

Orders and Inventory Management

Orders and inventory management involves managing the status, flow, and state of inventory that is purchased through orders.

Orders and inventory management is used in any industry that needs a heavily optimized process for moving users or goods along a funnel to be purchased. Without this system of movement, purchases can not be completed and the business can not make money.

This use case is heavily dictated by the consumer's experience. They have high demands when it comes to receiving their purchases in a timely manner and they are easily frustrated when items go out of stock.

Challenges

Orders and inventory management is often cyclical, meaning that there are predicted moments in time for when there's a spike in orders (holiday shopping season, yearly sales, etc). Capacity planning can become a challenge when budgets are tight and companies don't want to overspend on infrastructure. At the same time, spikes in traffic can knock a system offline if the supporting infrastructure is not prepared to handle it. Downtime (unplanned or planned) is detrimental to this use case.

This use case naturally creates a distributed system since the customers, factories, headquarters, physical stores, etc. are all in different locations. This distributed nature creates challenges when it comes to achieving a consolidated view of your data, meeting data privacy regulations, and ensuring end-users experience minimal latency.

Additionally, the consistency of data is important to this use case because there needs to be several workflows in place to ensure that the placement and modification of orders is propagated downstream. If data consistency is only eventually consistent, it can create a lot of operational challenges.

Database requirements

A horizontally scalable database is needed to manage fluctuating workloads and reduce the overall infrastructure spend. Additionally, a database that can scale across multiple regions aids in reaching a distributed customer base and lowering end-user latency.

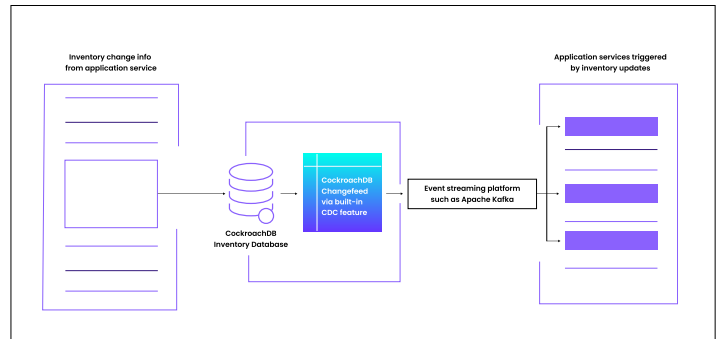
To make orders and inventory management systems less complex, it's useful if a database has built-in features such as ACID guarantees, replication, and change data capture (CDC). Data accuracy should be guaranteed and data should be replicated so it's never lost. CDC helps ensure all changes to the system are made to the data wherever it lives.

Why CockroachDB for orders and inventory management?

Orders and inventory management benefits from the distributed, resilient nature of CockroachDB and leverages the ability to horizontally scale up/down to save on yearly infrastructure costs.

Reference architecture

In the diagram, an application microservice updates the CockroachDB table with an inventory change. Since other services need that information, CockroachDB's CDC feature generates a changefeed that sends the inventory update to Kafka which triggers other application workflows.



Customer Story: CockroachDB powers Booksy's order management

Booksy is one of the most widely used scheduling apps for instant online appointment booking in hair salons, beauty salons, or barber shops.

They originally built their platform on PostgreSQL, but decided to switch to CockroachDB so that they could achieve 10x growth over the next 2 years. Since making the switch, Booksy's service level objective (SLO) is at the highest level of five nines. Their customers greatly benefit from a platform that's available 24/7, delivers low latency access, and can automatically scale to accommodate increases in traffic.

"Before migrating to CockroachDB we had challenges with resiliency, scaling, and keeping up with customer's needs... now we have a globally available platform."



Pawel Sobkowiak
Chief Technology Officer, Booksy

To learn more about Booksy's use case and read other customer stories, visit cockroachlabs.com/customers