

WHITEPAPER

The Modern Data Experience

By Benn Stancil

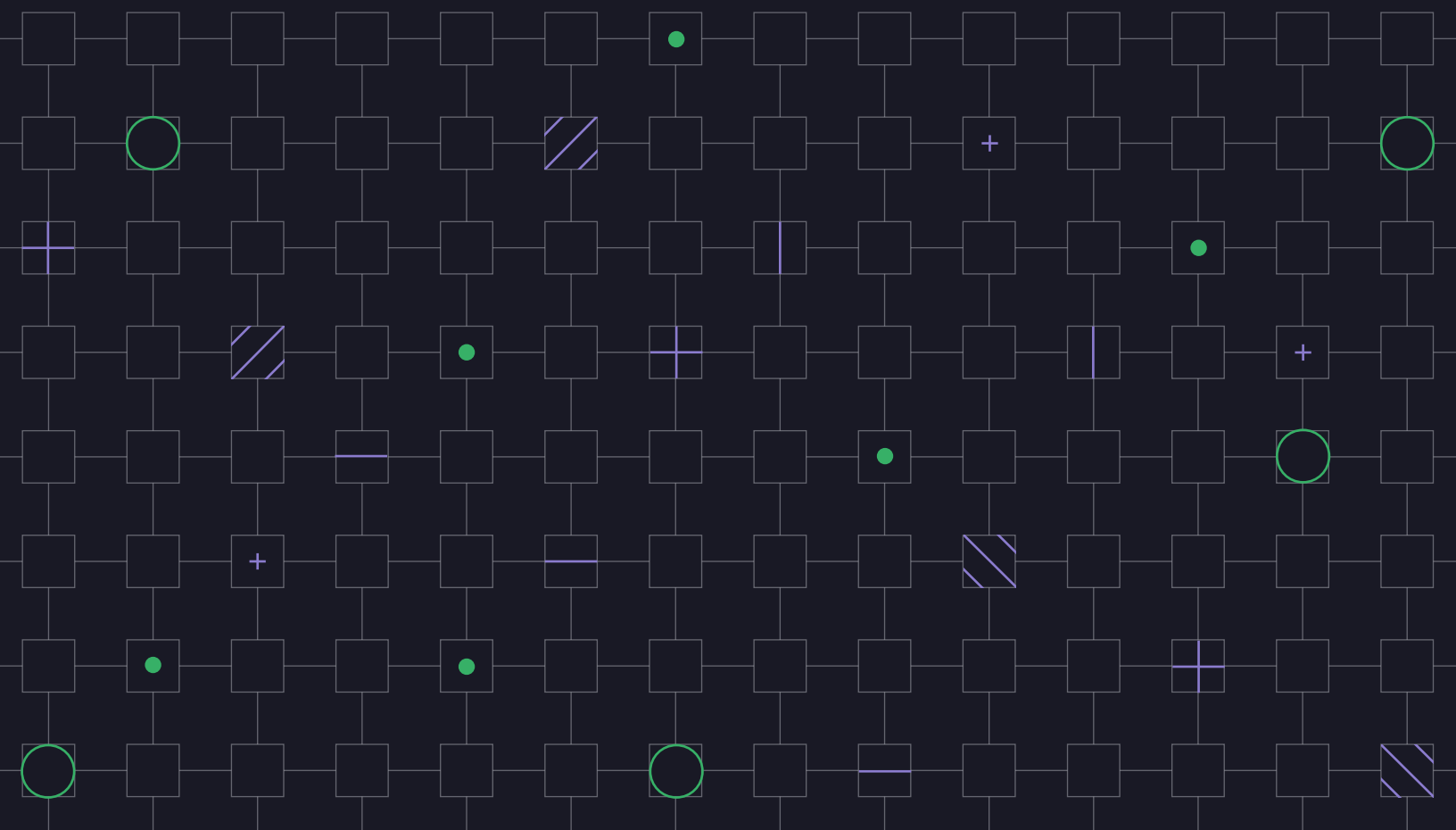


Table of Contents

Introduction	03
An overview of the modern data stack	04
The legacy data architecture	
The new philosophy	
The new stack	
The new software	
The problems with the stack	09
Building a modern data experience	11
The founding principles	
Gathering inspiration from Amazon, Airbnb, and Yelp	
Cloud computing services - designing an integrated foundation	
Internal data stacks - how tools can interoperate	
Yelp and consumer apps - making data inescapable	
What do you do now?	
Conclusion	18

Introduction

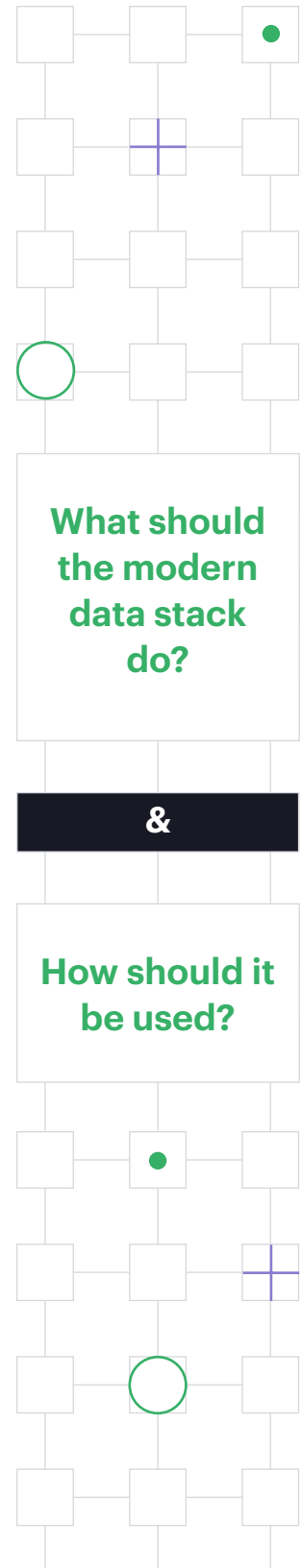
What is the modern data stack?

