



# How the right databases help Zulily engage customers

*A dbInsight case study for Amazon Web Services*

---

## Executive Summary

### Trigger

The changes to the global economy that are driving digital business are also driving demand for a new generation of scalable, cloud-ready, modern applications. Compared to traditional, internally-facing enterprise transaction applications, modern applications respond to a wide range of scenarios and data types that in turn drive varying requirements for performance, scale, and geographic reach. There is no single, silver bullet data management recipe for responding to use cases ranging from supporting millions of concurrent sessions for online gaming, to identifying patterns of fraud in financial transactions, and delivering consistent high performance for scenarios with unpredictable demand spikes. With a variety of options, ranging from traditional SQL relational databases to NoSQL key-value stores, document database, graph databases, and others, how can organizations make the right choices?

### Our Take

The use case drives the choice of database. While there is no single formula dictating whether an enterprise adopts one or more databases to address their business use case or scenario, they should depart from preset notions and dissect the problem into its constituent parts, assessing the need for requirements ranging from ACID transaction consistency, referential integrity, latency, availability, and other factors. In some instances, a mix of database systems may prove the best solution, for example with web and mobile commerce applications that may require a mix of simple lookups, search capabilities, and in some cases, leaderboards for showing hot selling items. In other cases, a single solution may address the challenge where the problem is narrow and well-defined, such as sorting through the complex interrelationships of entertainment content.

The first step in choosing the data solution always starts with the business goal and then evaluating the tasks that are necessary to address it. In many cases, the tasks and the requirements for data may be varied. For example, in e-commerce, where the operations often involve a mix of stable and volatile data, there is need for simple lookups and search, and the ability to quickly sort through and retrieve frequently accessed data. With numerous options for databases available, enterprises need not make compromises on choosing the best data solution for the job.

## How Zulily keeps customers returning with targeted searches

### Turning shopping into an event-based experience

Zulily, an online retailer selling merchandise to mothers and their children, draws customers by making shopping an event-based experience. As a members-only site, Zulily promises a personalized shopping experience, where each day brings new assortments of women's fashion, children's items, and home décor to browse. Zulily offers a "best price" promise to its members, launching over 100 events a day and featuring 9,000 styles that are live on the site for typically 72 hours.

### Behind the scenes: AWS databases play pivotal role

The online retailer has placed the bulk of its online infrastructure on AWS and uses multiple database services, including Amazon DocumentDB (with MongoDB compatibility) and Amazon Aurora MySQL; it also runs its own implementation of MongoDB on Amazon EC2 and uses a third-party data warehouse. Because the site is event-driven and refreshed every morning at 6am Pacific Time, Zulily relies on Amazon Kinesis Data Analytics to filter search events from clickstream data in Kinesis Data Streams. Kinesis Data Analytics helps Zulily easily write SQL-like statements to filter search events across huge data sets at scale. As customers log in to search or discover what's new, Zulily captures search keywords submitted by customers, looks up related brands and product categories, performs an inventory check and offers popular search keywords, brands, and product categories as search suggestions. Since it was recently introduced, Zulily's suggested searches feature has proven a big hit, with over 75% of Zulily's customers who use the search feature leveraging search suggestions as they shop.

The search suggestions provide a rich experience showing trending searches based on relevant keywords, brands, and categories. The suggested searches make liberal use of images and graphics that inspire customers. This new feature replaced a more monolithic search feature that was primarily focused on trending searches.

As is common practice in e-commerce today, Zulily performs extensive user research offline to understand the member's preferences, and that is the starting point for shaping the suggested search. The differentiation of suggested searches is based on how Zulily makes those preferences actionable.

### Choosing the right database matters

Behind the scenes, the choice of database and streaming analytics allowed Zulily to build this heavily-used feature faster and thus reduce the time to market. The team chose DocumentDB for storing search events, and Kinesis Data Analytics for smart filtering real-time clickstreams to extract search events that populate the database.

