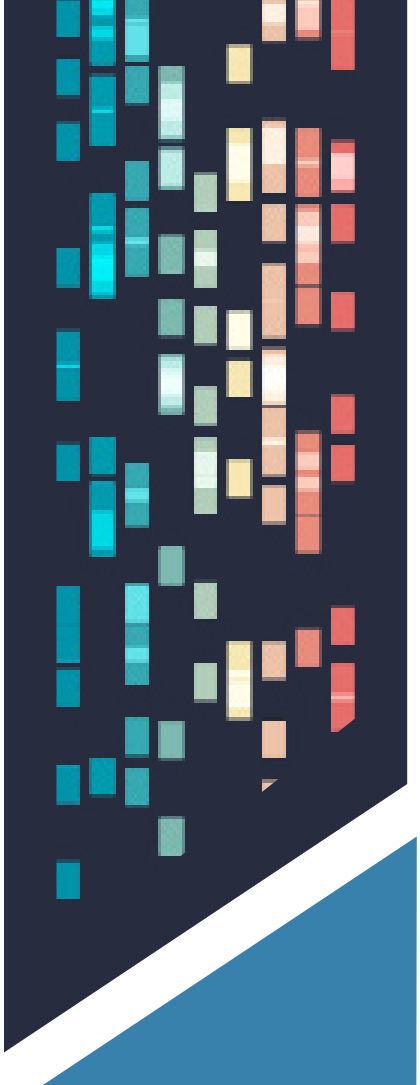
#5: Confusing “data-driven” business intelligence with “model-driven” data science

Problem:

Many organizations have spent a decade or more investing heavily in business intelligence and big data solutions in service of achieving a “data-driven” culture. As a result, the enterprise may be unaware of, or unable to fully understand, the significant differences (in both technique and intended results) between the investments in BI and investments in machine learning/AI as they work toward a more “model-driven” culture. Not only are machine learning/AI models more sophisticated and efficient at extracting actionable insights from data, but they are extremely well-suited for very cost-effective integration into business workflows that operate at far larger scales than the typical users of a BI platform. Failure to recognize the differences between BI and machine learning/AI results in fundamental barriers to investment and attention as the organization moves through its machine learning/AI journey.

Recommendation:

Treat your data science program, and the models it delivers, as first class and unique enterprise assets. Ensure that the enterprise does not view the machine learning/AI model lifecycle as a special case of your existing SDLC processes, and demonstrate to stakeholders the differences within the various stages. Develop an independent and complete MLOps capability with clear relationships and accountability to existing agile and DevOps capabilities, while accounting for the unique security, operational, and governance needs of an machine learning/AI model throughout its lifecycle.



The business benefits of a long-term approach to data science, analytics, and MLOps

Data science is certainly at the top of many business leaders' minds at the moment. It seems that almost daily, headlines announce things like the increase in the number of firms that are investing in AI and analytics, or about the trillions of dollars in "business value" that experts say will be created in the next decade by these technologies. On the other hand, other experts tell us that a shocking number of data science projects (50-85%, depending on who you ask) never make it to production.

These two threads don't seem to connect; how can so much business value be created if the vast majority of these efforts never make it to production?

Let's examine some trends in more detail, and explore an approach specifically designed to help businesses build and deploy data science, analytics, and MLOps solutions that provide impactful results. Let's dig into why projects like these can be difficult without the right support.

Business leaders are now IT buyers

While this observation will likely seem obvious, there are some interesting market forces that make addressing this challenge more complex than it may seem.
First, the last several years have been dominated by a tectonic shift in the technology landscape: the transition from on-premise systems to the cloud. In fact, the use of distriabuted computing resources has become so commonplace that it almost seems trite to call it “the cloud” at all; it’s simply just how technology solutions are delivered in the 2020s. This transition has changed how technology investment decisions are made and, more importantly, who is making those decisions. Business leaders (think sales, marketing or customer service executives) are now IT buyers, thanks to the ubiquity of platforms like Salesforce. In turn, these platforms have advanced to the point that they make previously challenging parts of IT projects easy; security concerns, platform scalability, and ongoing platform support are simply part of the price of admission for these business IT buyers. In addition, they have now been trained that implementing these platforms is relatively easy, and requires little ongoing maintenance.

Data science projects are different

What's more, these new IT buyers have never had to think about their data specifically. In the past, teams have had to make sure that the system captures the data needed to feed a particular business process, and the kinds of data collected would only change when the underlying process changed. Data science projects demand a different way of thinking about your information.

Here's a simple example: a picklist of lead sources that included "Other" as a choice would have been perfectly acceptable in the past. However, if half of your team selects that value when entering data into the system, it becomes meaningless from a data science point of view. A model that predicts "Other" as the most important lead source is not helpful for a business user. "Other" simply isn't actionable.

Data changes as quickly as business conditions change, which is certainly faster than your business processes are updated. If you don't maintain a predictive model as your data changes, those predictions will be based on old data, which will in turn cause those predictions to be equally outdated and therefore of diminishing value.

Businesses require these new (and difficult-to-find) skills

At first glance, this is also pretty obvious: data science projects require data scientists, and people with those skills are in high demand. However, the challenge is even greater than that. In order to implement a successful project, you need a team of people with complementary skills that support your data scientists:

Data engineering: There is a significant amount of effort required to extract and manipulate data to prepare it for easy analysis by a data scientist. Having this skill on your team is much more efficient.