It's a question we've all heard. As an industry, we've written countless articles and blog posts, sketched numerous diagrams, and had a myriad of conversations about the modern data stack—all describing what it is, what it isn't, where it's going, and how it's evolving. With each iteration, we find more tools to build and more problems to solve.

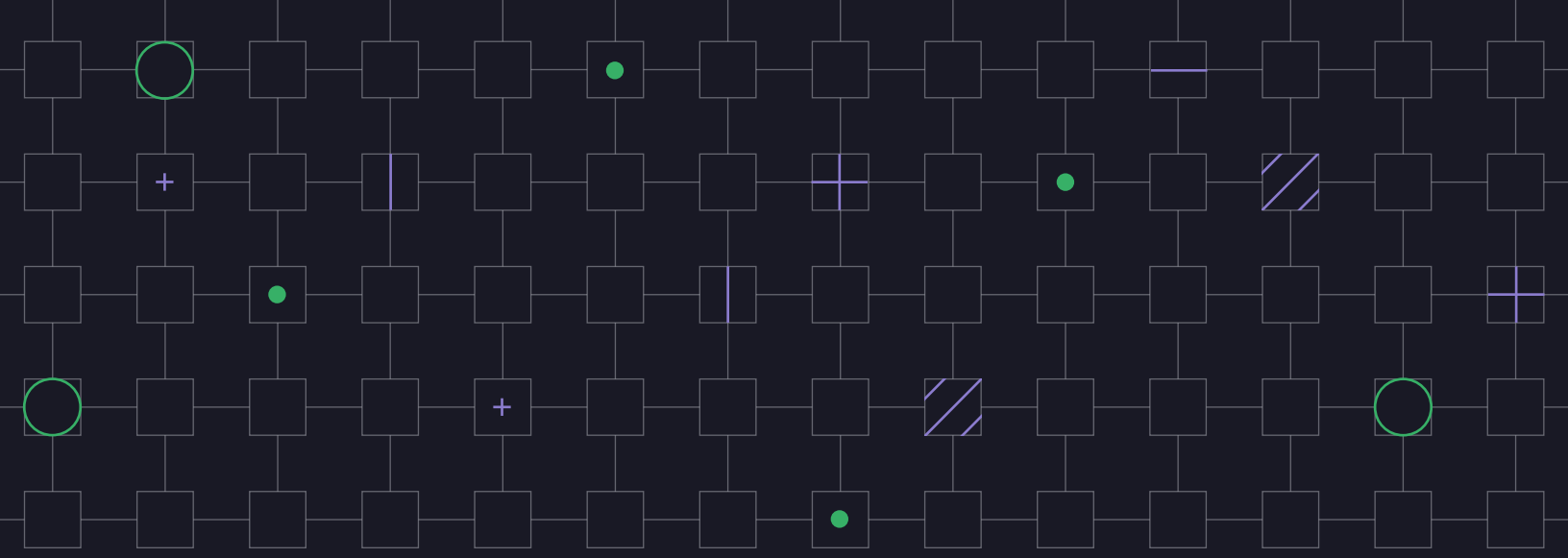
While these conversations push the capabilities of the stack forward, they often ignore more fundamental questions: "What should the modern data stack do?" and "How should it be used?"

We believe these questions are just as critical as those that define the tools and technologies in the modern data stack. After all, the point of these tools is to help everyone use data more effortlessly. Achieving that goal requires more than a tight architectural diagram—it requires us to think about the experience that the modern data stack can provide for us and our colleagues.

In this paper, we'll unpack how we think about building that experience. We'll start by looking at the architecture of the stack as it currently stands and discuss what experiential details this architecture leaves out. We'll then propose a list of foundational principles that we think can fill these gaps, and close with examples of teams that are leading the way in building the tools and technologies that can transform the modern data stack into the modern data experience.



To see Benn talk through this topic live, [check out this recording.](#)



An overview of the modern data stack



The legacy data architecture

The modern data stack is not the first data stack, nor is it the first generation of data tooling. To understand the modern data stack, it's helpful to take a look at the legacy systems that preceded it.