Here's how suggested search works. Any action that Zulily members take on the website or mobile app is recorded as clickstream events in Amazon Kinesis Data Streams. When a Zulily member requests a search, Kinesis Data Analytics filters relevant events from clickstream analytics. The microservice then feeds the search event to another Kinesis stream, which in turn triggers a Lambda function performing a lookup for relevant brands and categories in real time. The results are stored as enriched events in DocumentDB. These results populate the suggested search, which is composed of keywords, relevant brands, and categories. Once verified, the search results are populated back to the user in a rich display that, as noted above, carries additional relevant suggestions to provide shopping inspiration to customers. DocumentDB hosts the enriched search events and inventory checks are performed based on data stored in DocumentDB. The final results, after inventory check, are stored in Amazon ElastiCache for Redis for fast persistence and are accessible to customers via a REST API.

Both DocumentDB and Kinesis were chosen for several reasons. DocumentDB was selected, first, because as a MongoDB-compatible database, it was familiar to developers on the team and would not require much of a learning curve. Secondly, as a highly scalable AWS-managed service, it would allow developers to focus on developing and iterating the application rather than having to shoulder the burden of deploying, operating, and maintaining the database. Kinesis Analytics was chosen to simplify the process of filtering out search events from the huge data set containing hundreds of millions of user actions and pageviews.

It was all part of Zulily's strategy to decouple development of new features to make the site, in essence, future proofed. New features would be developed through APIs to avoid the need for constant re-architecting of the underlying infrastructure. More importantly, because the team utilized managed AWS services, rather than having to handle all deployment, maintenance, and operation of the underlying data infrastructure itself, Zulily was able to get the suggested search feature into production in 10 weeks, which was a fraction of the time that would have been required if they went the conventional route.

Having gone live in August 2020, the new DocumentDB implementation of the customer experience database has proven its ability to keep the experience fresh during a period where online commerce has exploded. As proof of its popularity, the search suggestion feature is used by 75% of Zulily's customers using search as they shop.

## Takeaways

Depending on the use case, there may be widely diverging requirements for managing and processing data. The data in a product lookup table that is stable will have different requirements from the data representing public health exposure from interpersonal contacts that are constantly changing. But the decision is not purely a matter of architecture. The same body of data that might have a relational structure may require different modes of access depending on whether the data is stable, such as a fact table, or volatile, such as tracking the latest product, gaming, or media content choices by the consumer. Can the requirements for reading and writing data be broken up into separate steps, such as member lookup and next-best action or recommendation, and therefore be addressed by database solutions that are designed for the purpose? Furthermore, does the nature of the use case translate to a need for extreme autoscaling and/or the need to read and/or write data with processes that are localized or highly distributed? And what is more important to the business case: returning data that is absolutely consistent, or ensuring that the service be highly available?

When choosing the database(s), look at the business use case first. With a multitude of database options out there, the choice does not have to be whittled down to an either-or.

## Author

Tony Baer, Principal, dbInsight

[tony@dbinsight.io](mailto:tony@dbinsight.io)

Twitter @TonyBaer

## About dbInsight

dbInsight LLC® provides an independent view on the database and analytics technology ecosystem. dbInsight publishes independent research, and from our research, distills insights to help data and analytics technology providers understand their competitive positioning and sharpen their message.

Tony Baer, the founder and principal of dbInsight, is a recognized industry expert on data-driven transformation. *Analytica* named him as one of its influencers for [data, data management](#), and [cloud](#) in 2019, 2020, and 2021. *Analytics Insight* named him one of the [2019 Top 100 Artificial Intelligence and Big Data Influencers](#). His combined expertise in both legacy database technologies and emerging cloud and analytics technologies shapes how technology providers go to market in an industry undergoing significant transformation. His regular ZDnet “*Big on Data*” posts are read 25,000 – 30,000 times monthly.