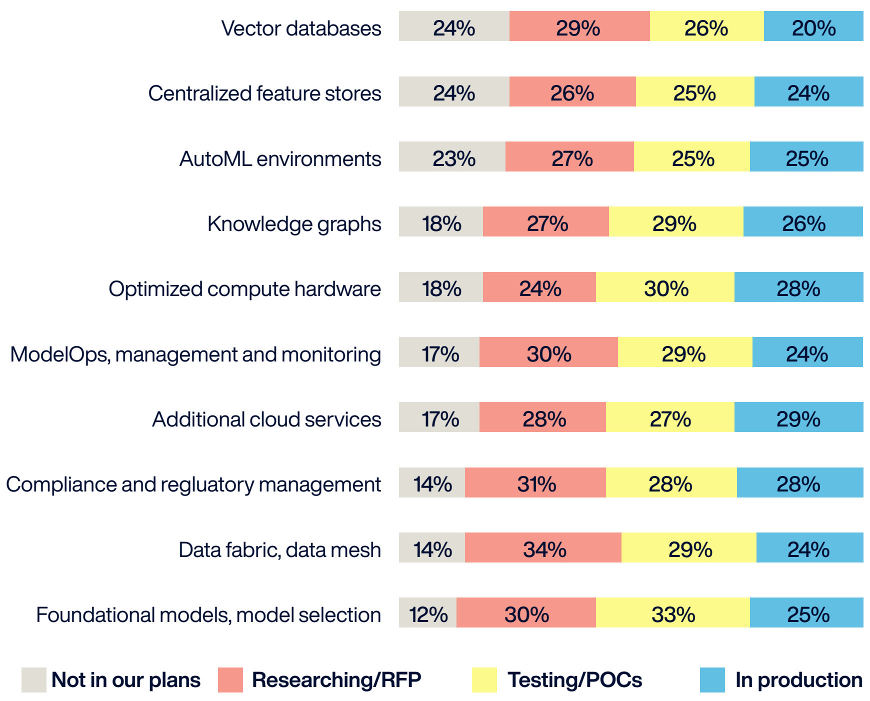
Achieving high AI readiness

10 key factors for your organization

Aerospike

Infinitely possible.™



Survey: What is the status of the following AI technology in your existing environment?

Introduction

Is your organization actively implementing AI? Planning to do so? Are you ready for it? According to a recent survey by BARC IT Market Strategy, only 20.5% of respondents could be characterized as being in a state of “High Readiness” to implement AI.

As the following chart from the study shows, a common cause of unreadiness is a lack of familiarity with the AI technology stack. This ebook explains each of the AI technical stack components, covering key concepts and evaluation criteria. You’ll gain the understanding required to make AI tech stack choices that are best suited for your business or use case.

This survey makes it clear that many companies seeking to implement AI might not even be familiar with the technology stack AI applications require, let alone be ready to incorporate them into their organization for a successful AI implementation.

That’s what this e-book will tell you.

1 Optimized compute hardware

Whether it's running your own on-premises servers or using cloud services, the first step is ensuring you have the computing power necessary to run the AI software you need.

Tailored hardware for AI workloads

It's easy to think of just throwing hardware at the problem, but that's not necessarily the best solution. First, that much hardware is going to be expensive to buy or obtain through cloud services. Second, it will be expensive to operate in terms of electricity and cooling.

AI-optimized hardware is computer hardware designed to perform AI tasks more efficiently than general-purpose hardware. It can improve speed, efficiency, scalability, flexibility, and cost-effectiveness. For example, graphics processing units (GPUs) can be used for both AI training and inference and have thousands of cores that can perform parallel computations.

The downside of using a GPU is that it can be expensive and scarce. Consequently, companies also look for alternatives.

Other hardware that can be used for AI includes:

- Central processing units (CPUs), general-purpose chips that can run AI algorithms
- Intel Xeon W and AMD Threadripper Pro
- CPU platforms that offer reliability, PCI-Express lanes for GPUs, and good memory performance
- 5th Gen Intel Xeon processor offers AI acceleration and up to 42% better AI performance than the 4th Gen
- NVIDIA Jetson Nano, a compact AI single-board computer that's suitable for developers, researchers, and enthusiasts
- AI teams can also optimize the performance of deep learning models by using the right algorithms and co-designing hardware and software. This is known as hardware-aware optimization.

Finally, there's the sustainability aspect. Training several AI models releases up to 626,000 pounds of carbon dioxide into the environment – five times that of your car. And that's just the beginning. AI models demand a constant stream of feeding and tuning, and the hardware required to sustain them adds to the carbon load.

With many organizations having environmental, social, and governance (ESG) goals, it's important to consider not just the cost to your company but also the cost to the planet.

Navigating the hardware landscape

So, what do you do?

When choosing software, you look for servers that can be made efficient and consider hardware efficiency. The right database software can still offer the same performance as an in-memory product without requiring the investment in RAM. It can also reduce carbon emissions by up to 80%, reduce the number of servers required by ten times or more, and lower hardware, power, and cooling costs.

Cost vs. performance considerations

Finally, consider the total cost of ownership (TCO), including operational and capital expenses. For example, how will your database licensing costs increase as your business grows? Will you need to acquire additional databases for other functions?