Analytics/visualization: The output of a predictive model is often difficult for a person to interpret without help. What does a score of 0.8 mean in a model that predicts customer attrition risk? Is that good or bad? Resources with good analytics and visualization skills can take these outputs and transform them into actionable insights.

Business analysis: Your team must also be able to understand what's going on in your business. Without this insight, these projects often turn into "science experiments" — models or dashboards that have very little tangible business value or impact.

Platform configuration/customization: You'll also need to have resources that can make necessary changes to the underlying platform to gather more or better data, or to display the right information in the right places to allow your users to take action.

Complicating this challenge even further is the fact that you need all of these resources working together in order to deliver a successful solution. If you lose any one of these capabilities, your ability to support your data science investment will be hampered.

Ongoing support and enhancements to help you achieve your goals and get more value from your data science investments

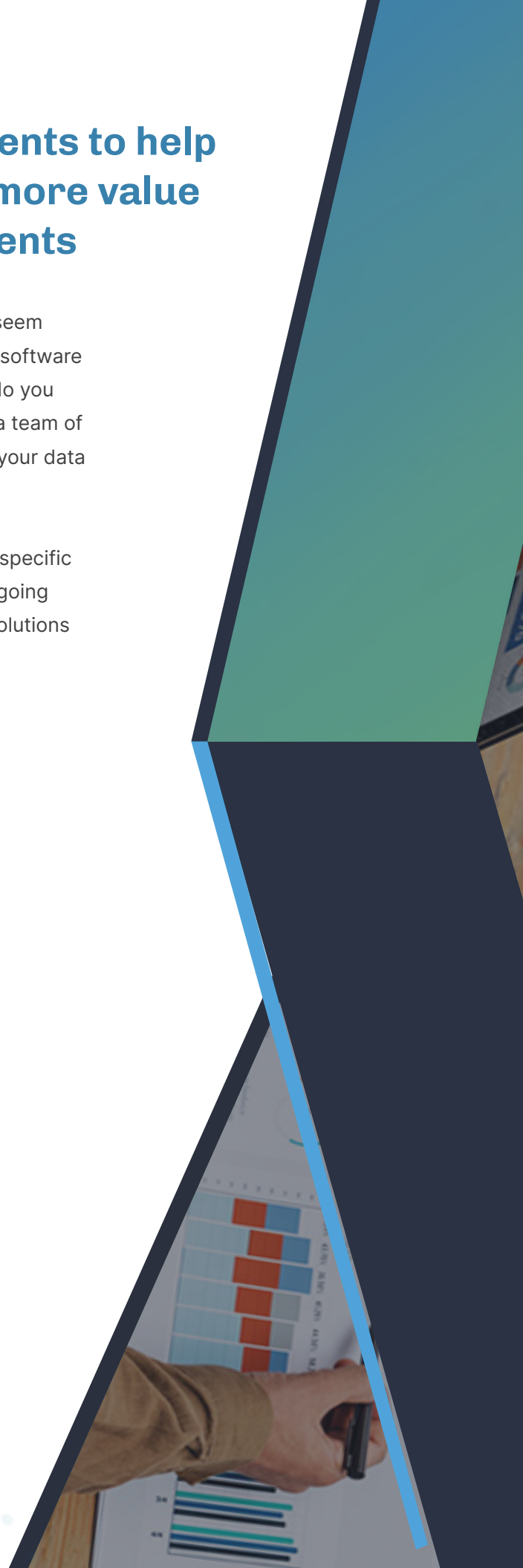
All of these challenges can make an investment in data science seem daunting, but these projects are simply different from traditional software projects; you cannot just make a quick-hit investment. Not only do you need to consider your data and data strategy, but you will need a team of people with expensive, difficult-to-find skills in order to support your data science or analytics investment.

Atrium's Elevate service offering was designed to address these specific challenges. Through Elevate, Atrium provides customers with ongoing support and enhancements of their data science and analytics solutions with dedicated resources with the essential capabilities.

Continuity. Our Elevate teams are organized into "pods"; each pod contains all of the skill sets required to support our customers: business analysis, data science and analytics, platform skills, and quality assurance. Each pod then supports a subset of our customers, which allows our teams to become deeply familiar with each of the environments they support. This allows us to work efficiently and provides our customers access to a dedicated team.

Flexibility. The pod structure also allows us to be flexible with work demands. If one customer needs more data science and less analytics in a particular release, we can simply redistribute work to different members of the pod.

Innovation. Our flexible approach to work enables innovation for our customers. We're able to work on what's most important and impactful to our customers at any given time. Our engagement leads work with our customers throughout the engagement to identify and prioritize the efforts and resources needed to deliver against our customer's highest priorities.



Are you ready for MLOps?

Whether you're well on your way to building out and operating several MLOps use cases or just getting started thinking about the role of MLOps in your organization and what it takes to succeed, it's important to have the right partners, approach, and access to flexible, scalable skill sets.

Reach out to us to discuss your needs and find out how we can advise and extend your MLOps strategy and solutions.

Contact Us

1 800 674 3103

info@atrium.ai

www.atrium.ai

About Atrium

Founded in 2018, Atrium is a next-generation consulting services company that helps organizations across diverse industries deliver on the promise of analytics and AI. As the market leader in intelligent solutions, we help organizations make smarter decisions and act on them. Learn more at *atrium.ai*.