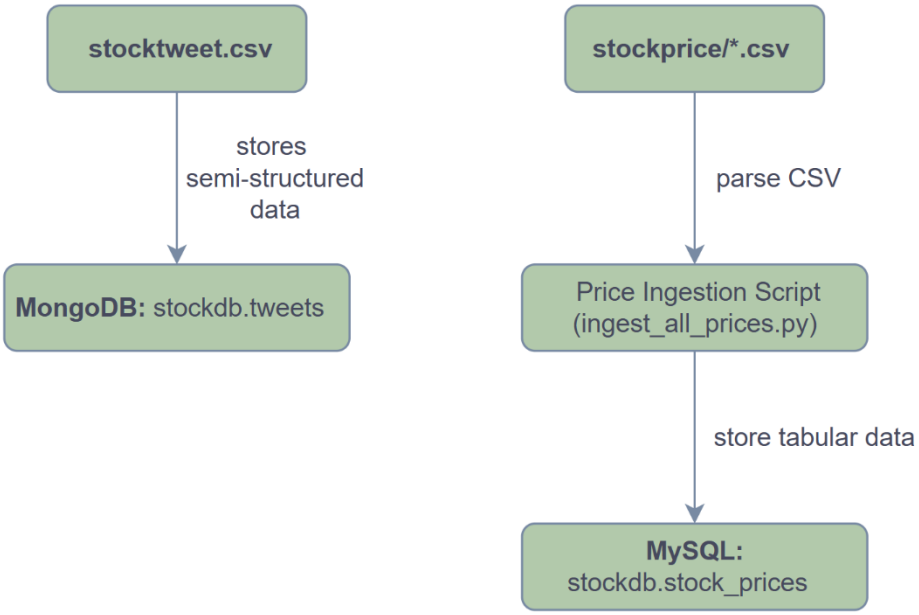
# Data Collection & Storage

## 1. Architecture of Data Flow



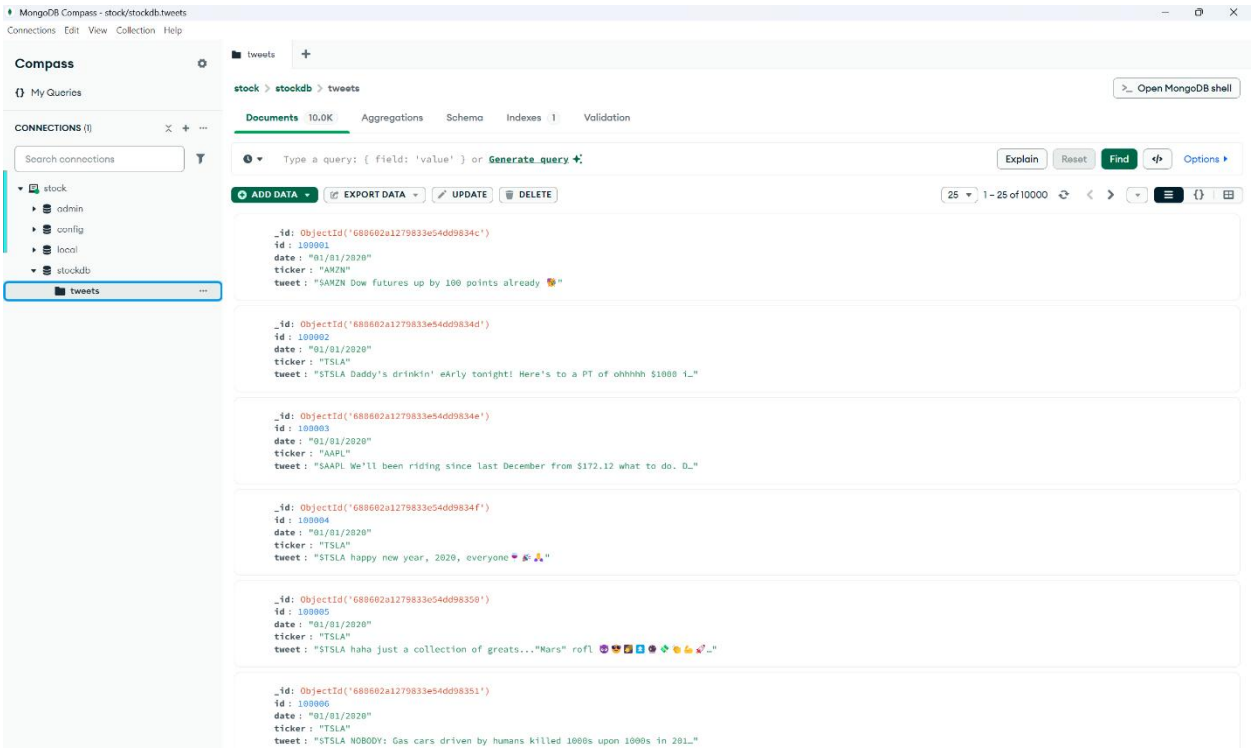

## 2. Tools Used

| Component | Tool | Reason |
|---|---|---|
| NoSQL Database | MongoDB | Flexible document model, schema-on-read, native JSON storage, easy indexing on ticker & date. |
| Rational DB | MySQL | ACID compliance, efficient indexed time-series queries. |
| Scripting | Python3 | Rich ecosystem like (pymongo, mysql-connector,csv), cross-platform; concise bulk loading. |
| GUI Validation | MongoDB Compass/ MySQL Workbench | Visual data inspection, index and schema management. |