Evaluating compute hardware

Here are some of the hardware factors to consider when you plan to run AI:

- Do you need specialized processor chips, such as a graphics processing unit (GPU), which provides more cores for parallel processing?
- If so, which ones do you need? How big? How many? Two major GPU vendors are NVIDIA and AMD, each with advantages and disadvantages depending on your use case.
- How much RAM do you need for acceptable performance? This is particularly important if you're going to use an in-memory database. Check with your vendor about the recommended amount of RAM, as well as bandwidth and memory hierarchy concerns.
- Don't neglect communications, especially if your AI system is centrally located and collecting and sending data to the edge. Insufficient communications hardware can introduce latency issues. In fact, depending on how much data is generated and received, you might want to consider edge AI processing.
- How many nodes and other infrastructure will you need, and how many staff members will be needed to monitor and manage it all?
- Finally, it's important to ensure that your AI hardware is flexible and extensible so you can add more memory or processing power as needed.
- Are you running AI on a cloud service? If so, you should check with your provider about these factors.

2 Foundation models and model selection

While some people tend to think of AI, GenAI, and ML as interchangeable, they are actually three separate concepts.

- Artificial Intelligence (AI) refers to the broad field of creating machines capable of performing tasks that typically require human intelligence, such as reasoning, learning, and problem-solving.
- Generative AI (GenAI) is a subset of AI focused on creating new content, such as text, images, or music.
- Machine Learning (ML) is a subset of AI that involves training algorithms on data to learn patterns and make predictions or decisions without being explicitly programmed for those tasks.

AI, and consequently GenAI, are based on a model. Think of the model as the "operating system" for AI. Generally, AI applications interact with the model through application programming interfaces (APIs), just as your applications interact with an operating system through APIs. That lets you update your model without having to rewrite all your existing AI applications.

Understanding foundation models

A particular subset of AI models is the foundation model. Neural networks are generally trained on massive datasets. The dataset type and the neural network features help determine the foundation model's capabilities and features.

For instance, chances are you've heard of ChatGPT. In that particular case, ChatGPT is the application, and GPT, a particular kind of foundation model known as a large language model (LLM), is the foundation model that powers it. Over the past few years, the GPT LLM has evolved from GPT-1 to GPT-4. Each version of GPT is based on a larger, more complex neural network that has been trained on a larger number of parameters, giving each generation more capabilities.

The art of model selection

In the same way that you could write your own operating system if you wanted to, to ensure it met your specific needs, you could also develop your own foundation model. But chances are, you don't want to. It takes time and expense to develop one, and there's plenty out there in the market already. It's likely that an existing foundation model will meet your needs or can be augmented by an additional domain-specific model.

Like choosing an operating system, different foundation models have different features and benefits. Depending on the applications you want to run, the features you want, and the performance you need, you might choose a Linux, Windows, or Apple operating system to run on your hardware. Similarly, the applications and features you want will help determine which foundation model you use.

Predicting the next wave of AI innovation

With the massive interest in AI and the great amounts of funding being thrown at AI companies, foundation models will undoubtedly continue to grow and evolve. Existing models will continue to support more complex neural networks with more parameters, while new models with different specialties will be developed.

Evaluating models

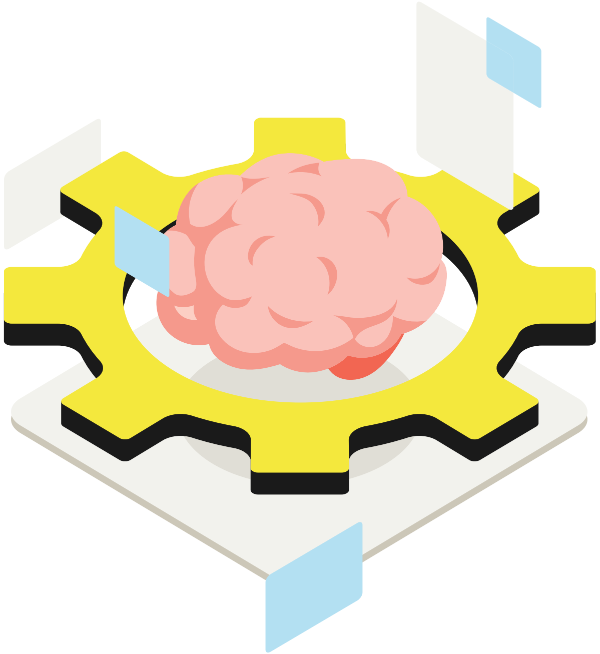
It's easy to assume that you'll be using one of the most popular models, but that isn't necessarily the most effective solution for your business. Here are some other factors to consider.

What do you expect your AI system to do?

What existing models meet that need?

What domain-specific models exist in your industry that could augment another model?

Are you planning to manage your models in-house, with a third-party tool, or as a hybrid of the two?



3 AutoML environments and tools

In the same way that AI can make everyone else's life easier, AI can make the lives of data scientists easier by giving an ML system more control over what model it uses. This is called automated machine learning or AutoML.

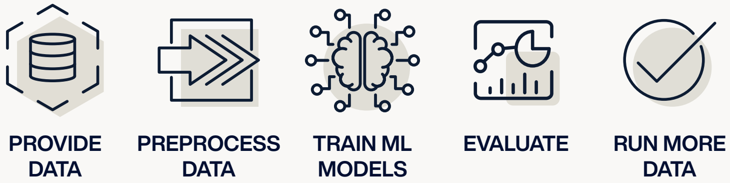
AutoML: Bridging the skill gap

In addition to making the lives of data scientists easier, AutoML's advantage is that it helps people who aren't experts create AI models. That way, organizations that haven't yet found or hired a data scientist can still take advantage of the benefits of machine learning (ML).

