Báo cáo đồ án 04 - Streaming

Thành viên nhóm

MSSV	Họ và tên	Công việc
22120334	Nguyễn Quang Thắng	Thực hiện phần Extract, viết báo cáo
22120336	Võ Tuấn Thành	Thực hiện phần Load, viết báo cáo
22120436	Lê Cao Tuấn Vũ	Thực hiện phần Transform, viết báo cáo

Extract

1. **Thu thập dữ liệu**:
   - `binance_client.py` sử dụng requests để gọi endpoint công khai của Binance: `https://api.binance.com/api/v3/ticker/price?symbol=BTCUSDT`
   - Dữ liệu trả về có dạng:

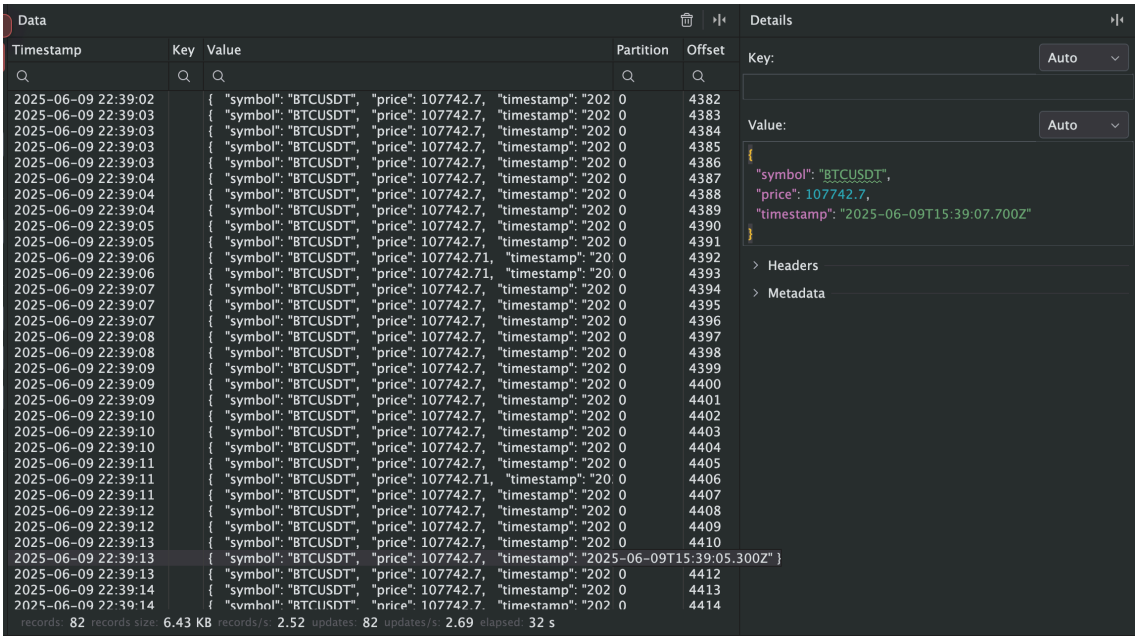
     ```
     {
       "symbol": "BTCUSDT",
       "price": "107878.81"
        }
     ```

   - Giá trị price được chuyển thành float để đảm bảo tính toán chính xác.
2. **Xử lý timestamp**:
   - `timestamp_handler.py` lấy thời gian UTC bằng datetime và dateutil.tz.tzutc().
   - Timestamp được làm tròn đến bội số của 100ms (100,000 micro giây):

     ```
     microseconds = now.microsecond
     rounding_interval = frequency_ms * 1000
     rounded_microseconds = (microseconds // rounding_interval) * roundi
     ng_interval
     ```

- Kết quả là chuỗi ISO8601, ví dụ: "2025-06-09T15:34:02.100Z".

3. **Gửi dữ liệu vào Kafka**:
   - `kafka_producer.py` khởi tạo **Kafka** producer với cấu hình từ `kafka.yaml`.
   - Dữ liệu JSON được mã hóa thành UTF-8 và gửi vào topic **btc-price**.
4. **Tần suất**:
   - Ban đầu, main.py sử dụng vòng lặp tuần tự với time.sleep(0.1) để đạt tần suất 100ms.
   - Tuy nhiên, do độ trễ mạng (~150ms), nhóm đã chuyển sang dùng **đa luồng (threading)** với 5 thread để fetch dữ liệu song song, đảm bảo ít nhất 1 message mỗi 100ms.



- Một số phần events được produce:

```
{
  "symbol": "BTCUSDT",
  "price": 107742.7,
  "timestamp": "2025-06-09T15:39:07.700Z"
}{
  "symbol": "BTCUSDT",
```

```
  "price": 107742.7,
  "timestamp": "2025-06-09T15:39:07.800Z"
}{
  "symbol": "BTCUSDT",
  "price": 107742.7,
  "timestamp": "2025-06-09T15:39:07.900Z"
}{
  "symbol": "BTCUSDT",
  "price": 107742.7,
  "timestamp": "2025-06-09T15:39:08.000Z"
}{
  "symbol": "BTCUSDT",
  "price": 107742.7,
  "timestamp": "2025-06-09T15:39:08.100Z"
}{
  "symbol": "BTCUSDT",
  "price": 107742.7,
  "timestamp": "2025-06-09T15:39:08.200Z"
}{
  "symbol": "BTCUSDT",
  "price": 107742.7,
  "timestamp": "2025-06-09T15:39:08.300Z"
}{
  "symbol": "BTCUSDT",
  "price": 107742.7,
  "timestamp": "2025-06-09T15:39:08.400Z"
}{
  "symbol": "BTCUSDT",
  "price": 107742.7,
  "timestamp": "2025-06-09T15:39:08.500Z"
}
```


Transform

Moving

Tổng quan

Giai đoạn **Moving** tính toán **trung bình động** (moving average) và **độ lệch chuẩn** (standard deviation) của giá Bitcoin theo các cửa sổ thời gian (30 giây, 1 phút, 5 phút, 15 phút, 30 phút, 1 giờ). Code đọc dữ liệu từ topic `btc-price`, xử lý streaming, và ghi kết quả vào topic `btc-price-moving`.

Các Giai đoạn Xử lý trong Code Giai đoạn Moving

1. **Khởi tạo SparkSession và Logging** (`init_spark_session`):
   - Thiết lập SparkSession với gói Kafka, checkpoint (`./checkpoint_data`), và 10 partition.
   - Cấu hình logging mức `INFO` để ghi log xử lý.
   - **Đầu ra**: SparkSession và hệ thống log.
2. **Định nghĩa Schema và Cửa sổ** (`get_data_schema_and_windows`):
   - Schema JSON: `timestamp` (Timestamp), `symbol` (String), `price` (Double).
   - Cửa sổ: 30s, 1m, 5m, 15m, 30m, 1h; độ trượt (ví dụ: 30s → 5s).
   - **Đầu ra**: Schema và dictionary cửa sổ.
3. **Đọc Stream từ Kafka** (`fetch_kafka_stream`):
   - Đọc từ topic `btc-price` với `startingOffsets="earliest"`, `maxOffsetsPerTrigger=100`.
   - **Đầu ra**: DataFrame streaming chứa JSON thô.
4. **Phân tích JSON** (`parse_kafka_data`):
   - Chuyển JSON thành DataFrame với cột `timestamp`, `symbol`, `price` bằng `from_json`.
   - **Đầu ra**: DataFrame có cấu trúc.
5. **Tính Thống kê window** (`compute_window_stats`):
   - Áp dụng watermark 10s, nhóm theo cửa sổ trượt và `symbol`.
   - Tính `mean_price` (trung bình) và `std_dev_price` (độ lệch chuẩn).

- **Đầu ra**: DataFrame thống kê cho một cửa sổ.

6. **Gộp window** (`aggregate_windows`):
   - Gộp DataFrame từ các cửa sổ bằng `unionByName`.
   - Nhóm theo `timestamp`, `symbol`, tạo cột `windows` (danh sách JSON) bằng `collect_list` và `to_json`.
   - **Đầu ra**: DataFrame định dạng JSON.
7. **Ghi vào Kafka** (`publish_to_kafka`):
   - Chuyển DataFrame thành JSON cột `value`, ghi vào topic `btc-price-moving` với `outputMode="update"`.
   - **Đầu ra**: Dữ liệu JSON trong `btc-price-moving`.
8. **Điều phối Pipeline** (`execute_pipeline`):
   - Gọi các hàm theo thứ tự, xử lý lỗi bằng `try-except`.
   - **Đầu ra**: Pipeline hoàn chỉnh.

