McKinsey Digital

How to build a data architecture to drive innovation—today and tomorrow

June 3, 2020 | Article

By Antonio Castro, Jorge Machado, Matthias Roggendorf, and Henning Soller

Yesterday's data architecture can't meet today's need for speed, flexibility, and innovation. The key to a successful upgrade—and significant potential rewards—is agility.

DOWNLOADS

Article (PDF-276 KB)

Over the past several years, organizations have had to move quickly to deploy new data technologies alongside legacy infrastructure to drive market-driven innovations such as personalized offers, real-time alerts, and predictive maintenance.

However, these technical additions—from data lakes to customer analytics platforms to stream processing—have increased the complexity of data architectures enormously, often significantly hampering an organization's ongoing ability to deliver new capabilities, maintain existing infrastructures, and ensure the integrity of artificial intelligence (AI) models.

Current market dynamics don't allow for such slowdowns. Leaders such as Amazon and Google have been making use of technological innovations in AI to upend traditional business models, requiring laggards to reimagine aspects of their own business to keep up. Cloud providers have launched cutting-edge offerings, such as serverless data platforms that can be deployed instantly, enabling adopters to enjoy a faster time to market and greater agility. Analytics users are demanding more seamless tools, such as automated model-deployment platforms, so they can more quickly make use of new models. Many organizations have adopted application programming interfaces (APIs) to expose data from disparate systems to their data lakes and rapidly integrate insights directly into front-end applications. Now, as companies navigate the unprecedented humanitarian crisis caused by the COVID-19 pandemic and prepare for the next normal, the need for flexibility and speed has only amplified.

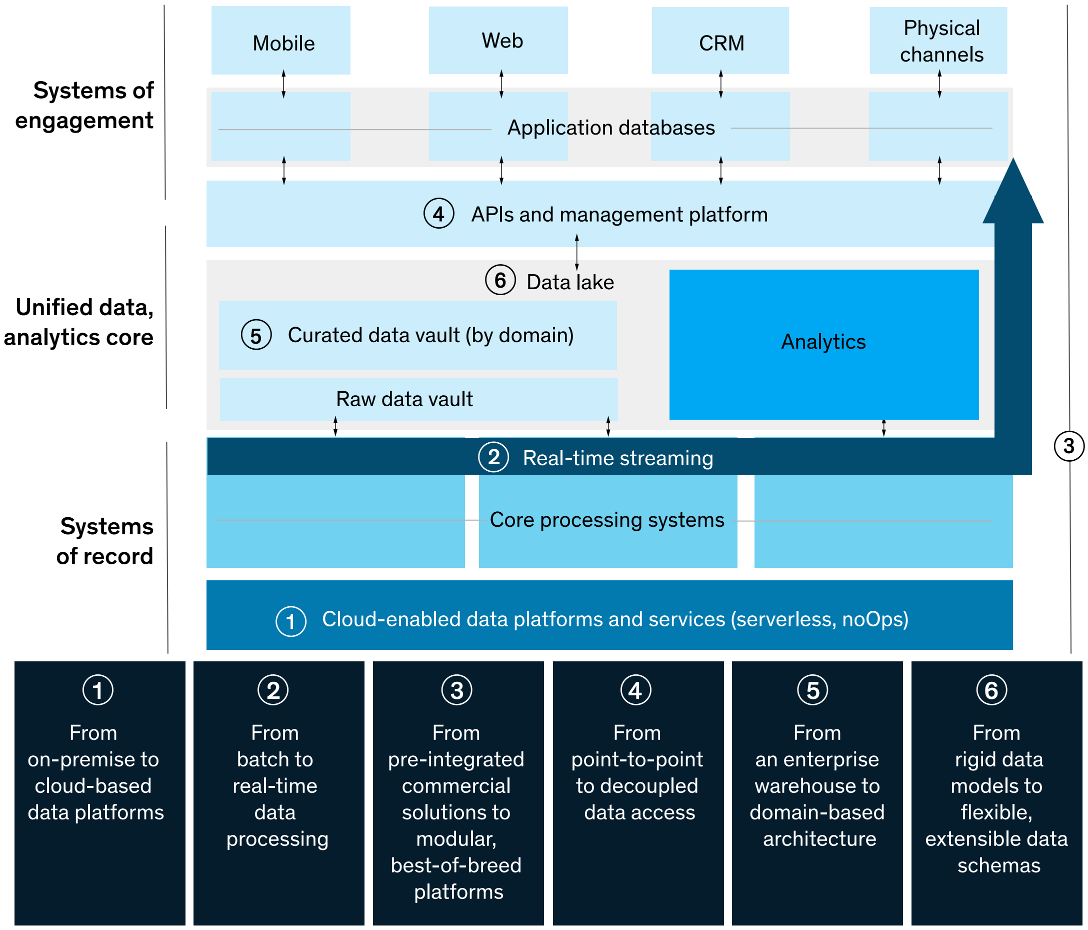
For companies to build a competitive edge—or even to maintain parity, they will need a new approach to defining, implementing, and integrating their data stacks, leveraging both cloud (beyond infrastructure as a service) and new concepts and components.