Streamlining model development with AutoML

Of course, AutoML tools vary, but generally, they work as follows:

- The user provides the tool with a cleaned dataset
- The tool performs a certain amount of preprocessing on the data
- The tool trains a variety of ML models on the data
- The tool then evaluates the results to let the user select the most appropriate one
- The user runs future data on the ML model



Evaluating AutoML tools

- In many ways, choosing an AutoML tool is like choosing any other product.
- How much does it cost?
- How easy is it to use?
- What sort of support does the company offer?
- How reliable is it?
- How customizable is it?

Beyond that, there are specific criteria to help choose an AutoML tool.

- Does it support a broad range of features or fewer features but with more in-depth coverage?
- How transparent is its decision-making process?
- How well can the system explain why it made its choices (also known as explainability)?

4 Feature stores

As the number of features used in ML models grows, identifying and managing them becomes more expensive. That is why many firms are buying or creating feature stores, which act as central repositories where features can be stored, shared, and served online or offline.

What are feature stores?

AI/ML applications rely on "features," relevant attributes about an event or phenomenon, to help predict outcomes. These feature stores can be offline or online. According to Feature Stores: A Hierarchy of Needs, feature stores offer the following characteristics:

- Access to feature information, data, transparency, and lineage
- Availability in production at high throughput and low latency
- Minimizing train-serve skew, point-in-time correct data, and monitoring
- Easy and quick to use, intuitive APIs, interactivity
- Automated backfilling and alerts, and feature selection

Overcoming data bottlenecks

Organizations such as Uber have stated that managing features is the biggest bottleneck in productizing their ML models. That no longer needs to be the case with the right feature store.

Evaluating feature stores

If you're going to depend on an off-the-shelf product for your feature stores, here are the factors it needs to have:

- Low latency
- Flexible memory and storage management architecture suitable for online and offline feature store support
- Self-healing and self-managing features for high availability
- Integration with popular AI/ML tools, streaming platforms, messaging systems, and legacy data infrastructures
- Massive parallelism and deep exploitation of advanced hardware and network technologies for fast, predictable runtime performance at scale

5 Vector databases

Popular AI applications involve inferring relationships or similarities between unstructured data. For example, if you want to show a client a product similar to the one they just clicked on, whether it's a movie or shoes, how do you do that? Often, it involves a vector database.

The concept of vector processing in AI

A vector database provides for efficient storage, swift retrieval, and processing of structured and unstructured data—including text, images, audio, or video—at scale to make search easier and more efficient. Vector databases don't store the actual data itself; rather, they're numerical representations of the data, known as embeddings. The embedding process takes different types of content and converts them into vectors.

Vectors are the data structures behind information retrieval. A vector translates source information into a compact high-dimensional (HD) numerical value. The more dimensions you include in your embedding, the more detail you can maintain on each piece of information.

Vectors can embed information of all types, from text and numbers to images and music. Once created and put into a database, vectors can be used for AI applications, such as real-time recommendation systems and content summarization.

Real-world use cases and benefits

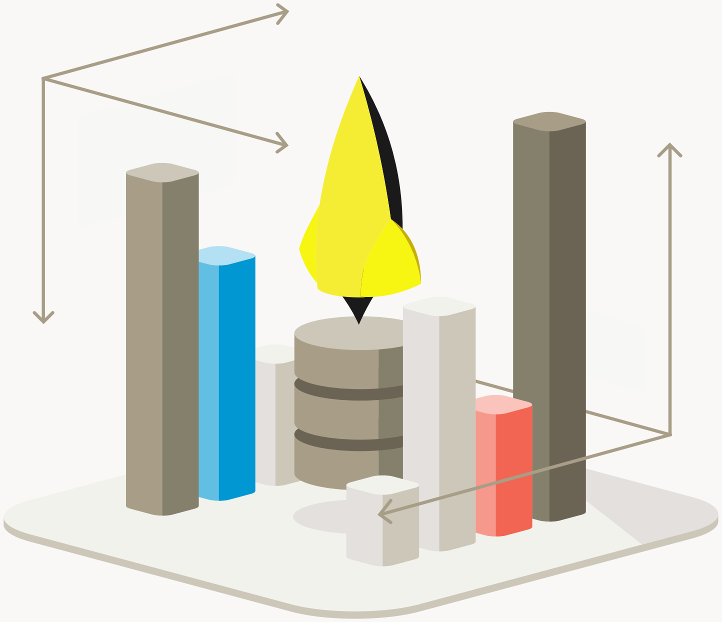
Here are some examples of how vector databases are used in AI applications:

- Customer 360 / Hyper-personalization: Vector databases sift through large amounts of unstructured data about a customer's entire history, then spot and act on patterns in real time, making recommendations when customers need them.
- Fraud prevention: Fraud prevention systems use pattern recognition to sift through dozens of data points to detect anomalies that signal fraud and then react to those suspicious patterns as they happen.
- Large language models (LLM): A lot of what we find so amazing about LLMs are actually powered by vector databases. When you type a question into an LLM, it typically will convert it into a vector to help determine the answer. It typically also stores the ongoing conversation as vectors to help the LLM remember what it's talking about from one input to the next.
- Retrieval augmented generation (RAG): Beyond the general use case of vector databases for LLMs, RAG adds the specific domain knowledge important for accurate and contextually relevant responses in business settings to a more general LLM.

Evaluating vector databases

As vector databases have become more associated with AI, more vendors are touting their database products as “vector databases” to make them more attractive to this burgeoning market. Here are the factors you need to consider.