The popular data architectures of the past were defined by a few characteristics:

THEN	NOW
Expensive Storing and processing data used to be very expensive, so it cost a lot to build data pipelines, run data warehouses, and manage BI tools. This made data tooling inaccessible to companies that didn't have a lot of money to spend on it.	Affordable Compute is cheap and storage is nearly free, so enterprise-grade data storage and processing solutions are widely available.
On-prem Data is obviously a sensitive asset. Twenty years ago, it was stored in on-premise servers because there was no cloud to put it in; ten years ago, people were reluctant to put it in the cloud because they didn't trust it would be safe.	Cloud Though there are lingering holdouts, far fewer people object to the cloud. Many of today's data applications are fully managed by vendors, removing the need for IT teams to deploy software and run servers.
Monolithic Data tools tended to be all-in-one suites, packaging lots of different functionality—data ingestion, storage, transformation, visualization, etc.—into a single platform.	Modular Tools are now more modular and tend to focus on narrower functions.
Proprietary Tools were built on proprietary languages, like SAS and vendor-specific versions of SQL. Though a lot of tools still rely on their particular flavors of SQL or YAML-like configuration languages, the data industry is pushing towards more open languages and frameworks.	Open-Source Languages like Python and R, and open source frameworks like dbt and Airflow, are becoming the new standards.

Legacy systems were products of their time: They were built by huge enterprise vendors, dependent on IT support, and limited by the foundational technology they were built on. But this is becoming quickly outdated. As is true in many other industries, legacy products are being rebuilt in the cloud as SaaS applications. But this revolution is doing more than just moving old tools to new architectures; it's fundamentally changing how companies think about data.

The new philosophy

One of the biggest changes in the industry isn't just the technology, but the development of a philosophy on how to think about data and data tooling. These days, data is no longer used for just backwards-looking reporting; it's also now responsible for guiding strategic questions, predictive forecasting, real-time data applications, and everything in between.

To support these new uses of data, data professionals are aligning around a new set of foundational principles. Many of these principles [are drawn from software engineering](#), and they define what is now the modern data stack.



Code-based and version-controlled

Data can't be managed by a bunch of ever-changing, opaque spreadsheets. Controlling data with code, and version controlling that code, ensures data assets are stable.



Modular

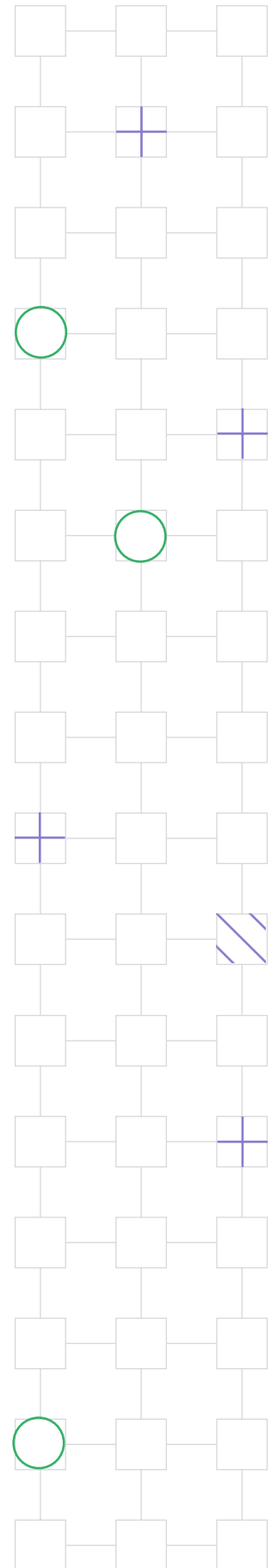
The data ecosystem is evolving quickly. New tools and technologies are exploding; within companies, new uses for data pop up all the time. In this context, companies don't have the time to deploy huge solutions that take months and years to roll out. They need modular tooling that can easily be swapped out when new opportunities emerge.



Collaborative

Every function now needs data. Everyone needs to be included in the analytical process—asking questions, iterating towards answers, making decisions—and tools need to be supportive of this.

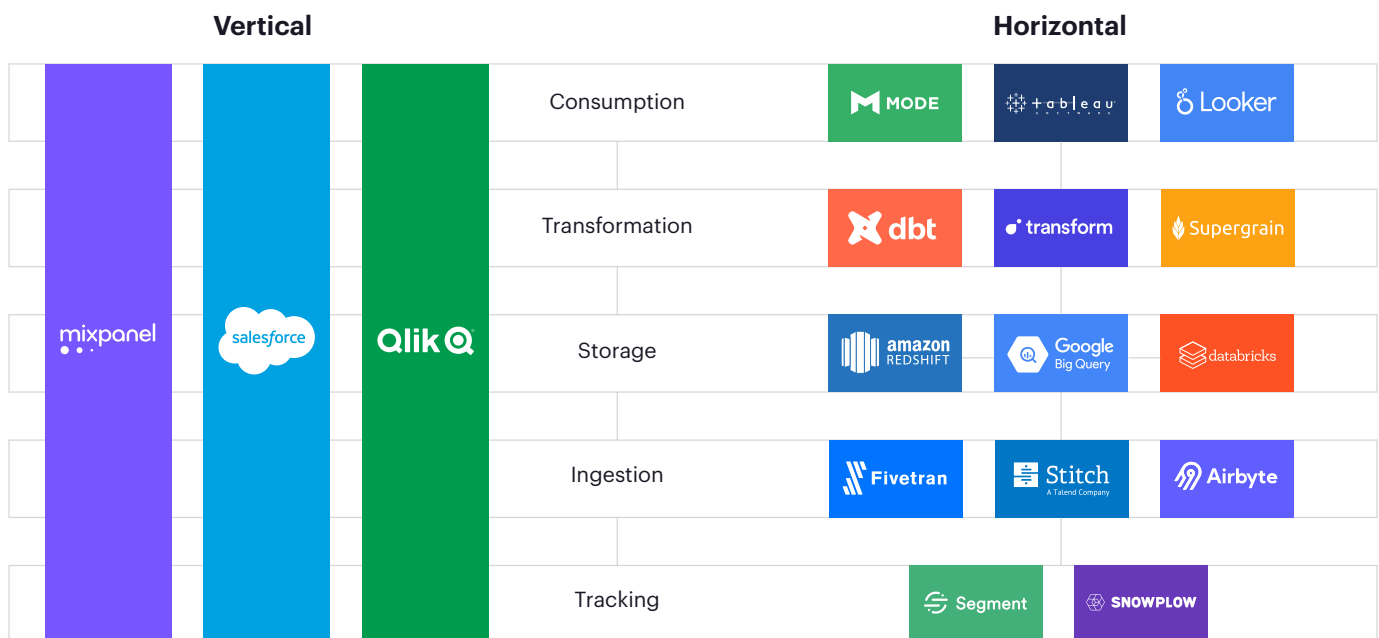
This philosophy informs how tools are built, and tools are built to enable this philosophy.