Quá trình chạy

- Chạy file 22120334_moving.py

```
└─❯ python src/transform/22120334_moving.py

2025-06-13 16:53:40,525 - INFO - Khởi tạo SparkSession...
:: loading settings :: url = jar:file:/usr/local/spark/jars/ivy-2.5.1.jar!/org/apache/ivy/core/settings/ivysettings.xml
Ivy Default Cache set to: /home/html/.ivy2/cache
The jars for the packages stored in: /home/html/.ivy2/jars
org.apache.spark#spark-sql-kafka-0-10_2.12 added as a dependency
:: resolving dependencies :: org.apache.spark#spark-submit-parent-acc992a5-f6b5-46c9-84d0-c8c39ff3d5fa;1.0
        confs: [default]
        found org.apache.spark#spark-sql-kafka-0-10_2.12;3.5.0 in central
        found org.apache.spark#spark-token-provider-kafka-0-10_2.12;3.5.0 in central
        found org.apache.kafka#kafka-clients;3.4.1 in central
        found org.lz4#lz4-java;1.8.0 in central
        found org.xerial.snappy#snappy-java;1.1.10.3 in central
        found org.slf4j#slf4j-api;2.0.7 in central
        found org.apache.hadoop#hadoop-client-runtime;3.3.4 in central
        found org.apache.hadoop#hadoop-client-api;3.3.4 in central
        found commons-logging#commons-logging;1.1.3 in local-m2-cache
        found com.google.code.findbugs#jsr305;3.0.0 in central
        found org.apache.commons#commons-pool2;2.11.1 in central
:: resolution report :: resolve 960ms :: artifacts dl 51ms
        :: modules in use:
        com.google.code.findbugs#jsr305;3.0.0 from central in [default]
        commons-logging#commons-logging;1.1.3 from local-m2-cache in [default]
        org.apache.commons#commons-pool2;2.11.1 from central in [default]
        org.apache.hadoop#hadoop-client-api;3.3.4 from central in [default]
        org.apache.hadoop#hadoop-client-runtime;3.3.4 from central in [default]
        org.apache.kafka#kafka-clients;3.4.1 from central in [default]
        org.apache.spark#spark-sql-kafka-0-10_2.12;3.5.0 from central in [default]
        org.apache.spark#spark-token-provider-kafka-0-10_2.12;3.5.0 from central in [default]
        org.lz4#lz4-java;1.8.0 from central in [default]
        org.slf4j#slf4j-api;2.0.7 from central in [default]
        org.xerial.snappy#snappy-java;1.1.10.3 from central in [default]
        ---------------------------------------------------------------------
        |                  |            modules            ||   artifacts   |
        |       conf       | number| search|dwnlded|evicted|| number|dwnlded|
```

- Xem dữ liệu đầu ra với lệnh `~/kafka/bin/kafka-console-consumer.sh --topic btc-price-moving --bootstrap-server localhost:9092 --from-beginning`
    - Kết quả


```
{"timestamp":"2025-06-13T16:12:55.000+07:00","symbol":"BTCUSDT","windows":["{\"window\":\"30s\",\"avg_price\":104695.98813333342,\"std_price\":4.4491088367756
95}","{\"window\":\"30s\",\"avg_price\":104696.93131428579,\"std_price\":5.178624055990898}","{\"window\":\"30s\",\"avg_price\":104698.73581132082,\"std_price
\":6.628307079261776}"]}
{"timestamp":"2025-06-13T16:25:00.000+07:00","symbol":"BTCUSDT","windows":["{\"window\":\"30m\",\"avg_price\":104748.42830188674,\"std_price\":0.0037906002414
46616}","{\"window\":\"30m\",\"avg_price\":104755.9911111111,\"std_price\":5.523890523928735}","{\"window\":\"30m\",\"avg_price\":104755.23422924898,\"std_pri
ce\":8.344364977861398}","{\"window\":\"30m\",\"avg_price\":104746.27070821523,\"std_price\":16.18582437123167 5}","{\"window\":\"30m\",\"avg_price\":104742.37
865342153,\"std_price\":16.406092859585872}","{\"window\":\"30m\",\"avg_price\":104735.59097649179,\"std_price\":20.91983727943894}","{\"window\":\"30m\",\"av
g_price\":104726.98049004588,\"std_price\":28.277652236646162}","{\"window\":\"30m\",\"avg_price\":104718.71845949531,\"std_price\":34.188115847129275}","{\"w
indow\":\"30m\",\"avg_price\":104710.9172684642,\"std_price\":38.781149157424686}","{\"window\":\"30m\",\"avg_price\":104707.67260230846,\"std_price\":37.8931
8755077146}","{\"window\":\"30m\",\"avg_price\":104705.04421652418,\"std_price\":36.949988087355486}","{\"window\":\"30m\",\"avg_price\":104702.87180398958,\"
std_price\":36.007277399645375}","{\"window\":\"30m\",\"avg_price\":104700.41755786112,\"std_price\":35.59711333432001}","{\"window\":\"30m\",\"avg_price\":10
4697.87943828526,\"std_price\":35.4430154982923}","{\"window\":\"30m\",\"avg_price\":104694.6615003441,\"std_price\":36.42855193706283}","{\"window\":\"30m\",
\"avg_price\":104693.09238892465,\"std_price\":36.04437454710651}","{\"window\":\"30m\",\"avg_price\":104692.77854809437,\"std_price\":35.11036190562712}","{\
"window\":\"30m\",\"avg_price\":104690.7537478608,\"std_price\":35.12245583885911}","{\"window\":\"30m\",\"avg_price\":104690.14756071236,\"std_price\":34.256
88944103972}","{\"window\":\"30m\",\"avg_price\":104689.03780337941,\"std_price\":33.71018943372219}","{\"window\":\"30m\",\"avg_price\":104688.69957135897,\"
std_price\":33.19448113743068}","{\"window\":\"30m\",\"avg_price\":104690.42535531815,\"std_price\":33.58308524632066}","{\"window\":\"30m\",\"avg_price\":104
690.2091122947,\"std_price\":32.868971126090265}","{\"window\":\"30m\",\"avg_price\":104692.45919677005,\"std_price\":34.097363212615676}","{\"window\":\"30m\
",\"avg_price\":104694.45364044026,\"std_price\":34.79534967446397}","{\"window\":\"30m\",\"avg_price\":104696.69400313355,\"std_price\":35.88475693828955}","
{\"window\":\"30m\",\"avg_price\":104698.12995099885,\"std_price\":35.96952866130631}","{\"window\":\"30m\",\"avg_price\":104699.77027969484,\"std_price\":36.
446195780582904}","{\"window\":\"30m\",\"avg_price\":104700.96558710128,\"std_price\":36.34690906978878}","{\"window\":\"30m\",\"avg_price\":104702.2972468675
7,\"std_price\":36.45257445581665}","{\"window\":\"30m\",\"avg_price\":104703.5388928922,\"std_price\":36.56164934065518}","{\"window\":\"30m\",\"avg_price\":
104704.76730098316,\"std_price\":36.64305641565056}","{\"window\":\"30m\",\"avg_price\":104705.90275745462,\"std_price\":36.64262554311057 6}","{\"window\":\"3
0m\",\"avg_price\":104706.16444079926,\"std_price\":36.2332442666687}","{\"window\":\"30m\",\"avg_price\":104706.93185635675,\"std_price\":35.98814317023089}"
,"{\"window\":\"15m\",\"avg_price\":104706.40238095241,\"std_price\":5.8095474017965305}","{\"window\":\"30m\",\"avg_price\":104707.06560934418,\"std_price\":
35.519909995017}","{\"window\":\"15m\",\"avg_price\":104706.02552147246,\"std_price\":5.972220377592372}","{\"window\":\"30m\",\"avg_price\":104707.0306378319
```


- Ví dụ một số dữ liệu đầu ra

```
{
  "timestamp": "2025-06-13T10:24:00Z",
  "symbol": "BTCUSDT",
```

```
  "windows": [
    {"window": "30s", "avg_price": 104710.50, "std_price": 8.25},
    {"window": "1m", "avg_price": 104709.75, "std_price": 10.50},
    {"window": "5m", "avg_price": 104708.20, "std_price": 12.75},
    {"window": "15m", "avg_price": 104706.40, "std_price": 15.80},
    {"window": "30m", "avg_price": 104707.29, "std_price": 34.62},
    {"window": "1h", "avg_price": 104705.10, "std_price": 22.90}
  ]
}
```


Z-Score

Tổng quan

Giai đoạn **Z-Score** tính toán **z-score** của giá Bitcoin so với trung bình động và độ lệch chuẩn được tính từ giai đoạn Moving, dựa trên các cửa sổ thời gian (30 giây, 1 phút, 5 phút, 15 phút, 30 phút, 1 giờ). Code đọc dữ liệu từ topic btc-price (giá gốc) và btc-price-moving (thống kê trung bình động), xử lý streaming, và ghi kết quả vào topic btc-price-zscore.