- How much data can the vector database handle? Remember that some AI use cases require terabytes of data.
- What is the memory structure of the vector database? For example, an in-memory vector database will likely have better performance but will be limited by the size of the memory you have—or are willing to pay for.
- Does the vector database support horizontal scaling, where you or your cloud provider add commodity hardware to support bigger jobs?
- How do infrastructure requirements, including processors as well as memory, grow as the database grows? This can become costly.
- What is the vector database's latency? While this seems small, it adds up, especially with compute-intensive tasks, as is the case with fraud detection.



6 Knowledge graphs

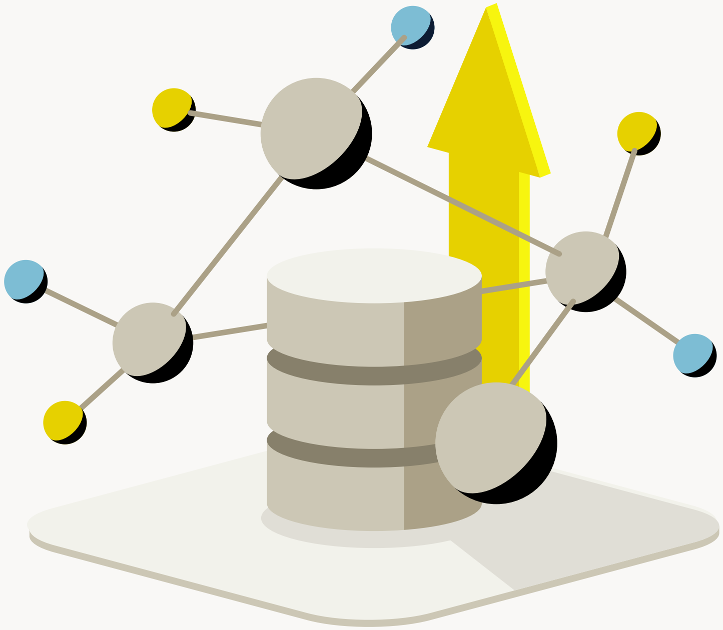
A great deal of AI is predicated on what it knows and can then infer about the relationships between items. Consequently, an important component of the AI tech stack is the knowledge graph, which depicts known relationships between items.

Knowledge graphs explained

A knowledge graph has two basic components: a data point, also known as a node or a vertex, and the relationship between data points, known as edges. For example, take a social media platform such as Facebook; on that platform, the data points are the people, and the relationship between them is a friend or friend of a friend.

Building connected data for AI

A type of database specifically designed for knowledge graphs is called a graph database. Graph databases can serve several functions in an AI/ML data pipeline, such as in GenAI. Graph databases can help model different entities (users, customers, households), feed data to train ML models, and provide graph algorithms to ML use cases (PageRank, similarity matching, community detection).



Case studies: Graph databases in action

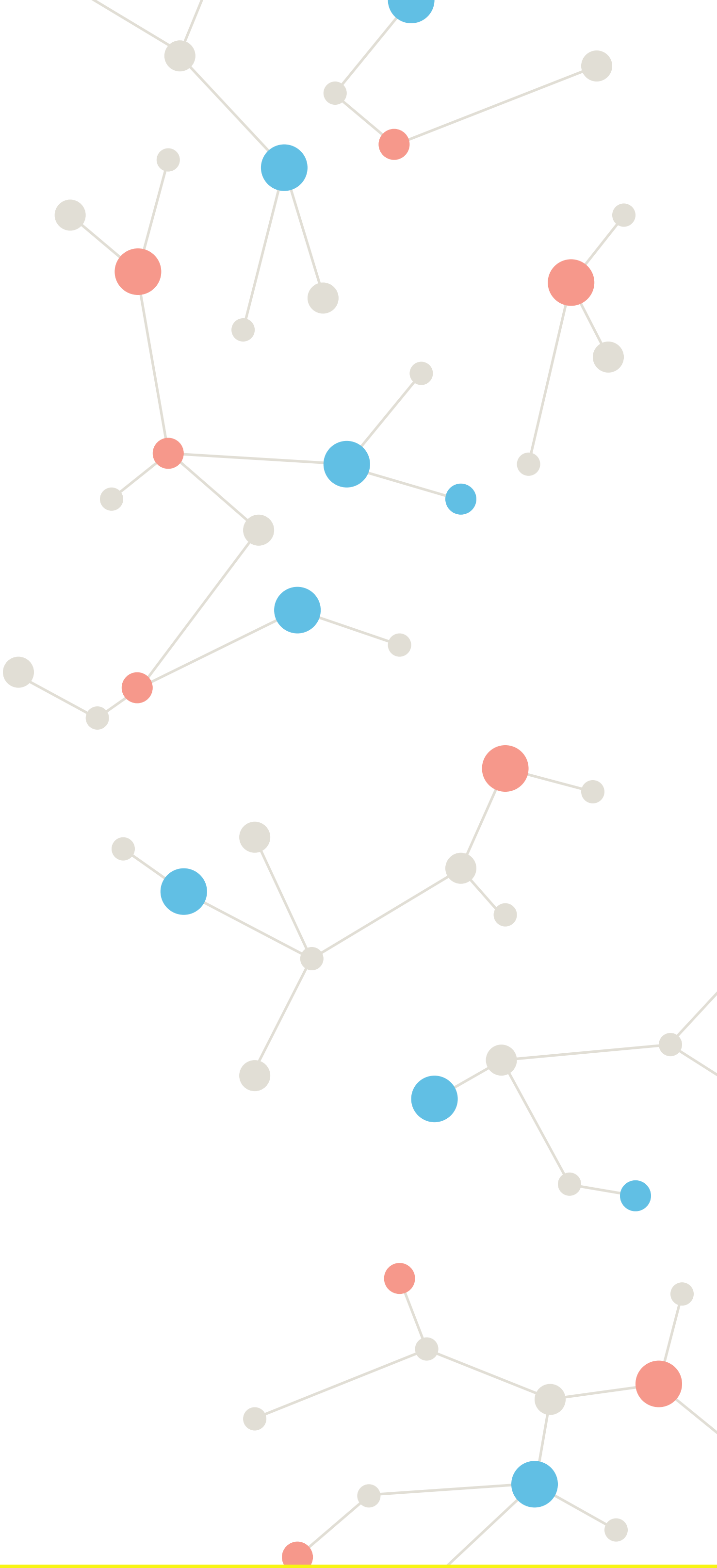
There are two basic use cases for graph databases: analytical purposes (graph OLAP) and transactional or operational purposes (graph OLTP).