The new stack

New tools have emerged alongside this philosophy. Many of them follow the tenets of this new philosophy directly—they are code-based, they have collaborative elements baked in, and so on.

More broadly, these tools break apart the monolithic stack and [pivot it horizontally](#). Rather than data tools solving end-to-end problems or packaging several functions together into one platform, tools now target narrow layers of the data stack. Instead of one being responsible for data logging, data storage, and reporting, modern tools tend to focus on one job at a time. For example, as the diagram below shows, prior generations of vertically-integrated data tools are being unbundled and split into specialized horizontal tools.



The diagram above shows tools that strive to do everything in a vertical stack vs tools that specialize—or sit horizontally—in one part of the stack.

This reorientation has several significant benefits:

- Companies get best-in-class solutions at each layer of the stack.
- Companies aren't locked in to any one vendor because they can more easily swap out tools to meet their shifting needs.
- Companies can also adopt new technologies without replacing the rest of their stack.

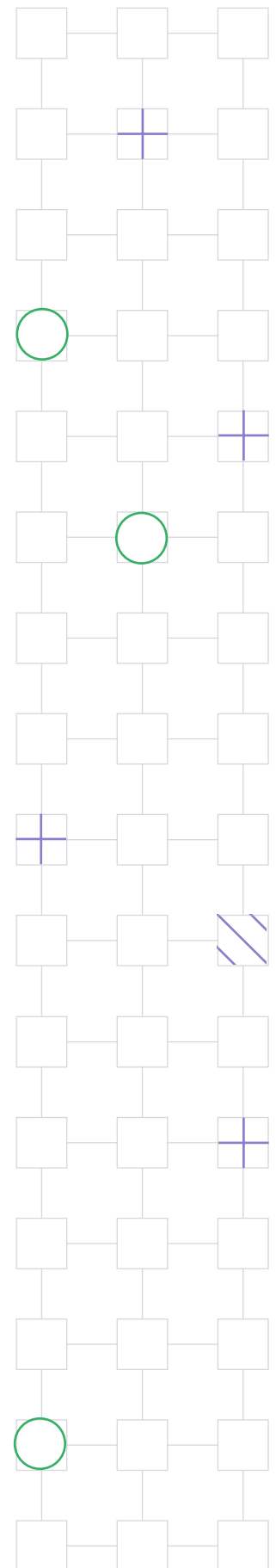
The new software

This new modern data stack is transforming how companies operate. Ten years ago, [Marc Andreessen famously said](#) that “software is eating the world.” The idea behind this phrase is that every company is becoming a software company. To be a bank, you have to build software that executes trades; to be an airline company, you have to build software to manage your fleet and schedules; to be a retailer, you have to build an ecommerce site.

The rise of SaaS products is changing this dynamic. Not everyone needs to be a software company because you can just buy the software you need. Stripe handles your financial transactions; Shopify is your ecommerce store.

“Tomorrow’s companies won’t all be software companies; they’ll all be data companies.”

Because these software tools are widely available, most businesses won’t differentiate themselves by the software they use or build; they’ll differentiate themselves by how they use the data that that software produces. And all of this software generates tons of data. Companies that use data to improve their products, to learn quickly from their customers, and to operate effectively, will succeed. Tomorrow’s companies won’t all be software companies; [they’ll all be data companies](#). And the modern data stack is emerging as the foundation that will make this possible.