Các Giai đoạn Xử lý trong Code Giai đoạn Z-Score

1. **Khởi tạo SparkSession** (init_spark_session):
    - Thiết lập SparkSession với gói Kafka, checkpoint (./checkpoints/zscore_main), và 4 partition.
    - Cấu hình logging mức ERROR để giảm thiểu thông báo rác.
    - **Đầu ra**: SparkSession đã cấu hình.
2. **Định nghĩa Schema** (get_price_schema, get_moving_schema):
    - Schema cho btc-price: timestamp (Timestamp), symbol (String), price (Double).
    - Schema cho btc-price-moving: timestamp (Timestamp), symbol (String), windows (Array chứa window (String), avg_price (Double), std_price (Double)).

- **Đầu ra**: Schema cho dữ liệu JSON từ Kafka.

3. **Đọc Stream Giá từ Kafka** (load_price_stream):
   - Đọc từ topic btc-price với startingOffsets="earliest".
   - Phân tích JSON thành DataFrame với cột timestamp, symbol, price bằng from_json.
   - Áp dụng watermark 10 giây để xử lý dữ liệu trễ.
   - **Đầu ra**: DataFrame streaming chứa giá Bitcoin.
4. **Đọc Stream Thống kê từ Kafka** (load_moving_stream):
   - Đọc từ topic btc-price-moving với startingOffsets="earliest".
   - Phân tích JSON và mở rộng mảng windows bằng explode để tạo DataFrame với cột timestamp, symbol, window, avg_price, std_price.
   - Áp dụng watermark 10 giây.
   - **Đầu ra**: DataFrame streaming chứa thống kê trung bình động.
5. **Tính Z-Score** (compute_zscores):
   - Join DataFrame giá (btc-price) và thống kê (btc-price-moving) dựa trên timestamp và symbol.
   - Tính z-score: (price - avg_price) / std_price khi std_price > 0.0001, nếu không thì gán 0.0.
   - **Đầu ra**: DataFrame chứa z-score cho từng cửa sổ.
6. **Định dạng Kết quả Z-Score** (prepare_zscore_output):
   - Loại bỏ bản sao bằng cách nhóm theo timestamp, symbol, window và lấy giá trị z-score lớn nhất.
   - Nhóm theo timestamp, symbol, gộp z-score của các cửa sổ vào mảng zscores bằng collect_list.
   - Chuyển thành JSON với cột value chứa timestamp, symbol, zscores.
   - **Đầu ra**: DataFrame định dạng JSON.
7. **Ghi vào Kafka** (stream_to_kafka):

- Ghi cột value (JSON) vào topic btc-price-zscore với outputMode="append".
- Sử dụng checkpoint tại ../checkpoints/zscore để đảm bảo tính nhất quán.
- **Đầu ra**: Dữ liệu JSON trong btc-price-zscore.

8. **Ghi ra Console** (stream_to_console):
   - Ghi DataFrame ra console để kiểm tra với outputMode="append".
   - **Đầu ra**: Kết quả z-score hiển thị trên console.
9. **Điều phối Pipeline** (run_pipeline):
   - Gọi các hàm theo thứ tự, khởi động stream Kafka và console.
   - Chờ pipeline hoàn thành bằng awaitTermination.
   - **Đầu ra**: Pipeline z-score hoàn chỉnh.

Quá trình chạy

- Chạy file 22120334_zscore.py

```
└─> python src/transform/22120334_zscore.py

:: loading settings :: url = jar:file:/usr/local/spark/jars/ivy-2.5.1.jar!/org/apache/ivy/core/settings/ivysettings.xml
Ivy Default Cache set to: /home/html/.ivy2/cache
The jars for the packages stored in: /home/html/.ivy2/jars
org.apache.spark#spark-sql-kafka-0-10_2.12 added as a dependency
:: resolving dependencies :: org.apache.spark#spark-submit-parent-fed53c9c-b77c-4773-b58c-70855823b2e6;1.0
        confs: [default]
        found org.apache.spark#spark-sql-kafka-0-10_2.12;3.5.0 in central
        found org.apache.spark#spark-token-provider-kafka-0-10_2.12;3.5.0 in central
        found org.apache.kafka#kafka-clients;3.4.1 in central
        found org.lz4#lz4-java;1.8.0 in central
        found org.xerial.snappy#snappy-java;1.1.10.3 in central
        found org.slf4j#slf4j-api;2.0.7 in central
        found org.apache.hadoop#hadoop-client-runtime;3.3.4 in central
        found org.apache.hadoop#hadoop-client-api;3.3.4 in central
        found commons-logging#commons-logging;1.1.3 in local-m2-cache
        found com.google.code.findbugs#jsr305;3.0.0 in central
        found org.apache.commons#commons-pool2;2.11.1 in central
:: resolution report :: resolve 1208ms :: artifacts dl 53ms
        :: modules in use:
        com.google.code.findbugs#jsr305;3.0.0 from central in [default]
        commons-logging#commons-logging;1.1.3 from local-m2-cache in [default]
        org.apache.commons#commons-pool2;2.11.1 from central in [default]
        org.apache.hadoop#hadoop-client-api;3.3.4 from central in [default]
        org.apache.hadoop#hadoop-client-runtime;3.3.4 from central in [default]
        org.apache.kafka#kafka-clients;3.4.1 from central in [default]
        org.apache.spark#spark-sql-kafka-0-10_2.12;3.5.0 from central in [default]
        org.apache.spark#spark-token-provider-kafka-0-10_2.12;3.5.0 from central in [default]
        org.lz4#lz4-java;1.8.0 from central in [default]
        org.slf4j#slf4j-api;2.0.7 from central in [default]
        org.xerial.snappy#snappy-java;1.1.10.3 from central in [default]
        ---------------------------------------------------------------------
        |                  |            modules            ||   artifacts   |
        |       conf       | number| search|dwnlded|evicted|| number|dwnlded|
        ---------------------------------------------------------------------
```