Online analytical processing (OLAP) is typically used to perform multidimensional data analysis, deriving meaning through calculations and visualizations of aggregated historical data. These include real-time exploration of buyer patterns, related purchases, and connections to similar products; pricing analysis, such as competitive and dynamic pricing; and measuring influential nodes, such as page ranking.

Online transaction processing (OLTP) is typically used for high volumes—thousands or millions of concurrent users or devices—of short transactions and fast—sub-10 millisecond response times—query processing. These include AdTech for displaying ads in near real time; recommendation engines for displaying recommendations, also in near real time; and fraud detection, looking for and detecting suspicious transaction patterns.

Evaluating graph databases

- Is the graph database in memory? Particularly for OLAP use cases, that puts an upper bound on its scalability.
- Does the graph database, particularly for OLTP use cases, provide the fast throughput and low latency these applications require?
- What is the total cost of ownership for the graph database? For example, an in-memory graph database requires a large amount of memory as the database grows in size. Even a graph database that isn't in memory can require many nodes and associated infrastructure, as well as the personnel to manage it all.



7 ModelOps/ MLOps

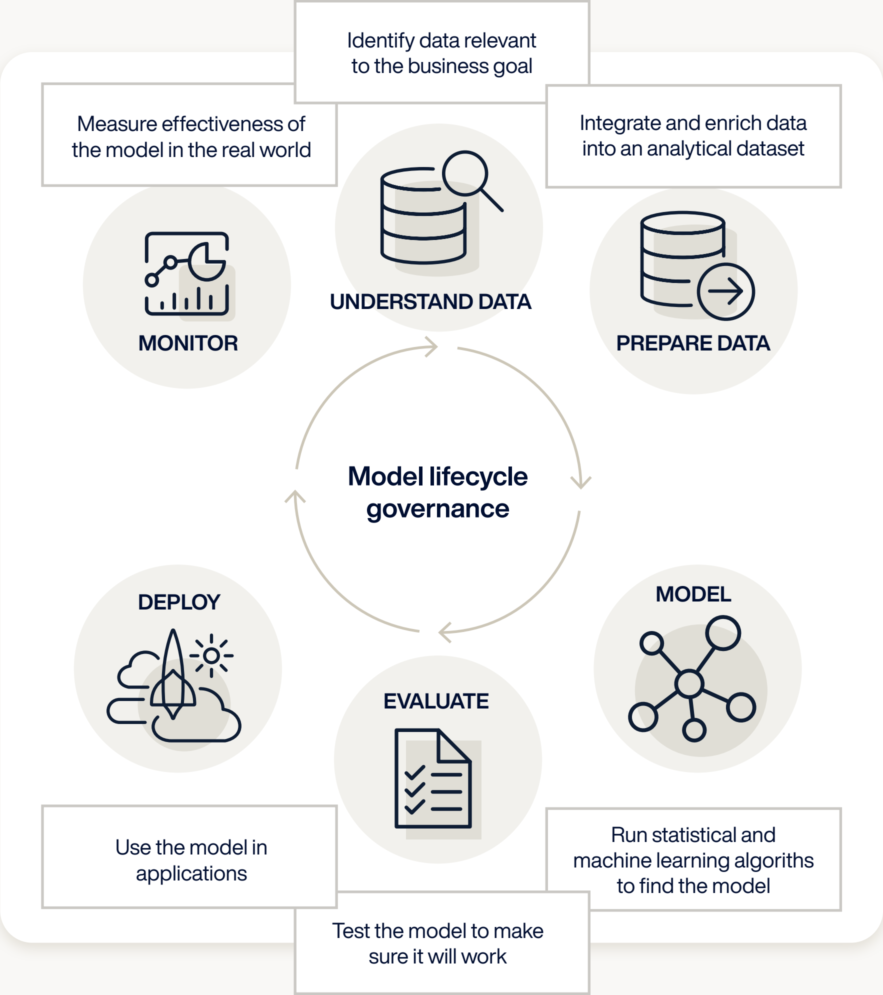
You've probably heard of DevOps, which GitHub defines as combining "development (Dev) and operations (Ops) to increase the efficiency, speed, and security of software development and delivery compared to traditional processes." ModelOps, sometimes known as MLOps for machine learning operations, is like that, except for AI/ML models. It's intended to help data scientists and businesses work together better.