# 3. Storage Design Decisions

## 1. MongoDB (stockdb.tweets)

### A) Schema:




### B) Storage Engine & Configuration

- WiredTiger for document compression, high concurrency, and write throughput.
- Journaling enabled for crash recovery.

### C) Scalability & Availability

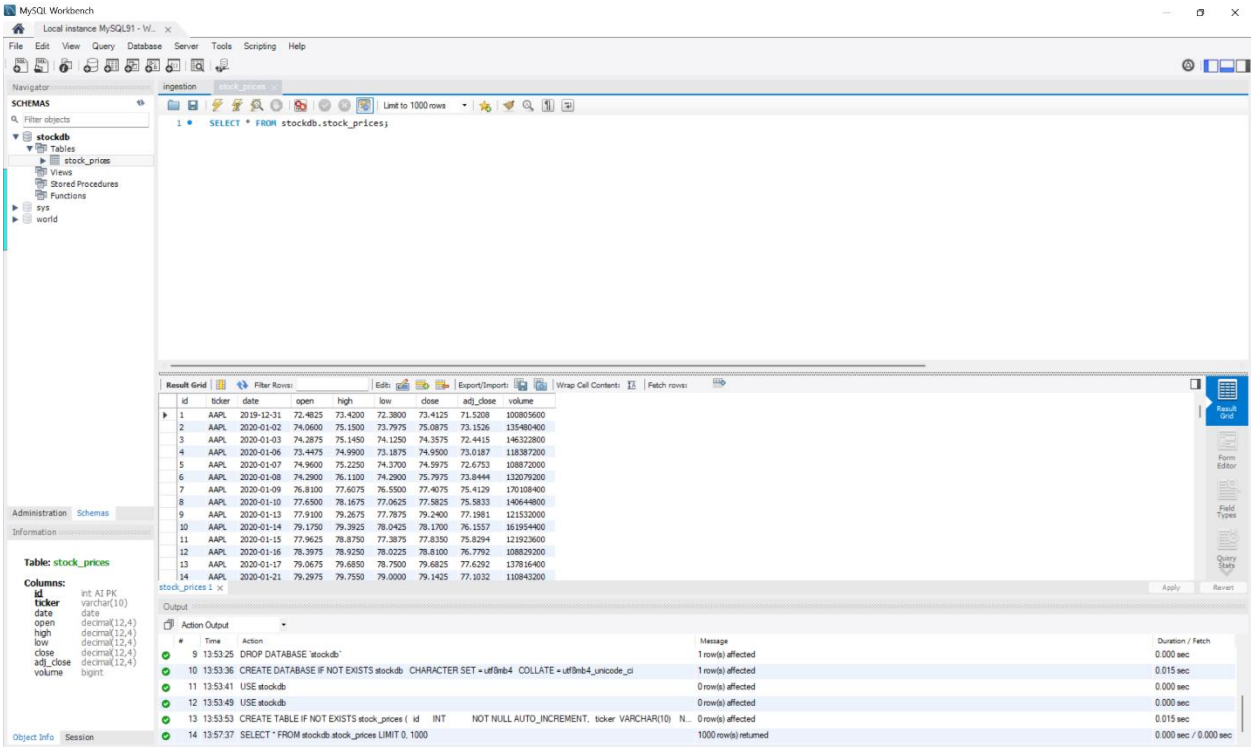
- Replica Set for high availability and automatic failover.
- Sharding strategy (hash-shard on ticker) when collection grows beyond single-node capacity.

### D) Retention & Archiving

- Archive older raw tweets to cold storage or S3 after sentiment extraction.

## 2. MySQL (stockdb.stock_prices)

### A) Schema:




### B) Partitioning & Maintenance

- Range Partitioning by date (e.g., monthly or yearly) to limit scan scope and speed up purging old partitions.
- Event Scheduler or external job to drop/archive partitions beyond a retention period.

### C) High Availability & Scaling

- Master-Replica setup for read-scaling and zero-downtime failover.
- Evaluate MySQL Cluster or Amazon Aurora for automatic sharding and multi-AZ resilience if volumes spike.

### D) Backup & Recovery

- Point-in-Time Recovery via binary logs.
- Logical Backups (mysqldump) for schema snapshots; Physical Backups (Percona XtraBackup) for large datasets.