CLOUD-BASED DATA COLLECTION FOR REPRICING IN ONLINE MARKETPLACES

Computer Science
TEAM 17.02

THE TEAM

STUDENTS: Ishan Tiwathia
Steven Nguyen
Sean Taing
Kevin Snyder
Abdul Almadani

ADVISOR: Dr. Eric Larson

SPONSORS: Jared Stiff
Marcel Englmaier
Jim Ohlund

BACKGROUND

What does CommerceHub do?
CommerceHub provides various cloud-based services that enable sellers to manage products, keep track of inventory, and stay competitive across multiple online marketplaces via a single platform.

What is repricing?
Repricing uses a rule-based system to automatically update prices for products based on competitor’s prices. It keeps vendors competitive and gives them a better chance of being the primary seller for a product, commonly referred to as “winning the buy box.” 90% of sales are made through the buy box.

What is our solution?
Our solution retrieves competitor’s prices in near real-time and sends this information to CommerceHub’s repricing engine. Fast processing of this data enables the repricing engine to quickly respond to price changes and make vendors more competitive.

PROJECT GOALS

• INCREASE THROUGHPUT
  – Respond to price changes more quickly

• SUPPORT MULTIPLE MARKETPLACES
  – Enable repricing in multiple marketplaces
  – Use other marketplaces’ data to make better repricing decisions

TECHNOLOGIES USED

1. AMAZON SQS: Marketplace queue containing price change notifications for merchants.
2. WALMART / JET SQS: These marketplaces do not send price change notifications. The SQS queues contain locations of data in S3. The actual data is retrieved from S3 by the readers.
3. AMAZON READER: Reads data from Amazon SQS queue. Formats the data and forwards it to the collector.
4. WALMART / JET READERS: Reads data from the S3 bucket. Formats the data and forwards to the collector.
5. AWS S3: Stores data for Walmart and Jet.
6. COLLECTOR: Collects the data from readers, including historical price data to make better pricing decisions.
7. PRICE CHANGE DATABASE: Stores all the data received from the readers for the enrichment process. Also used to check if merchant data has been changed.
8. AWS KINESIS LOG: Stores the enriched data.
9. REPRICING SYSTEM: Reprices merchant products using the data collected by readers. (not in scope)

ARCHITECTURE DESCRIPTION

ARCHITECTURE DIAGRAM

RESULTS

PRODUCTS PROCESSED PER SECOND

35
23
20
10
0
OLD
NEW (Avg.)
NEW (Best)

5x faster on average
9x faster at best

THREE MARKETPLACES SUPPORTED

Amazon
Walmart
Jet

CONCLUSION

• Efficiency — Increased message throughput.
  — Largely reduced message traffic by removing dependencies on scraping.

• Extensibility — Object-Oriented Design in both the reader and the collector to support additional marketplaces in the future.

• Modularity — All three marketplaces use a similar process for retrieving prices, making it easy to add additional marketplaces in the future.

• Enrichment — Data enrichment via storage of historical product prices for better repricing.