The ModelOps framework

According to Forrester, the ModelOps framework is a constantly evolving circle of:

- Identifying data relevant to the business goal
- Enriching the data into an analytical data set
- Running statistical and ML algorithms to find the model
- Testing the model to ensure its success
- Using the model in applications
- Measuring its effectiveness in the real world
- Returning to identifying data

The ModelOps framework isn't that different from any other continuous improvement cycle—just applied to AI/ML models.



Identify data relevant to the business goal

Measure effectiveness of the model in the real world

Integrate and enrich data into an analytical dataset

UNDERSTAND DATA

MONITOR

PREPARE DATA

Model lifecycle governance

DEPLOY

MODEL

EVALUATE

Use the model in applications

Run statistical and machine learning algorithms to find the model

Test the model to make sure it will work

Best practices for model management

No matter what kinds of AI models your organization uses, a few practices are universal.

- **Define your objectives:** What problem are you trying to solve? What data do you have? How will you know whether you're successful?
- **Follow a systematic and iterative process for deploying a model.** This includes setting up a regular schedule for retraining on updated data, a phased rollout, and a rollback plan in case of errors.
- **Test before putting it into production:** This includes doing A/B testing for results and checking for implicit biases.
- **Monitor the results continuously:** That way, you'll know when additional retraining is necessary or when the model should be decommissioned. That includes checking for user response.

Monitoring AI models at scale

For an AI/ML project to be effective, it must operate at production scale—one of the obstacles to deploying such projects. Monitoring results continuously, beyond just the initial test phase, is important because sometimes issues show up at scale that didn't appear in test environments, such as biases, anomalies, outliers, or "drift" caused by data different from the kind on which the model was tested.

And simply scaling itself might be an issue. A model that works fine on a smaller amount of data might choke on production levels or might require manual scaling rather than automatic scaling. You want to find that out before the model goes into production, so be sure to test it on large datasets in case you need to add more hardware resources. You also want to ensure that your modeling system itself scales to the size and performance you need so you don't miss out on problems.

Evaluating ModelOps/MLOps tools

Whether you call it ModelOps or MLOps, the point is to have a tool to help you evaluate and manage AI and ML model operations. But how do you evaluate the ModelOps/MLOps tools themselves? According to Intellerts, a data science consultancy, here are some factors to consider.

- **Flexibility:** Can the tool be easily adopted in multiple situations, meeting the needs for different modeling techniques?
- **Framework support:** Are the most popular ML and deep learning technologies and libraries integrated and supported?
- **Multi-language support:** Can the tool support code written in multiple languages? Does it have packages for the most popular languages used by data scientists, like R and Python?
- **Multi-user support:** Can the tool be used in a multi-user environment? Does it meet security requirements?
- **Maturity:** Is the tool mature enough to be used in production? Is it still developed? Is it used by any large companies?
- **Community support:** Is the tool supported by any developer groups or backed by large companies? Does it have a commercial version?

Also, it should go without saying, but any ModelOps/MLOps tool you consider should work with your framework and platforms. Not all tools work with all frameworks and all platforms.

8 Managing your cloud AI

As with many other compute-intensive tasks, AI works well with the cloud because it's easier to add additional resources when needed and release them when they're not.

The cloud AI ecosystem

Major cloud providers such as Amazon, Google, and Microsoft sell services specifically intended for AI. For example, Google offers AI Platform, Microsoft offers Azure OpenAI Service, and Amazon offers Amazon Bedrock. Several other vendors also offer AI services in the cloud.

Selecting the right cloud services

There's no single "right" cloud service for AI, and all three major cloud providers offer AI services—in fact, more than 800 between the three providers. Chances are, your organization already has relationships with cloud providers, so using the same cloud provider for AI in the cloud probably makes sense. (At the same time, there is some concern about the influence of the "big 3" cloud vendors on the AI marketplace.)

It also depends on whether your AI system has a specific relationship with a particular cloud vendor. For example, OpenAI, which developed ChatGPT, has a relationship with Microsoft and runs on Microsoft's Azure cloud platform.

And as with tailoring hardware for sustainability goals, it's also important to look at the environmental impact of cloud services. Microsoft has reported that its Scope 3 emissions have risen 30% since 2020, which it attributes to data centers supplying the need for AI. This isn't to pick on Microsoft; cloud service providers Google and Amazon are expected to report similar results. That's why ensuring that your database is architected for efficiency is important as you prepare your organization for a state of "high readiness."

Balancing flexibility and control

As with any cloud application, AI in the cloud has advantages and disadvantages. It offers more flexibility in terms of adding and removing resources, keeping systems updated without needing to do it yourself, and automatically performing maintenance such as backups.

That said, performing these functions in the cloud means giving up control, which can be an issue, particularly for heavily regulated industries, and there is some concern about unauthorized access. In addition, should a cloud system go down, the AI system on which it is based would also be unavailable. Finally, using the cloud introduces some level of latency compared with performing analysis at the same place as the data.

Evaluating cloud AI services

Realistically, your company probably already has a cloud provider, and that cloud provider—particularly if it's one of the "big three" of Amazon AWS, Microsoft Azure, or Google Cloud—provides some level of AI support that's going to be adequate enough that you don't need to switch over to or add a new cloud provider. That said, here are factors to consider before choosing a cloud service for AI.

- If you've already picked out an AI application or model, which cloud services work well with it? For example, Microsoft Azure has partnered with OpenAI, while Google has its own, DeepMind and Gemini.
- If your business requires certain certifications, such as FedRAMP, ensure that the cloud service you are considering provides them.