Six shifts to create a game-changing data architecture

We have observed six foundational shifts companies are making to their data-architecture blueprints that enable more rapid delivery of new capabilities and vastly simplify existing architectural approaches (exhibit). They touch nearly all data activities, including acquisition, processing, storage, analysis, and exposure. Even though organizations can implement some shifts while leaving their core technology stack intact, many require careful re-architecting of the existing data platform and infrastructure, including both legacy technologies and newer technologies previously bolted on.

Exhibit

Upgrade data architecture by making six foundational shifts.



1. From on-premise to cloud-based data platforms
2. From batch to real-time data processing
3. From pre-integrated commercial solutions to modular, best-of-breed platforms
4. From point-to-point to decoupled data access
5. From an enterprise warehouse to domain-based architecture
6. From rigid data models to flexible, extensible data schemas

McKinsey & Company

Such efforts are not insignificant. Investments can often range in the tens of millions of dollars to build capabilities for basic use cases, such as automated reporting, to hundreds of millions of dollars for putting in place the architectural components for bleeding-edge capabilities, such as real-time services in order to compete with the most innovative disruptors. Therefore, it is critical for organizations to have a clear strategic plan, and data and technology leaders will need to make bold choices to prioritize those shifts that will most directly impact business goals and to invest in the right level of architecture sophistication. As a result, data-architecture blueprints often look very different from one company to another.

When done right, the return on investment can be significant (more than $500 million annually in the case of one US bank, and 12 to 15 percent profit-margin growth in the case of one oil and gas company). We find these types of benefits can come from any number of areas: IT cost savings, productivity improvements, reduced regulatory and operational risk, and the delivery of wholly new capabilities, services, and even entire businesses.

So what key changes do organizations need to consider?

1. From on-premise to cloud-based data platforms

Cloud is probably the most disruptive driver of a radically new data-architecture approach, as it offers companies a way to rapidly scale AI tools and capabilities for competitive advantage. Major global cloud providers such as Amazon (with Amazon Web Services), Google (with the Google Cloud Platform), and Microsoft (with Microsoft Azure) have revolutionized the way organizations of all sizes source, deploy, and run data infrastructure, platforms, and applications at scale.

One utility-services company, for example, combined a cloud-based data platform with container technology, which holds microservices such as searching billing data or adding new properties to the account, to modularize application capabilities. This enabled the company to deploy new self-service capabilities to approximately 100,000 business customers in days rather than months, deliver large amounts of real-time inventory and transaction data to end users for analytics, and reduce costs by “buffering” transactions in the cloud rather than on more expensive on-premise legacy systems.

Enabling concepts and components

- *Serverless data platforms,* such as Amazon S3 and Google BigQuery, allow organizations to build and operate data-centric applications with infinite scale without the hassle of installing and configuring solutions or managing workloads. Such offerings can lower the expertise required, speed deployment from several weeks to as little as a few minutes, and require virtually no operational overhead.
- *Containerized data solutions* using Kubernetes (which are available via cloud providers as well as open source and can be integrated and deployed quickly) enable companies to decouple and automate deployment of additional compute power and data-storage systems. This capability is particularly valuable in ensuring that data platforms with more complicated setups, such as those required to retain data from one application session to another and those with intricate backup and recovery requirements, can scale to meet demand.

2. From batch to real-time data processing

The costs of real-time data messaging and streaming capabilities have decreased significantly, paving the way for mainstream use. These technologies enable a host of new business applications: transportation companies, for instance, can inform customers as their taxi approaches with accurate-to-the-second arrival predictions; insurance companies can analyze real-time behavioral data from smart devices to individualize rates; and manufacturers can predict infrastructure issues based on real-time sensor data.

Real-time streaming functions, such as a subscription mechanism, allow data consumers, including data marts and data-driven employees, to subscribe to “topics” so they can obtain a constant feed of the transactions they need. A common data lake typically serves as the “brain” for such services, retaining all granular transactions.