- Xem dữ liệu đầu ra với lệnh `~/kafka/bin/kafka-console-consumer.sh --topic btc-price-zscore --bootstrap-server localhost:9092 --from-beginning`

```
└─❯ ~/kafka/bin/kafka-console-consumer.sh --topic btc-price-zscore --bootstrap-server localhost:9092 --from-beginning
{"timestamp":"2025-06-13T12:12:10.000Z","symbol":"BTCUSDT","zscores":[{"window":"1m","zscore_price":515.7854070585632},{"window":"30s","zscore_price":3.030212
8931972234}]}
{"timestamp":"2025-06-13T12:18:25.000Z","symbol":"BTCUSDT","zscores":[{"window":"30s","zscore_price":0.0}]}
{"timestamp":"2025-06-13T12:13:15.000Z","symbol":"BTCUSDT","zscores":[{"window":"30s","zscore_price":14.217338824348131}]}
{"timestamp":"2025-06-13T12:12:25.000Z","symbol":"BTCUSDT","zscores":[{"window":"30s","zscore_price":-0.123880013996474}]}
{"timestamp":"2025-06-13T12:17:30.000Z","symbol":"BTCUSDT","zscores":[{"window":"30s","zscore_price":0.0},{"window":"5m","zscore_price":22.85030912540455},{"w
indow":"1m","zscore_price":3133.1092666775035}]}
{"timestamp":"2025-06-13T12:13:00.000Z","symbol":"BTCUSDT","zscores":[{"window":"15m","zscore_price":62.058843188020845},{"window":"30s","zscore_price":6.3915
55847313442},{"window":"1m","zscore_price":6.654786243574772}]}
{"timestamp":"2025-06-13T12:18:00.000Z","symbol":"BTCUSDT","zscores":[{"window":"1m","zscore_price":0.0},{"window":"30s","zscore_price":-1.6635306417456328}]}
{"timestamp":"2025-06-13T12:17:55.000Z","symbol":"BTCUSDT","zscores":[{"window":"30s","zscore_price":-2.1541965868574993}]}
{"timestamp":"2025-06-13T12:11:45.000Z","symbol":"BTCUSDT","zscores":[{"window":"30s","zscore_price":-2.411566738995351}]}
{"timestamp":"2025-06-13T12:10:55.000Z","symbol":"BTCUSDT","zscores":[{"window":"30s","zscore_price":3098.127563398499}]}
{"timestamp":"2025-06-13T12:14:40.000Z","symbol":"BTCUSDT","zscores":[{"window":"1m","zscore_price":16.307620982809347},{"window":"30s","zscore_price":0.0}]}
{"timestamp":"2025-06-13T12:18:45.000Z","symbol":"BTCUSDT","zscores":[{"window":"30s","zscore_price":-0.3944805211990896}]}
{"timestamp":"2025-06-13T12:18:05.000Z","symbol":"BTCUSDT","zscores":[{"window":"30s","zscore_price":-1.2432421015749753}]}
{"timestamp":"2025-06-13T12:13:25.000Z","symbol":"BTCUSDT","zscores":[{"window":"30s","zscore_price":8.459287822768598}]}
{"timestamp":"2025-06-13T12:13:10.000Z","symbol":"BTCUSDT","zscores":[{"window":"30s","zscore_price":3.494045602215803},{"window":"1m","zscore_price":10156.75
4511042882}]}
{"timestamp":"2025-06-13T12:12:50.000Z","symbol":"BTCUSDT","zscores":[{"window":"1m","zscore_price":19720.66774601702},{"window":"30s","zscore_price":5712.912
446283232}]}
{"timestamp":"2025-06-13T12:12:00.000Z","symbol":"BTCUSDT","zscores":[{"window":"1m","zscore_price":1254.5350078788767},{"window":"30s","zscore_price":3.03690
37235373306}]}
{"timestamp":"2025-06-13T12:15:50.000Z","symbol":"BTCUSDT","zscores":[{"window":"5m","zscore_price":31047.762481534155},{"window":"1m","zscore_price":778.0433
749279706},{"window":"30s","zscore_price":12.958498701193434}]}
{"timestamp":"2025-06-13T12:15:55.000Z","symbol":"BTCUSDT","zscores":[{"window":"30s","zscore_price":6.707417637669432}]}
{"timestamp":"2025-06-13T12:12:30.000Z","symbol":"BTCUSDT","zscores":[{"window":"5m","zscore_price":11.664101668351028},{"window":"1m","zscore_price":2.383429
951325084},{"window":"30s","zscore_price":-0.648896290836535}]}
{"timestamp":"2025-06-13T12:15:05.000Z","symbol":"BTCUSDT","zscores":[{"window":"30s","zscore_price":50.849116406669424}]}
{"timestamp":"2025-06-13T12:17:00.000Z","symbol":"BTCUSDT","zscores":[{"window":"30s","zscore_price":9516.57244521058},{"window":"15m","zscore_price":42.57017
2729226435},{"window":"1m","zscore_price":11.466538360168597}]}
{"timestamp":"2025-06-13T12:16:30.000Z","symbol":"BTCUSDT","zscores":[{"window":"1m","zscore_price":-0.15533215285556762},{"window":"30s","zscore_price":1.423
8551869353915}]}
{"timestamp":"2025-06-13T12:10:15.000Z","symbol":"BTCUSDT","zscores":[{"window":"30s","zscore_price":4194.884043077141}]}
{"timestamp":"2025-06-13T12:17:50.000Z","symbol":"BTCUSDT","zscores":[{"window":"30s","zscore_price":-2.331311159612019},{"window":"1m","zscore_price":-3.0421
17812907144}]}
```

- Ví dụ một số dữ liệu đầu ra

```
{
  "timestamp": "2025-06-13T22:00:00Z",
  "symbol": "BTCUSDT",
  "zscores": [
    {"window": "30s", "zscore_price": 1.15},
    {"window": "15m", "zscore_price": -0.25}
  ]
}
```

Load

Báo cáo phần Load - Lab 04 Spark Streaming

1. Tổng quan

Phần Load trong pipeline ETL có nhiệm vụ đọc dữ liệu Z-score đã được tính toán từ Kafka topic `btc-price-zscore` và lưu trữ vào MongoDB theo các collection tương ứng

với từng cửa sổ thời gian. Đây là giai đoạn cuối cùng trong pipeline, đảm bảo dữ liệu được lưu trữ bền vững để phục vụ cho các truy vấn và phân tích sau này.

1.1. Kiến trúc tổng quan

```
Kafka Topic (btc-price-zscore) → Spark Structured Streaming → MongoDB C
ollections
```

1.2. Các thành phần chính

- **Spark Structured Streaming**: Đọc dữ liệu streaming từ Kafka
- **MongoDB Connector**: Kết nối và ghi dữ liệu vào MongoDB
- **Schema Handler**: Quản lý schema và collection mapping
- **Stream Writer**: Xử lý việc ghi dữ liệu theo batch

2. Cấu trúc mã nguồn

2.1. Cấu trúc thư mục

```
src/load/
├── __init__.py
├── main.py              # Entry point chính
├── config.py            # Quản lý cấu hình
├── mongodb_connector.py   # Kết nối MongoDB
├── schema_handler.py      # Xử lý schema và collection
└── stream_writer.py       # Ghi dữ liệu streaming
```

2.2. Cấu hình hệ thống

Các file cấu hình được đặt trong thư mục `config/`:

- **mongodb.yaml**: Cấu hình kết nối MongoDB và các collection
- **kafka.yaml**: Cấu hình Kafka consumer
- **spark.yaml**: Cấu hình Spark application

3. Thiết lập MongoDB

3.1. Cài đặt và khởi chạy MongoDB

MongoDB được thiết lập thông qua `docker-compose.yml`

3.2. Cấu hình Collections

Hệ thống tự động tạo các collection tương ứng với từng cửa sổ thời gian:

- `btc-price-zscore-30s`: Dữ liệu window 30 giây
- `btc-price-zscore-1m`: Dữ liệu window 1 phút
- `btc-price-zscore-5m`: Dữ liệu window 5 phút
- `btc-price-zscore-15m`: Dữ liệu window 15 phút
- `btc-price-zscore-30m`: Dữ liệu window 30 phút
- `btc-price-zscore-1h`: Dữ liệu window 1 giờ

4. Schema dữ liệu

4.1. Schema đầu vào (Z-score data)

```
StructType([
    StructField("timestamp", TimestampType(), True),
    StructField("symbol", StringType(), True),
    StructField("zscores", ArrayType(
        StructType([
```

```
            StructField("window", StringType(), True),
            StructField("zscore_price", DoubleType(), True)
        ])
    ), True)
])
```

4.2. Schema trong MongoDB

Mỗi document trong MongoDB collection có cấu trúc:

```
{
  "_id": ObjectId("..."),
  "timestamp": ISODate("2024-01-01T00:00:00.000Z"),
  "symbol": "BTCUSDT",
  "window": "30s",
  "zscore_price": 1.23456
}
```

5. Luồng xử lý dữ liệu

5.1. Đọc dữ liệu từ Kafka

```
df = spark.readStream \
    .format("kafka") \
    .option("kafka.bootstrap.servers", kafka_bootstrap_servers) \
    .option("subscribe", "btc-price-zscore") \
    .option("startingOffsets", "latest") \
    .load()
```