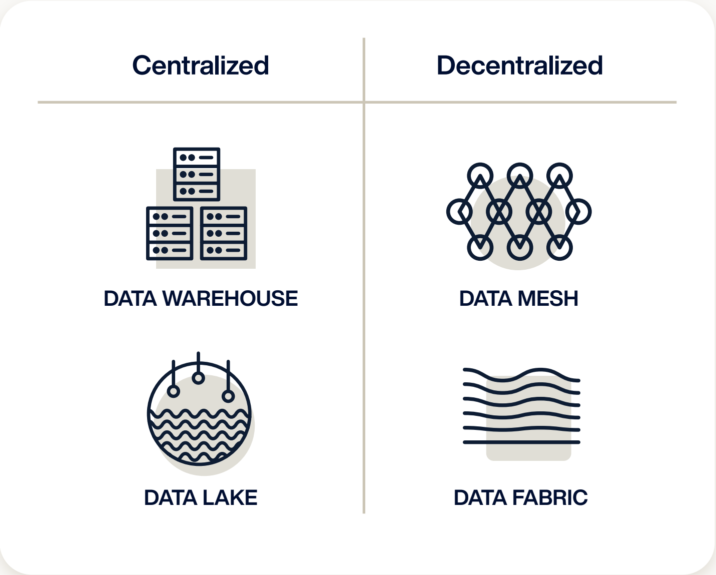
9 Managing data

When you use AI and ML, you'll need data. Chances are, your company has plenty of data. How do you get it to the people who need it? There are several philosophies, each with advantages and disadvantages.

Centralized vs. decentralized data management

Every company has large amounts of data, generally created by different domains within the company. The finance people create and manage financial data. The salespeople create and manage data about sales and customers. The IT teams create and manage data about the company's network and other IT resources. All that data is valuable, but how do you place it where people can access it?

One possibility is putting all the data together into a "data lake" or "data warehouse." The disadvantage is that those often involve moving the data elsewhere, and introduce security problems (what if the single place goes down or is hacked?).



The other possibility is a "data fabric" or "data mesh." The idea of a data mesh is that instead of moving the data to a data warehouse or other repository, the data stays right where it is and is available for access by other people. And because the data is still created and managed by the same people who created and managed it before, the data is in domains rather than an amorphous data lake, making it easier to identify.

In fact, with a data mesh, the data itself becomes a product.

Data fabric and data mesh are not mutually exclusive—in fact, you can have both—but a data fabric is slightly different. A data fabric uses metadata and data management tools to make connections between different data sources without human involvement—at least in that part. However, a more centralized approach is required to ensure the metadata is consistent and useful.

Architecting a scalable data infrastructure

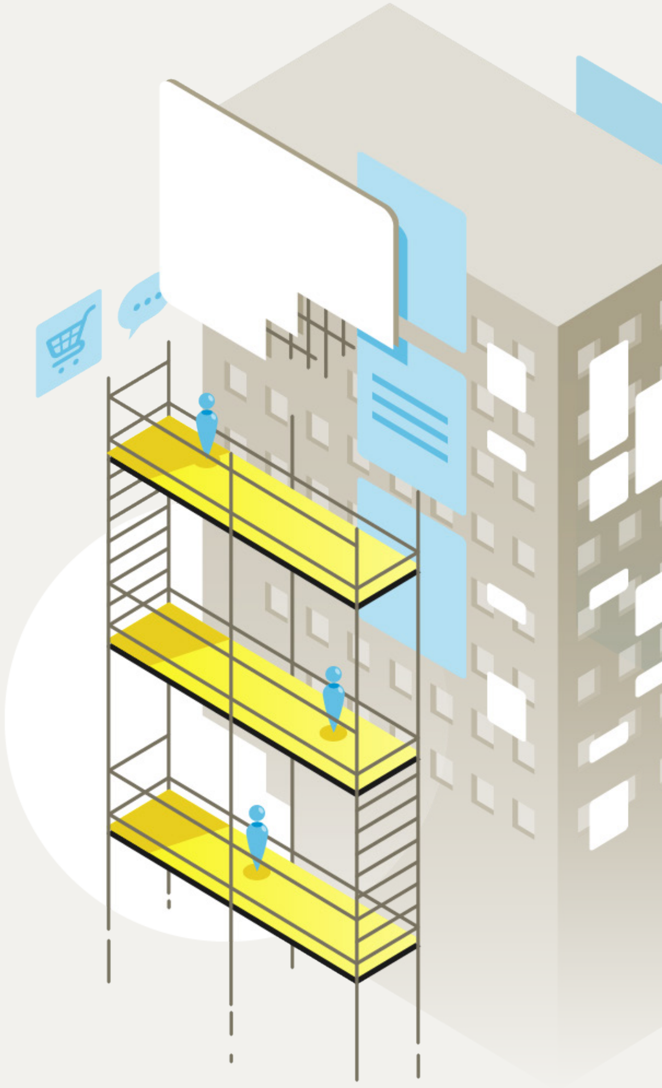
The challenge with having a data fabric or data mesh, particularly for applications such as AI/ML, is that the data needs to be accessible with high throughput and low latency. Suppose you're using AI for customer recommendations. In that case, the application needs to be able to access potentially huge amounts of data, essentially in real time, so it can make those customer recommendations when they're still useful.

Evaluating data management

The idea of moving all the company's data into a data lake or data warehouse may seem attractive, but realistically, assess your company's ability and willingness to move all the data around in a useful timeframe.