Enabling concepts and components

- *Messaging platforms* such as Apache Kafka provide fully scalable, durable, and fault-tolerant publish/subscribe services that can process and store millions of messages every second for immediate or later consumption. This allows for support of real-time use cases, bypassing existing batch-based solutions, and a much lighter footprint (and cost base) than traditional enterprise messaging queues.
- *Streaming processing and analytics solutions* such as Apache Kafka Streaming, Apache Flume, Apache Storm, and Apache Spark Streaming allow for direct analysis of messages in real time. This analysis can be rule based or involve advanced analytics to extract events or signals from the data. Often, analysis integrates historic data to compare patterns, which is especially vital in recommendation and prediction engines.
- *Alerting platforms* such as Graphite or Splunk can trigger business actions to users, such as notifying sales representatives if they’re not meeting their daily sales targets, or integrate these actions into existing processes that may run in enterprise resource planning (ERP) or customer relationship management (CRM) systems.

3. From pre-integrated commercial solutions to modular, best-of-breed platforms

To scale applications, companies often need to push well beyond the boundaries of legacy data ecosystems from large solution vendors. Many are now moving toward a highly modular data architecture that uses best-of-breed and, frequently, open-source components that can be replaced with new technologies as needed without affecting other parts of the data architecture.

The utility-services company mentioned earlier is transitioning to this approach to rapidly deliver new, data-heavy digital services to millions of customers and to connect cloud-based applications at scale. For example, it offers accurate daily views on customer energy consumption and real-time analytics insights comparing individual consumption with peer groups. The company set up an independent data layer that includes both commercial databases and open-source components. Data is synced with back-end systems via a proprietary enterprise service bus, and microservices hosted in containers run business logic on the data.

Enabling concepts and components

- *Data pipeline and API-based interfaces* simplify integration between disparate tools and platforms by shielding data teams from the complexity of the different layers, speeding time to market, and reducing the chance of causing new problems in existing applications. These interfaces also allow for easier replacement of individual components as requirements change.
- *Analytics workbenches* such as Amazon Sagemaker and Kubeflow simplify building end-to-end solutions in a highly modular architecture. Such tools can connect with a large variety of underlying databases and services and allow highly modular design.

4. From point-to-point to decoupled data access

Exposing data via APIs can ensure that direct access to view and modify data is limited and secure, while simultaneously offering faster, up-to-date access to common data sets. This allows data to be easily reused among teams, accelerating access and enabling seamless collaboration among analytics teams so AI use cases can be developed more efficiently.

One pharmaceutical company, for example, is setting up an internal "data marketplace" for all employees via APIs to simplify and standardize access to core data assets rather than relying on proprietary interfaces. The company is gradually—over 18 months—migrating its most valuable existing data feeds to an API-based structure and deploying an API management platform to expose the APIs to users.

Enabling concepts and components

- *An API management platform* (often called an API gateway) is necessary to create and publish data-centric APIs, implement usage policies, control access, and measure usage and performance. This platform also allows developers and users to search for existing data interfaces and reuse them rather than build new ones. An API gateway is often embedded as a separate zone within a data hub but can also be developed as a standalone capability outside of the hub.
- *A data platform to "buffer" transactions outside of core systems* is often required. Such buffers could be provided by central data platforms such as a data lake or in a distributed data mesh, which is an ecosystem consisting of best-fit platforms (including data lakes, data warehouses, and so on) created for each business domain's expected data usage and workloads. For example, one bank built a

columnar database to provide customer information, such as their most recent financial transactions, directly to online and mobile banking applications and reduce costly workloads on its mainframe.

5. From an enterprise warehouse to domain-based architecture

Many data-architecture leaders have pivoted from a central enterprise data lake toward “domain-driven” designs that can be customized and “fit for purpose” to improve time to market of new data products and services. With this approach, while the data sets may still reside on the same physical platform, “product owners” in each business domain (for example, marketing, sales, manufacturing, and so on) are tasked with organizing their data sets in an easily consumable way both for users within their domain and for downstream data consumers in other business domains. This approach requires a careful balance to avoid becoming fragmented and inefficient, but in return it can reduce the time spent up front on building new data models into the lake, often from months to just days, and can be a simpler and more effective choice when mirroring a federated business structure or adhering to regulatory limitations on data mobility.

One European telecommunications provider used a distributed domain-based architecture so sales and operations staff could expose customer, order, and billing data to data scientists for use in AI models or directly to customers via digital channels. Rather than building one central data platform, the organization deployed logical platforms that are managed by product owners within the company’s sales and operations teams. Product owners are incentivized to promote the use of the data for analytics and are using digital channels as well as forums and hackathons to drive adoption.