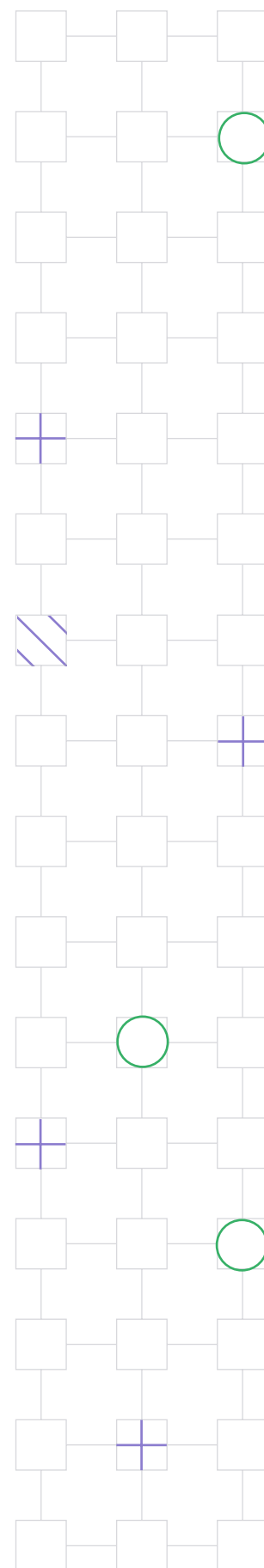


The problems with the stack

The modern data stack is full of a lot of new technology, and a lot of promise. But in many companies, its potential remains just that: potential. Many businesses are still struggling with foundational data challenges. Data teams still find themselves duplicating analyses, with analysts working in one tool and business users exploring data in another. People, even those who work on the same teams, struggle to know what others are working on. Executives and business stakeholders—the folks for whom the data stack is ultimately meant to serve—often struggle to get the data they need when they need it. As Ali Ghodsi, CEO of Databricks, recently put it in [his talk at Future Data 2021](#), most companies still spend the majority of their time figuring out what data is accurate and arguing over who has the right numbers.

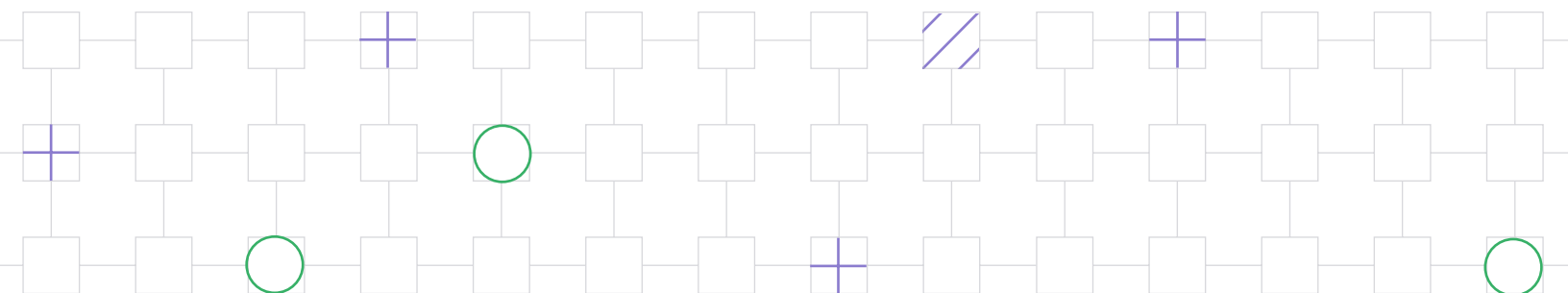
Given all the advances in data over the last decade, how have we not solved these problems yet? The answer is that, as the industry has grown, **we've built tools, and not experiences**. Too often, we deploy tools as discrete solutions that are, at best, loosely integrated with one another. In doing so, we've lost sight of how they're all supposed to work together.

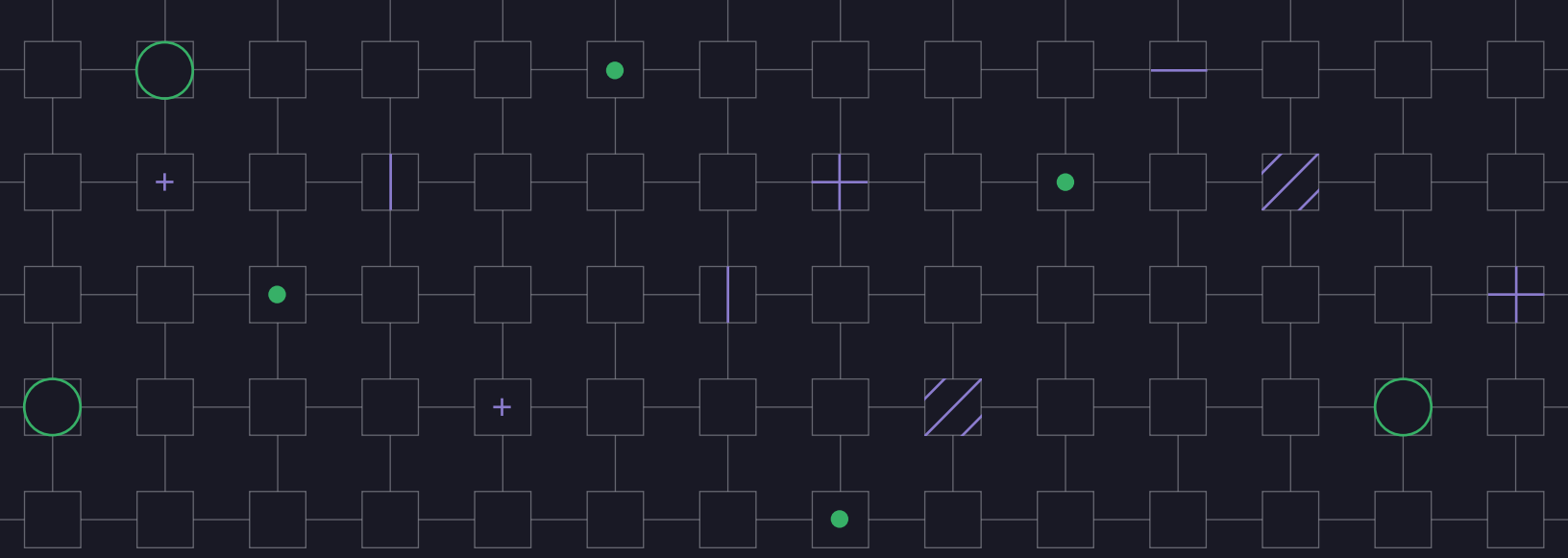
We want people to view dashboards, so we buy a dashboarding tool. We want people to have a place for answering strategic questions, so we buy an advanced analytics tool. But we think less about the experience of using these tools together. What, for example, should be the experience of going from a dashboard, to a new question, through a collaborative and iterative analytical process, to an answer? Our solutions make for great [architectural diagrams](#), but [blueprints aren't experiences](#).