5.2. Phân tích JSON và áp dụng schema

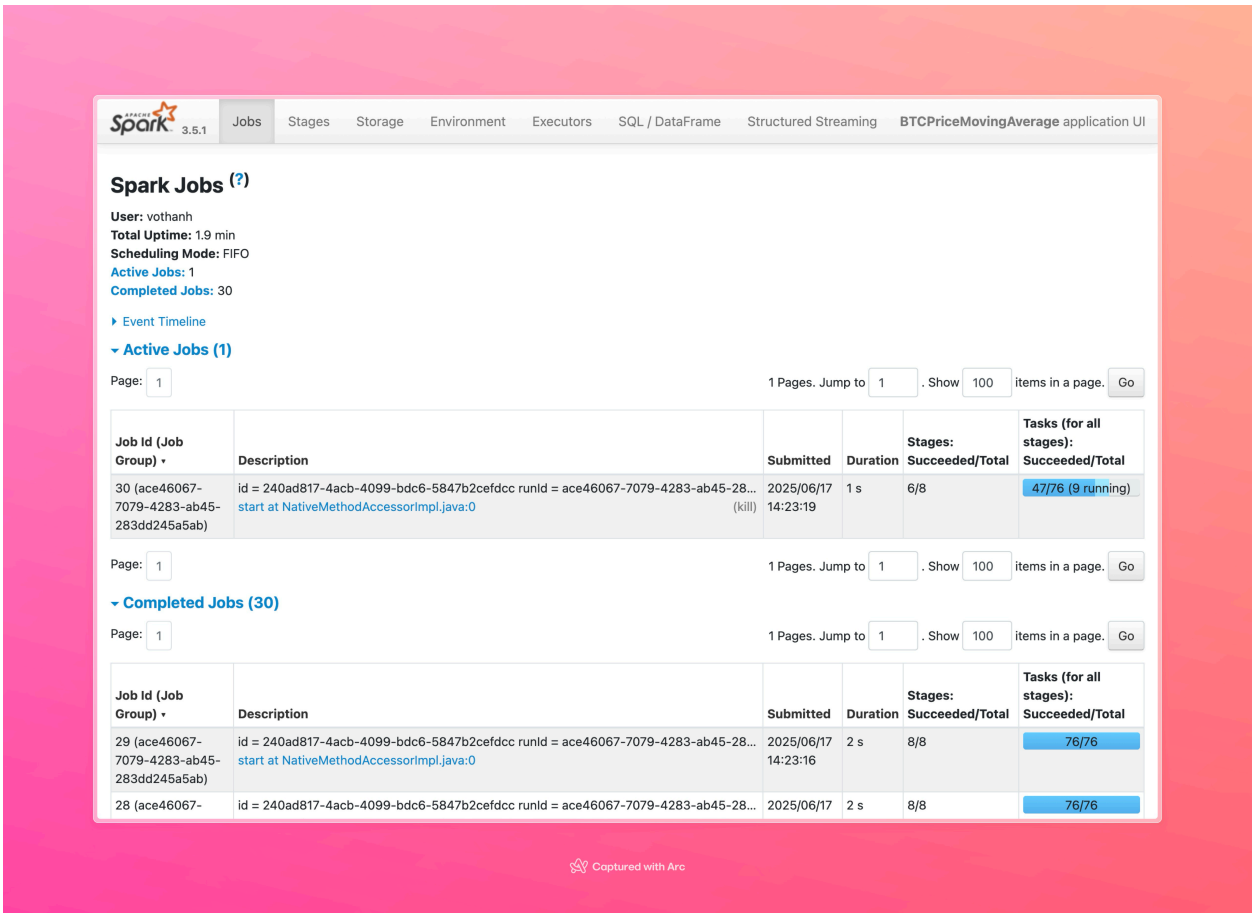
```
parsed_df = df.select(
    from_json(
        col("value").cast("string"),
```

```
        get_zscore_schema()
    ).alias("data")
).select("data.*")
```

5.3. Explode dữ liệu zscores

Do mỗi record chứa array các Z-score cho nhiều cửa sổ thời gian khác nhau, cần explode để tách thành các record riêng biệt:

```
exploded_df = df.select(
    "timestamp",
    "symbol",
    explode("zscores").alias("zscore_data")
).select(
    "timestamp",
    "symbol",
    col("zscore_data.window").alias("window"),
    col("zscore_data.zscore_price").alias("zscore_price")
)
```



5.4. Ghi dữ liệu vào MongoDB

Dữ liệu được ghi vào từng collection tương ứng với time window:

```
for window in ["30s", "1m", "5m", "15m", "30m", "1h"]:
    collection_name = get_collection_name(window)
    window_df = exploded_df.filter(col("window") == window)

    window_df.write \
        .format("com.mongodb.spark.sql") \
        .mode("append") \
        .option("uri", mongo_uri) \
        .option("database", database) \
        .option("collection", collection_name) \
        .save()
```

6. Xử lý lỗi và khả năng chịu lỗi

6.1. Watermarking

Hệ thống sử dụng watermarking để xử lý dữ liệu đến muộn với tolerance 10 giây:

```
parsed_df = parsed_df.withWatermark("timestamp", "10 seconds")
```

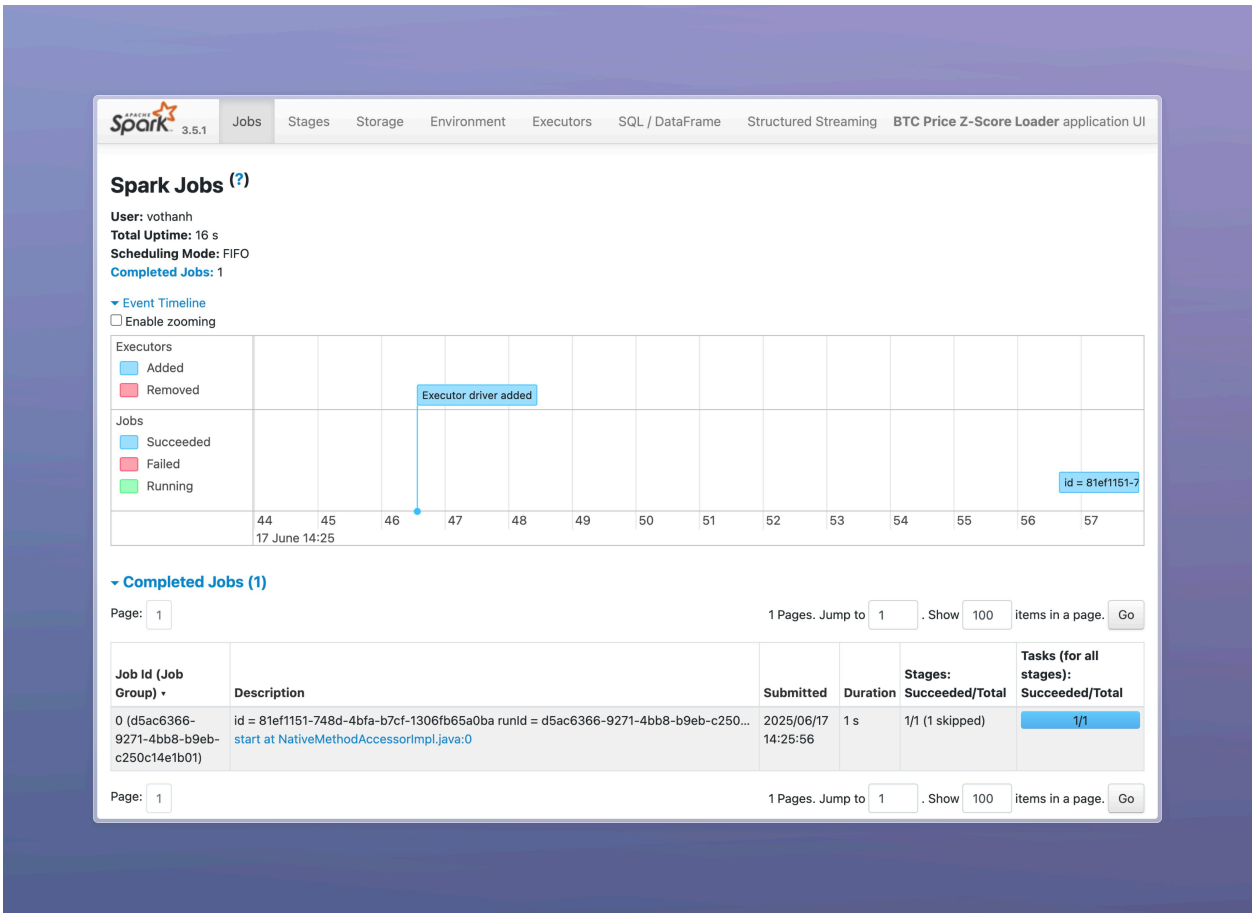
6.2. Checkpoint

Spark Streaming sử dụng checkpoint để đảm bảo fault tolerance:

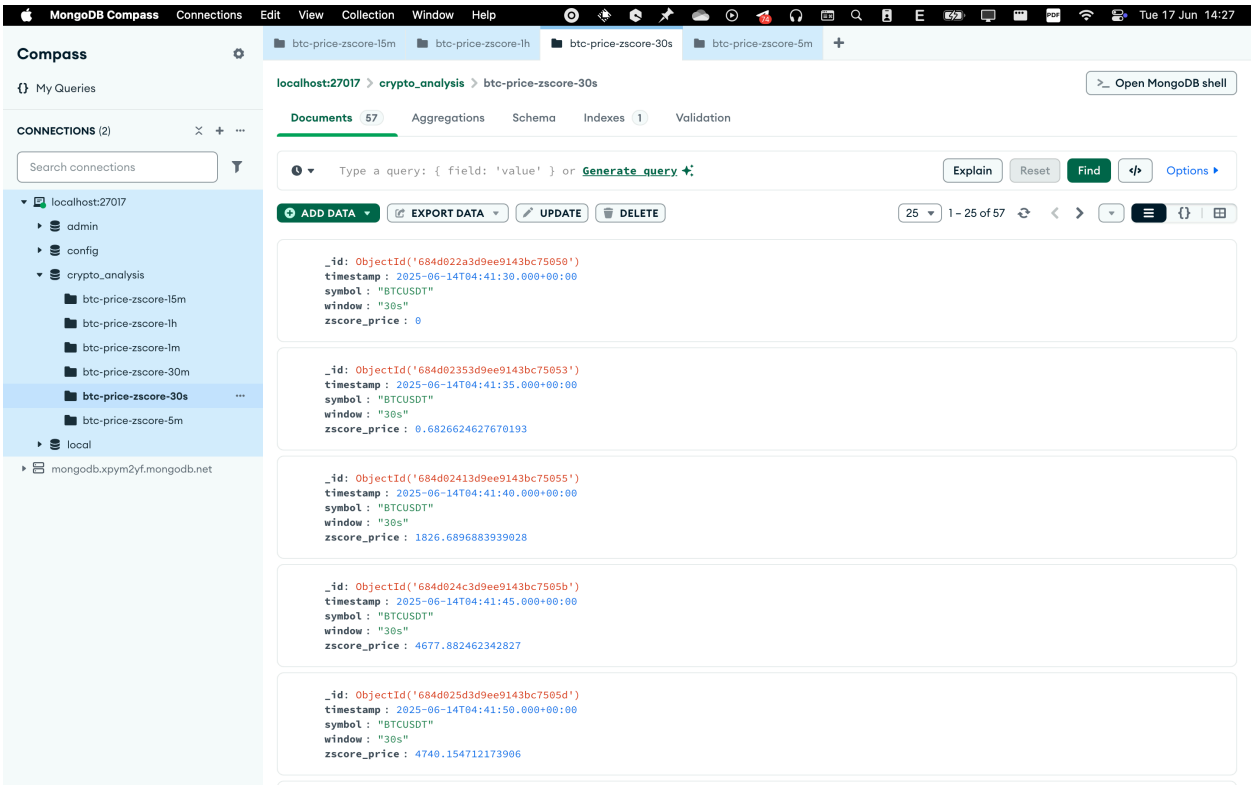
```
spark.sql.streaming.checkpointLocation: "/tmp/checkpoint"
spark.sql.streaming.forceDeleteTempCheckpointLocation: "true"
```

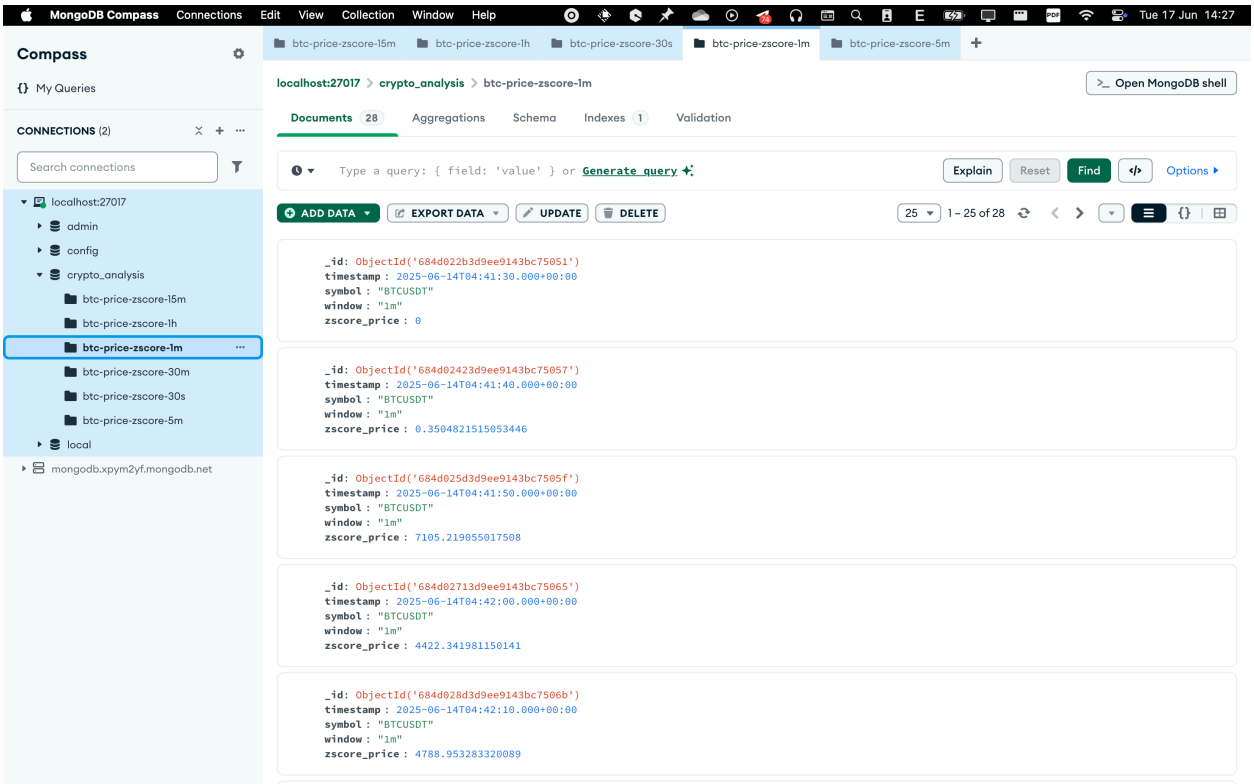
7. Kết quả

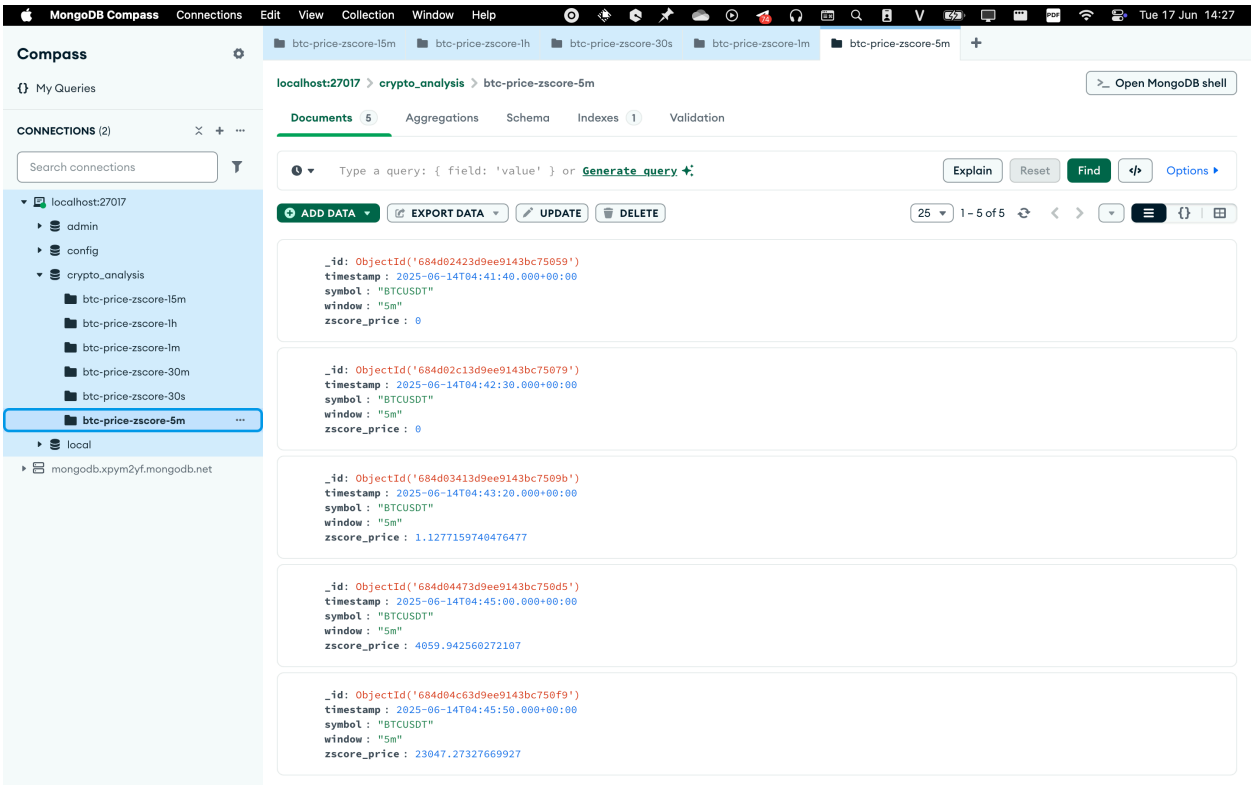
Sau khi chạy tiến trình `load/main.py`, ứng dụng Spark Structured Streaming sẽ khởi chạy, lắng nghe dữ liệu từ topic `btc-price-zscore` và ghi vào MongoDB.