Enabling concepts and components

- *Data infrastructure as a platform* provides common tools and capabilities for storage and management to speed implementation and remove from data producers the burden of building their own data-asset platform.
- *Data virtualization techniques*, which started in niche areas such as customer data, are now being used across enterprises to organize access to and integrate distributed data assets.
- *Data cataloging tools* provide enterprise search and exploration of data without requiring full access or preparation. The catalog also typically provides metadata definitions and an end-to-end interface to simplify access to data assets.

6. From rigid data models toward flexible, extensible data schemas

Predefined data models from software vendors and proprietary data models that serve specific business-intelligence needs are often built in highly normalized schemas with rigid database tables and data elements to minimize redundancy. While this approach remains the standard for reporting and regulatory-focused use cases, it also requires that organizations undergo lengthy development cycles and have strong system knowledge when they want to incorporate new data elements or data sources, as any changes can affect data integrity.

To gain greater flexibility and a powerful competitive edge when exploring data or supporting advanced analytics, companies are evolving to “schema-light” approaches, using denormalized data models, which have fewer physical tables, to organize data for maximum performance. This approach offers a host of benefits: agile data exploration, greater flexibility in storing structured and unstructured data, and reduced complexity, as data leaders no longer need to introduce additional abstraction layers, such as multiple "joins" between highly normalized tables, to query relational data.

Enabling concepts and components

- *Data vault 2.0 techniques*, such as data-point modeling, can ensure that data models are extensible so data elements can be added or removed in the future with limited disruption.
- *Graph databases*, a type of NoSQL database, have gained attention in recent years. NoSQL databases in general are ideal for digital applications that require massive scalability and real-time capabilities, and for data layers serving AI applications, thanks to their ability to tap into unstructured data. Graph databases, in particular, offer the ability to model relationships within data in a powerful and flexible manner, and many companies are building master data repositories using graph databases to accommodate changing information models.
- *Technology services* such as Azure Synapse Analytics allow querying file-based data akin to relational databases by dynamically applying table structures onto the files. This provides users the flexibility to continue using common interfaces such as SQL while accessing data stored in files.
- *Using JavaScript Object Notation (JSON)* to store information enables organizations to change database structures without having to change business information models.

How to get started

Data technologies are evolving quickly, making traditional efforts that define and build toward three-to-five-year target architectural states both risky and inefficient. Data and technology leaders will be best served by instituting practices that enable them to rapidly evaluate and deploy new technologies so they can quickly adapt. Four practices are crucial here:

- *Apply a test-and-learn mindset* to architecture construction, and experiment with different components and concepts. Such agile practices have been applied in application development for quite a while and have recently moved into the data space. For example, rather than engage in drawn-out discussions about optimal designs, products, and vendors to identify the "perfect" choice followed by lengthy budget approvals, leaders can start with smaller budgets and create minimum viable products or string together existing open-source tools to create an interim product, releasing them into production (using cloud to accelerate) so they can demonstrate their value before expanding and evolving further.

- *Establish data "tribes,"* where squads of data stewards, data engineers, and data modelers work together with end-to-end accountability for building the data architecture. These tribes also work to put in place standard, repeatable data- and feature-engineering processes to support development of highly curated data sets ready for modeling. These agile data practices can help accelerate time to market of new data services.

- *Invest in DataOps* (enhanced DevOps for data), which can help to accelerate the design, development, and deployment of new components into the data architecture so teams can rapidly implement and frequently update solutions based on feedback.

- *Create a data culture* where employees are eager to use and apply new data services within their roles. One essential tool to achieve this is ensuring that data strategy ties to the business goals and is reflected in the C-suite's messages to the organization, which can help reinforce the importance of this work to business teams.

As data, analytics, and AI become more embedded in the day-to-day operations at most organizations, it's clear that a radically different approach to data architecture is necessary to create and grow the data-centric enterprise. Those data and technology leaders who embrace this new approach will better position their companies to be agile, resilient, and competitive for whatever lies ahead.

How relevant and useful is this article for you?

☆☆☆☆☆

ABOUT THE AUTHOR(S)

Antonio Castro and **Jorge Machado** are partners in McKinsey's New York office, **Matthias Roggendorf** is a partner in the Berlin office, and **Henning Soller** is a partner in the Frankfurt office.

The authors wish to thank Josh Gottlieb, Sameer Kohli, Aziz Shaikh, and Nikhil Srinidhi for their contributions to this article.

EXPLORE A CAREER WITH US

Search Openings