

Creating an Accurate Inventory Picture at a Leading Retail Chain

Background and Business Problem

SVDS was engaged by a leading retail chain that needed to innovate and evolve their business towards new forms of commerce. The client realized that in order to compete with the likes of Amazon, any retailer needs to provide excellent visibility into item availability, inventory, and shipping options, as well as provide more incentive in terms of price and convenience. Since our client has a wide network of retail stores as well as warehouse locations, the differentiation they found was the ability for the user to order an item from a store or a warehouse location nearest to them. It was absolutely necessary for them to fully capitalize on this advantage.

An increase in online demand caused a greater impact on their back-end data systems, which couldn't keep up with incremental changes to the inventory picture of all the available items. This led to overselling—promising a product to a customer that is no longer available. This was especially challenging with popular, fast-moving items. Even a very short lag in information meant the possibility of overselling.

Understandably, this led to customer dissatisfaction and loss of business over time; in the near-term our client also lost money by offering unhappy guests gift cards. Any attempts to compensate for the overselling by withholding available stock meant losing out on legitimate business. The client needed an accurate inventory picture so that they could maximize sales while avoiding lost business.



A leading retail chain needed an up-to-date view of product inventory and shipping options available in their online store.

Silicon Valley Data Science built a solution that correctly showed the client's inventory, provided a richer set of shipping options, and prevented oversell.

The Challenge

Needed more robust options for order fulfillment to increase potential sales

Lack of visibility into real-time inventory picture caused overselling

Needed a system that showed accurate inventory and provided a set of shipping options

Lacked a robust architecture to handle the high volume of inventory updates during peak demand



Solution

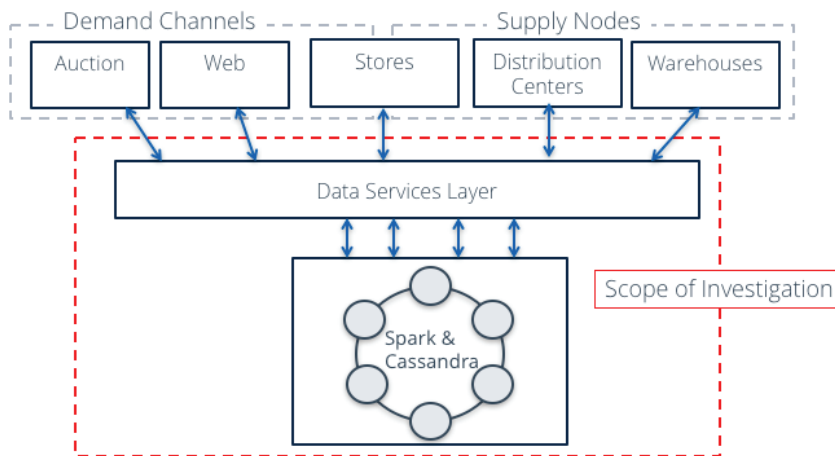
SVDS delivered a set of solutions to the client that provided visibility into inventory positions for stores, warehouses, and third party vendors. The system communicates:

- whether there is a limited quantity of an item available
- if an item is sold out
- whether the item will be available sooner, as long as it's ordered within a certain time frame

Working closely with the client delivery team, we fully integrated into their agile workflow. We worked on bridging the communication between the application delivery team, order processing team, and the enterprise services team in order to better coordinate all the different steps in the development process.

By using Cassandra's fast-write capability, the combined team was able to correctly track inventory updates even when the orders would come in at a peak rate. We created an algorithm that bypassed a limitation of Cassandra's CAS (Compare and Set) feature, which would periodically fail to inform the application of a failure in updating the inventory count.

We worked with the client team to create a set of Spring Boot-based microservices in order to facilitate modularization and reuse of services for different high-level use-cases. This required porting a large Groovy/Grails-based application, which was monolithic by nature. The team was able to create a set of classes that could be reused between the microservices, as well as document in a simple manner the status of the microservices in various stages of development (staging, QA, and production).



Our Approach

SVDS re-architected the way inventory was tracked in the client's high-throughput database

We built a set of microservices that allows for fast inventory tracking without oversell, and components on top of Cassandra that prevent oversell

We enabled communication between the client technical teams in order to come up with best practices for microservice architecture development

New Capabilities

Ability to update the inventory system based on processed orders and insight into delivery time estimates

Accurate inventory picture even during peak demand

Optimized set of shipping options