Similarly, if you're considering a data fabric, determine how the required metadata is going to be updated and who's going to do it.

Wherever you put the data, is it accessible to everyone who needs it, with high throughput and low latency?



10 Compliance and regulatory management

Finally, your organization must ensure it meets the AI compliance and regulatory requirements for your government(s) and your industry.

Regulatory challenges in AI

The biggest regulatory challenge in AI is that it's a moving target. Ever since AI made a splash with the public introduction of ChatGPT, industries, states, and federal governments such as the United Kingdom, the European Union, and the United States have been instituting new laws and requirements for using AI. This includes restrictions to help prevent "deep fakes," especially in politics, and copyright and privacy restrictions that limit the content people can use to train their AI systems.

Designing for compliance

Like security, the easiest way to ensure AI compliance is to include it from the beginning rather than trying to tack it on at the end. That includes making sure that legal and privacy officers are involved. It also involves regular checks on how the laws and regulations have changed, ensuring that AI systems still meet the requirements.

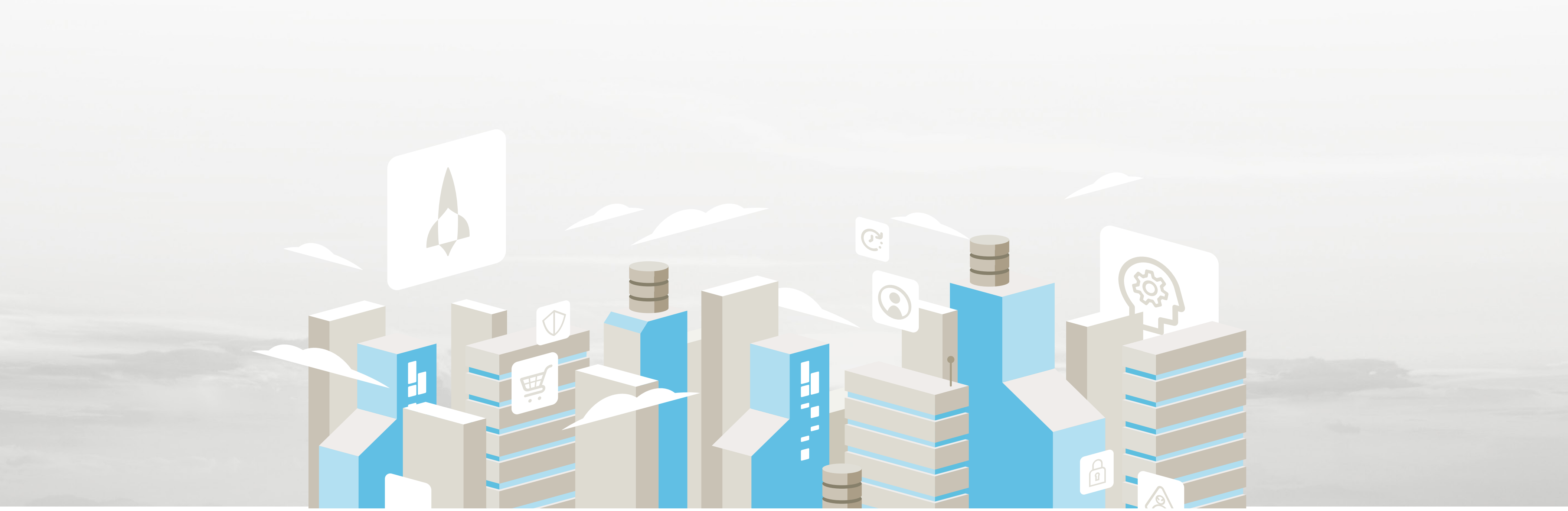
The specifics likely depend on which systems you must comply with. For example, the EU's General Data Protection Regulation (GDPR) focuses on data collection, storage, and protection. Another common target for regulatory compliance is transparency—that is, ensuring that the AI system can explain how it made its decisions.

AI ethics and governance

Beyond legal requirements for AI, there are also ethical and moral issues, particularly related to privacy. Because AI systems are often trained on existing data, many cases reflect existing biases, whether assuming a doctor is a man and a nurse is a woman, not recognizing people of color in photos, or re-instituting "redlining" in a financial services organization. In an effort to reduce such biases, some organizations are turning to AI-generated synthetic data.

Evaluating compliance and regulatory management

- Make sure you're aware of any national requirements around the use of AI, and that you're in a position to find out about any changes.
- Similarly, do the same for state and local requirements.
- Finally, work with the appropriate people in your organization—perhaps the CTO or CFO—to ensure that you're complying with any additional regulations required in your industry.



Ready for high readiness?

The advantage of having a tech stack devoted to AI is that different parts of your company are all working together on AI. What you don't want to have happen is for one department to use one technology while a different department uses another. That introduces silos and limits the power of AI to make business decisions.

AI has the potential to transform organizations and your entire industry. It's up to us to ensure that we do so responsibly.

About Aerospike

Aerospike is the real-time database built for infinite scale, speed, and savings. Our customers are ready for what's next with the lowest latency and the highest throughput data platform. Cloud and AI-forward, we empower leading organizations like Adobe, Airtel, Criteo, DBS Bank, Experian, PayPal, Snap, and Sony Interactive Entertainment. Headquartered in Mountain View, California, our offices include London, Bangalore, and Tel Aviv.

For more information, please visit https://www.aerospike.com.

2440 W. El Camino Real, Suite 100, Mountain View, CA 94040 | (408) 462-2376