To get the most out of the data stack, we need to stop thinking about it as a collection of components, and instead think of it as [an experience](#) for solving problems. Focusing on the experience will bridge the painful gaps between analytical tools, like the frustration many analysts feel when they need to bounce between a query editor, a Python notebook, and Excel just to answer a product manager's question. Focusing on the experience will also make it easier to make data truly universal. Most people's resistance to using data isn't about data, but about [how challenging we make it to use](#).

“To get the most out of the data stack, we need to stop thinking about it as a collection of components, and instead think of it as an experience for solving problems.”





Building a modern data experience

The founding principles

Just as we have principles that guide the modern data stack, we need similar principles for defining what we want to create in a modern data experience.

These principles are particularly important because this experience won't be built by a single vendor, but by an entire ecosystem. Though we'll never be perfectly aligned or agree to the same ideas, these principles can help keep us rowing in the same direction.

The 10 principles of the modern data experience

1

Enables everyone to do their job rather than asking them to be an analyst

In a modern data experience, we don't hand people data and ask them to analyze it; we incorporate it into the [operational systems](#) where they already live. Data should help people do their jobs, rather than add a new job for them to do.

2

Merges BI and data science

People should transition seamlessly between viewing a key metric sourced from a well-vetted data catalog, to exploring that metric with groupings and filters, to incorporating it in deep technical analyses. Those consuming data should never have to fully leave one system and start over in another.

3

Makes status and trust explicit

Every data asset should show if upstream processes are operating abnormally, out-of-date, or in some state of development or disrepair. Our goal should be to spend more time debating what to do because of a number on a dashboard than we spend verifying if that number's right.

4

Remembers what we've learned

To make sure our time is spent exploring new territory rather than retracing old steps, a modern data experience should remember and catalog what we learn and what we say about it.

5

Governs business logic globally

Business logic—instructions for transforming data and computing metrics—[should be centralized](#) such that it's accessible anywhere data is consumed, whether that's a BI dashboard, a Python notebook, or an operational ML pipeline.

6

Doesn't communicate in only tables

To data analysts, data is made up of tables and relational structures. To everyone else, data is ever-changing. Sometimes it's a time series of a single metric, sometimes it's a complex pivot table, and sometimes it's a document of explanatory narratives. People should be able to search for, ask questions of, and explore data in these terms, not just as tables and columns.

7

Builds bridges between the past and future

The modern data stack is not a leap: It's a transition and some uncomfortable anchors are coming with us. Most notably, Excel isn't going away. A modern data experience has to negotiate with it and not treat it as an outdated pariah.

8

Goes backwards and forwards

We're borrowing this one [from Tristan Handy](#) at dbt. Analytical work requires a lot of revisions, and trial and error. Tools should support this iterative process, allowing people to easily go from point A to point B and back again.

9

Is elastic

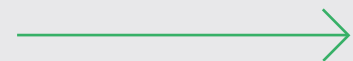
The modern data experience should be emergent—able to start small and grow into new, unforeseen territory. Rigid experiences and systems are debt that will quickly come due.

10

Is a melting pot

Technical modularity can't tempt us to build more walls. A modern data experience needs to break down walls by [encouraging collaboration and conversation](#) between business, data, and engineering teams.

To get a better understanding of how we can apply these principles, let's look at a few data systems that already embrace them.



Gathering inspiration from Amazon, Airbnb, and Yelp

Though the modern data experience is still being defined, there are a few existing models that can serve as inspiration for those of us who want to create it. The first examples below show how to build an experience from a collection of technologies; the second demonstrate how to apply these principles to data tools; and the third highlight how effective data experiences can be at making everyone more data-driven.

Cloud computing services - designing an integrated foundation

Major cloud providers like Amazon, Google, and Microsoft all offer a huge list of interconnected services to their customers. These services are technically independent, in that they're separate products that don't need to be used with one another. However, within each cloud, the services share a central nervous system that makes it easy for them to interoperate with one another. For instance, if you want to move data from S3 to Redshift (both of which are AWS services), AWS offers easy ways to do this. If you want to move data from S3 to BigQuery, Google's primary cloud warehouse, it's more difficult.

A modern data experience should operate like these clouds: Tightly integrated tools that can be used independently, but work best when integrated with one another.

What's not yet clear, however, is what the data stack's shared backbone is. Currently, it's the database: Most tools in the stack are only aware of each other via the tables they create and update (e.g., Tableau doesn't know anything about Fivetran; it just sees the tables Fivetran updates).