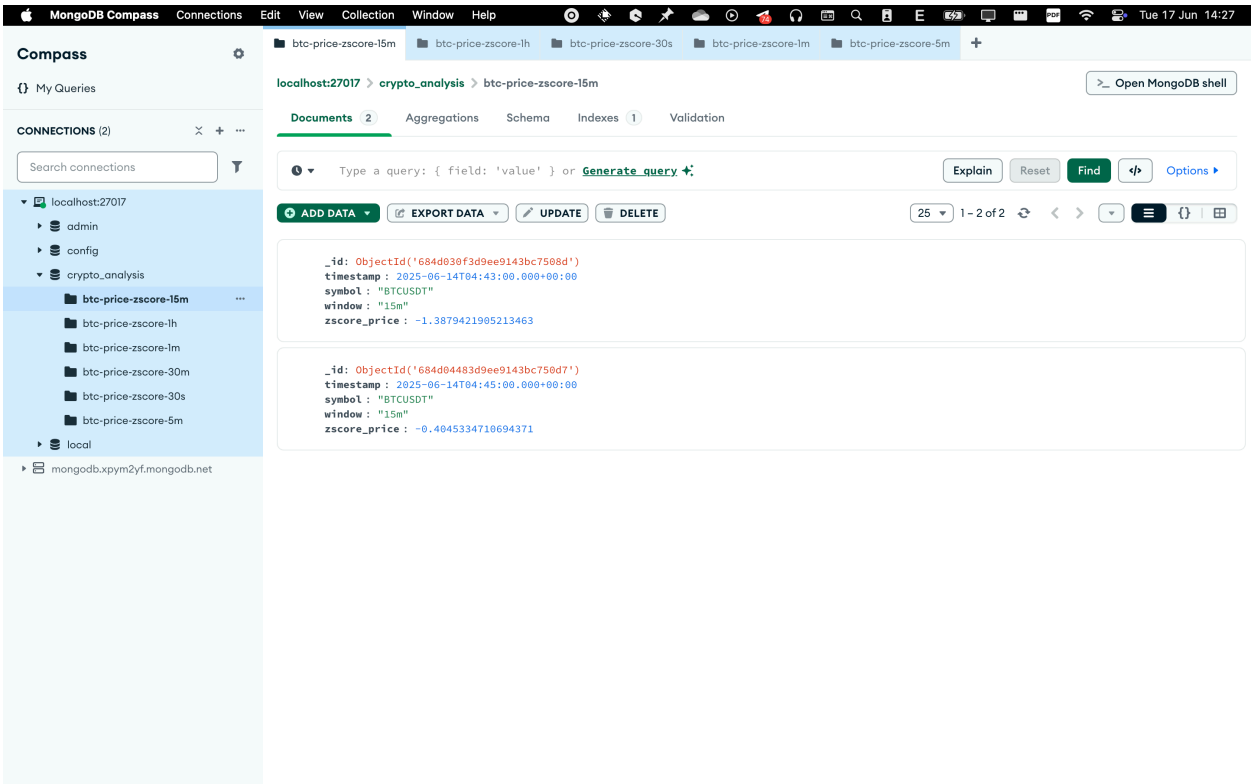


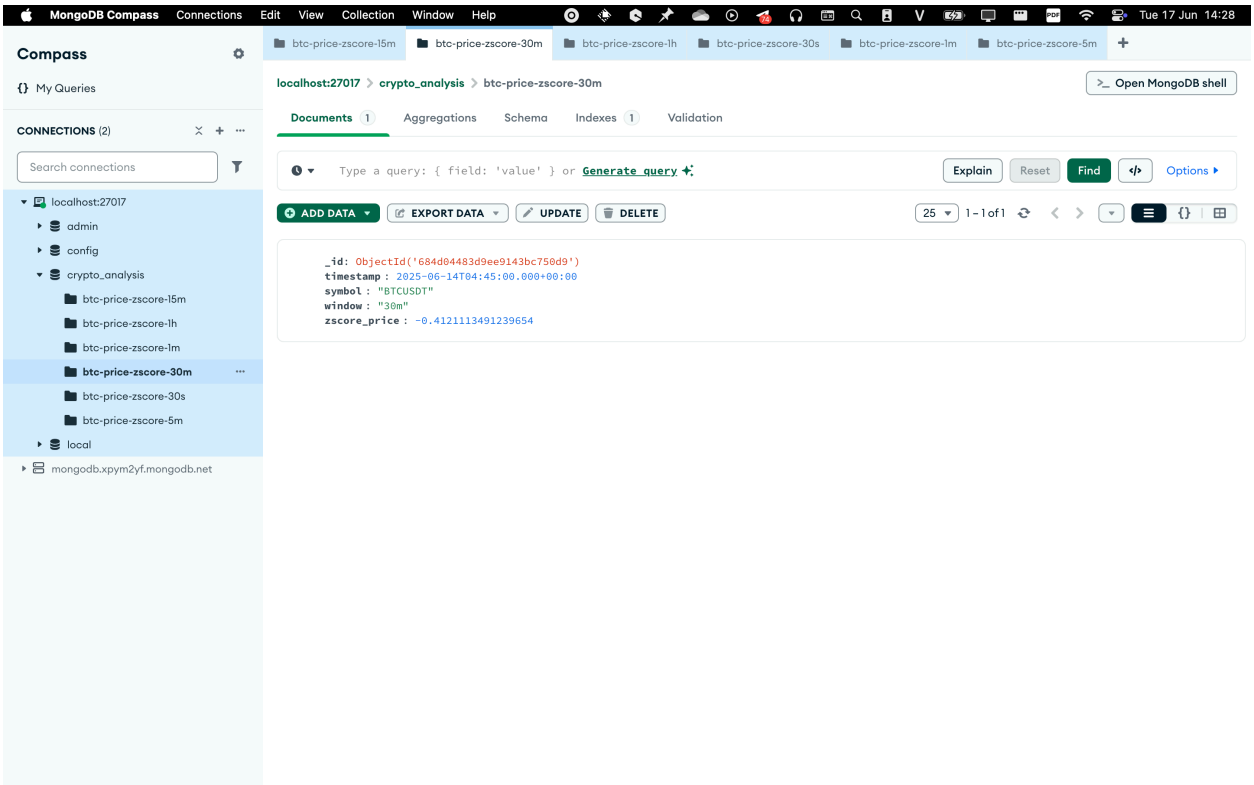

```
        :: modules in use:
        com.google.code.findbugs#jsr305;3.0.0 from central in [default]
        commons-logging#commons-logging;1.1.3 from central in [default]
        org.apache.commons#commons-pool2;2.6.2 from central in [default]
        org.apache.hadoop#hadoop-client-api;3.3.1 from central in [default]
        org.apache.hadoop#hadoop-client-runtime;3.3.1 from central in [default]
        org.apache.htrace#htrace-core4;4.1.0-incubating from central in [default]
        org.apache.kafka#kafka-clients;2.8.0 from central in [default]
        org.apache.spark#spark-sql-kafka-0-10_2.12;3.2.0 from central in [default]
        org.apache.spark#spark-token-provider-kafka-0-10_2.12;3.2.0 from central in [default]
        org.lz4#lz4-java;1.7.1 from central in [default]
        org.mongodb#bson;4.0.5 from central in [default]
        org.mongodb#mongodb-driver-core;4.0.5 from central in [default]
        org.mongodb#mongodb-driver-sync;4.0.5 from central in [default]
        org.mongodb.spark#mongo-spark-connector_2.12;3.0.2 from central in [default]
        org.slf4j#slf4j-api;1.7.30 from central in [default]
        org.spark-project.spark#unused;1.0.0 from central in [default]
        org.xerial.snappy#snappy-java;1.1.8.4 from central in [default]
        ---------------------------------------------------------------------
        |                  |            modules            ||   artifacts   |
        |       conf       | number| search|dwnlded|evicted|| number|dwnlded|
        ---------------------------------------------------------------------
        |      default     |   17  |   0   |   0   |   0   ||   17  |   0   |
        ---------------------------------------------------------------------
:: retrieving :: org.apache.spark#spark-submit-parent-d45d80c0-47fc-4ff9-84ab-e1b41c5de949
        confs: [default]
        0 artifacts copied, 17 already retrieved (0kB/5ms)
```