Existing models for modern data experience



We need a better system for tools to communicate. Tools like dbt or Databricks' Lakehouse could begin to serve as an ["OS" for the data stack](#). Until something like this exists, however, the modern data experience will only be as good as the bilateral integrations between individual services. Given the number of tools in the stack, these won't suffice for long.

Internal data stacks - how tools can interoperate

Airbnb, Uber, and Netflix have all built end-to-end integrated data stacks that include everything from reporting tools, metrics repositories, data catalogs, and ML platforms. While cloud computing services provide an architectural guide for the modern data experience, these internal tools provide an experiential guide for what it's like to work with interconnected data tools.

Uber

At Uber, people can search for metrics in a metric catalog, explore it across different dimensions, and open their explorations up in a code-first interface if they want to explore it further. Moreover, they have an AI that dedupes metrics and helps people find concepts that might be overlapping or duplicative with what they're looking for.

<https://eng.uber.com/databook/>

<https://eng.uber.com/umetric/>



Airbnb has a similar platform to that of Uber's, complete with a metrics store, query tools, visualization and dashboard tools, an A/B testing platform to that of Uber's, and a knowledge repository for data scientists to record what they've learned.

<https://medium.com/airbnb-engineering/supercharging-apache-superset-b1a2393278bd>

<https://medium.com/airbnb-engineering/how-airbnb-achieved-metric-consistency-at-scale-f23cc53dea70>

<https://medium.com/airbnb-engineering/democratizing-data-at-airbnb-852d76c51770>

<https://medium.com/airbnb-engineering/scaling-knowledge-at-airbnb-875d73eff091>

NETFLIX

Netflix has designed an entire workflow for creating, sharing, deploying, scheduling, and discovering notebook applications that support everything from production dashboards to production ML models.

<https://netflixtechblog.com/notebook-innovation-591ee3221233>

<https://netflixtechblog.com/exploring-data-netflix-9d87e20072e3>

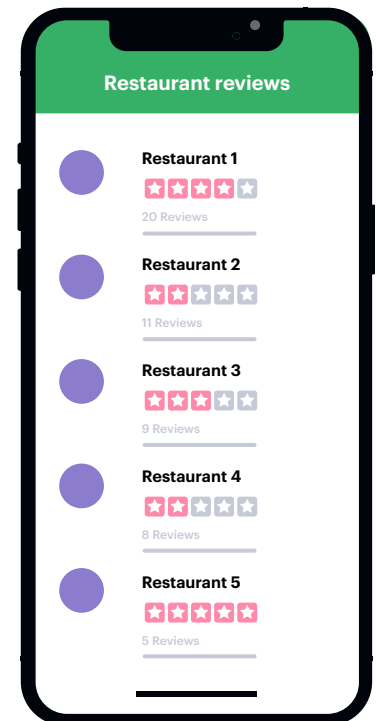
Yelp and consumer apps - making data inescapable

What does the modern data experience feel like to non-analysts and decision-makers? The best examples actually come from products that we don't even think of as data tools.

Think about the last time you needed to choose a place to eat. You probably looked up what was popular on Yelp, scanned the reviews listed on different pages, and found a place that had both a high average score and a respectable number of reviews. This process has become such a habit—and data is so seamlessly integrated within the experience—that we barely even notice how much we rely on data to make this decision.

We do the same thing in other choices we make as consumers. We check Kayak for ticket prices and trends, and look up review data when we're shopping on Amazon. We've never been formally trained to use this data, and nobody is encouraging us to be more data-driven in how we shop, but we are all making data-driven decisions because these companies integrate exactly the data we need right at the moment we need it.

The same experiences are possible within our own companies. People often need to make decisions in the flow of their actual jobs. Support agents need to decide how to best speak to a customer who's writing in; designers need to know engagement within the product as they design it. In these cases, we should provide data directly to the support agent and the designer in the app in which they're working.

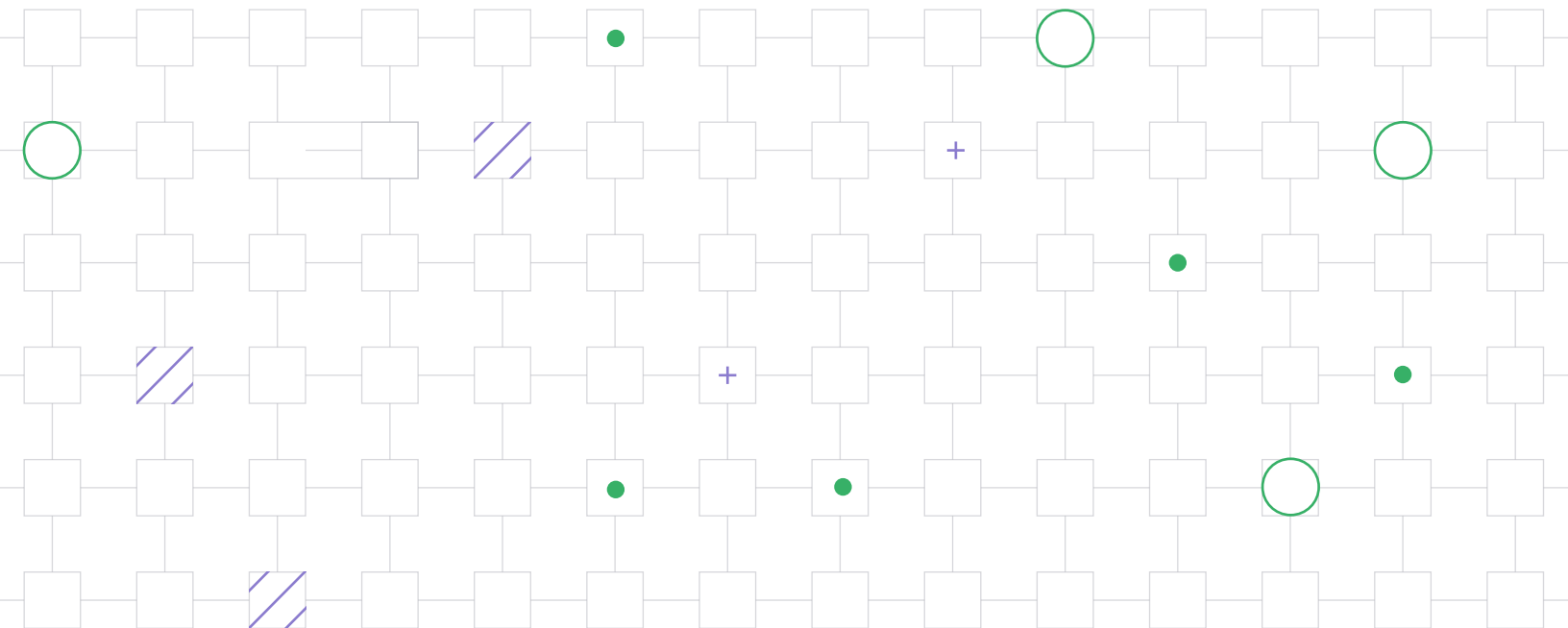


Creating a global exploration experience

Of course, these experiences can't cover everything. Data isn't always needed for an operational decision; there are many cases when we need to make strategic decisions or ask open-ended questions. In these cases, we need a global exploration experience.

Today, we fragment most data exploration along technical lines: We have a tool for dashboards, for visual analysis, for SQL, a fourth for Python and R, and so on. This artificially [breaks apart how people use data](#), and segregates people into different buckets (analysts work in one set of tools; non-analysts work in another). But these divisions are artificial. Data is actually explored along a spectrum. We might want to include analysis from a Python notebook in a dashboard, or open visual analytics tools up in a technical IDE.

The experience of exploring data is better when people can [hop between different modes of consumption](#). Moreover, by bringing these modes of working together, we bring people together, making analysis the collaborative, company-wide process it should be.



What do you do now?

As an industry, we need to start having conversations about the principles that define the modern data experience. As vendors, we need to start aligning those principles. But what do we do as data teams that can't build products or change a community? How do we build a modern data experience with the products and cultures we have?

I have two pieces of advice:

Understand your customer and problem

Building a great experience requires, first and foremost, understanding what problem you're solving and who you're solving it for. Right now, data teams get asked a ton of questions by a wide range of people, which they—understandably—collapse into generic problems like “business users need to explore data.” Nobody's ever built a great experience on top of such a vague problem statement. Teams need to [better understand who their customers](#) are and what they want to do. Dashboards are a good example of teams failing to do this. A lot of dashboards get built because people want data, but the request is imprecise. Rather than being a solution, dashboards are often data searching for a problem. Imagine if Yelp operated this way: People want to find a good restaurant, and rather than helping them do that directly, Yelp builds a dashboard full of different charts of restaurant reviews. Of course it'd never work, but it's what we do as data teams every day.

Be ruthlessly disciplined

There will always be more questions than answers, and teams can always say that they can't build great experiences because they have too many needs to serve. I'd challenge this idea. People don't often need everything they ask for. Plenty of decisions can be made with data that's “good enough.” Again, consider Yelp: We could say we need more data to decide which restaurant is best; a rounded star rating isn't enough. But, in practice, it is enough. We actually can make good decisions with it; more data would probably paralyze us rather than make us smarter. Data teams should be comfortable with the same kind of simplified solutions. Push people to the metrics they need, and integrate those metrics into the places they make decisions as much as possible. Making decisions with limited data, presented well and in the right moment, is often a much better experience than making decisions with lots of data.



Conclusion

The modern data stack is, to use a word that's often overused by the technology industry, transformative. It's shaping a new philosophy for how companies build data tools and data teams. It's creating a host of new technologies that make collecting, storing, modeling, and sharing data faster, easier, and more reliable than ever before. And it's building a new foundation that will underpin the next generation of defining companies.

But it could do more. One thing it has yet to transform is the everyday experience of using data. For most people, data is still an out-of-date dashboard, a mismatched set of numbers, and a bunch of Excel files floating around shared corporate folders. Until these experiences are as good as the technology underneath them, the modern data stack will remain incomplete.

We must focus less on the tools we're building for the modern data stack, and more on the experiences we want to create. And in order to make this shift, we need to be disciplined, deeply understanding of our customers' problems, and guided by shared principles. Without thinking about the experience that the stack creates, a technological revolution—a shift in how industries function—won't come.

The modern data experience is the next frontier.

**Make
your data
experience
better with
Mode**

[Request a demo](#)

[Try for free](#)