MongoDB Compass Connections Edit View Collection Window Help
Tue 17 Jun 14:27
Compass
My Queries
CONNECTIONS (2)
Search connections
localhost:27017
admin
config
crypto_analysis
btc-price-zscore-15m
btc-price-zscore-1h
btc-price-zscore-1m
btc-price-zscore-30m
btc-price-zscore-30s
btc-price-zscore-5m
local
mongodb.xpym2yf.mongodb.net
btc-price-zscore-15m
btc-price-zscore-1h
btc-price-zscore-30s
btc-price-zscore-5m
localhost:27017 > crypto_analysis > btc-price-zscore-30s
Open MongoDB shell
Documents 57
Aggregations
Schema
Indexes 1
Validation
Type a query: { field: 'value' } or Generate query
Explain
Reset
Find
Options
ADD DATA
EXPORT DATA
UPDATE
DELETE
25
1 - 25 of 57
_id: ObjectId('684d022a3d9ee9143bc75050')
timestamp : 2025-06-14T04:41:30.000+00:00
symbol : "BTCUSDT"
window : "30s"
zscore_price : 0
_id: ObjectId('684d02353d9ee9143bc75053')
timestamp : 2025-06-14T04:41:35.000+00:00
symbol : "BTCUSDT"
window : "30s"
zscore_price : 0.6826624627670193
_id: ObjectId('684d02413d9ee9143bc75055')
timestamp : 2025-06-14T04:41:40.000+00:00
symbol : "BTCUSDT"
window : "30s"
zscore_price : 1826.6896883939028
_id: ObjectId('684d024c3d9ee9143bc7505b')
timestamp : 2025-06-14T04:41:45.000+00:00
symbol : "BTCUSDT"
window : "30s"
zscore_price : 4677.882462342827
_id: ObjectId('684d025d3d9ee9143bc7505d')
timestamp : 2025-06-14T04:41:50.000+00:00
symbol : "BTCUSDT"
window : "30s"
zscore_price : 4740.154712173906


MongoDB Compass Connections Edit View Collection Window Help
Tue 17 Jun 14:27
Compass
My Queries
CONNECTIONS (2)
Search connections
localhost:27017
admin
config
crypto_analysis
btc-price-zscore-15m
btc-price-zscore-1h
btc-price-zscore-1m
btc-price-zscore-30m
btc-price-zscore-30s
btc-price-zscore-5m
local
mongodb.xpym2yf.mongodb.net
btc-price-zscore-15m
btc-price-zscore-1h
btc-price-zscore-30s
btc-price-zscore-1m
btc-price-zscore-5m
localhost:27017 > crypto_analysis > btc-price-zscore-1m
Open MongoDB shell
Documents 28
Aggregations
Schema
Indexes 1
Validation
Type a query: { field: 'value' } or Generate query
Explain
Reset
Find
Options
ADD DATA
EXPORT DATA
UPDATE
DELETE
25
1 - 25 of 28
_id: ObjectId('684d022b3d9ee9143bc75051')
timestamp : 2025-06-14T04:41:30.000+00:00
symbol : "BTCUSDT"
window : "1m"
zscore_price : 0
_id: ObjectId('684d02423d9ee9143bc75057')
timestamp : 2025-06-14T04:41:40.000+00:00
symbol : "BTCUSDT"
window : "1m"
zscore_price : 0.3504821515053446
_id: ObjectId('684d025d3d9ee9143bc7505f')
timestamp : 2025-06-14T04:41:50.000+00:00
symbol : "BTCUSDT"
window : "1m"
zscore_price : 7105.219055017508
_id: ObjectId('684d02713d9ee9143bc75065')
timestamp : 2025-06-14T04:42:00.000+00:00
symbol : "BTCUSDT"
window : "1m"
zscore_price : 4422.341981150141
_id: ObjectId('684d028d3d9ee9143bc7506b')
timestamp : 2025-06-14T04:42:10.000+00:00
symbol : "BTCUSDT"
window : "1m"
zscore_price : 4788.953283320089


MongoDB Compass Connections Edit View Collection Window Help
Tue 17 Jun 14:27
Compass
My Queries
CONNECTIONS (2)
Search connections
localhost:27017
admin
config
crypto_analysis
btc-price-zscore-15m
btc-price-zscore-1h
btc-price-zscore-1m
btc-price-zscore-30m
btc-price-zscore-30s
btc-price-zscore-5m
local
mongodb.xpym2yf.mongodb.net
btc-price-zscore-15m
btc-price-zscore-1h
btc-price-zscore-30s
btc-price-zscore-1m
btc-price-zscore-5m
localhost:27017 > crypto_analysis > btc-price-zscore-5m
Open MongoDB shell
Documents 5
Aggregations
Schema
Indexes 1
Validation
Type a query: { field: 'value' } or Generate query
Explain
Reset
Find
Options
ADD DATA
EXPORT DATA
UPDATE
DELETE
25
1 - 5 of 5
_id: ObjectId('684d02423d9ee9143bc75059')
timestamp : 2025-06-14T04:41:40.000+00:00
symbol : "BTCUSDT"
window : "5m"
zscore_price : 0
_id: ObjectId('684d02c13d9ee9143bc75079')
timestamp : 2025-06-14T04:42:30.000+00:00
symbol : "BTCUSDT"
window : "5m"
zscore_price : 0
_id: ObjectId('684d03413d9ee9143bc7509b')
timestamp : 2025-06-14T04:43:20.000+00:00
symbol : "BTCUSDT"
window : "5m"
zscore_price : 1.1277159740476477
_id: ObjectId('684d04473d9ee9143bc750d5')
timestamp : 2025-06-14T04:45:00.000+00:00
symbol : "BTCUSDT"
window : "5m"
zscore_price : 4059.942560272107
_id: ObjectId('684d04c63d9ee9143bc750f9')
timestamp : 2025-06-14T04:45:50.000+00:00
symbol : "BTCUSDT"
window : "5m"
zscore_price : 23047.27327669927


MongoDB Compass Connections Edit View Collection Window Help
Tue 17 Jun 14:27
Compass
My Queries
CONNECTIONS (2)
Search connections
localhost:27017
admin
config
crypto_analysis
btc-price-zscore-15m
btc-price-zscore-1h
btc-price-zscore-1m
btc-price-zscore-30m
btc-price-zscore-30s
btc-price-zscore-5m
local
mongodb.xpym2yf.mongodb.net
btc-price-zscore-15m
btc-price-zscore-1h
btc-price-zscore-30s
btc-price-zscore-1m
btc-price-zscore-5m
localhost:27017 > crypto_analysis > btc-price-zscore-15m
Open MongoDB shell
Documents 2
Aggregations
Schema
Indexes 1
Validation
Type a query: { field: 'value' } or Generate query
Explain
Reset
Find
Options
ADD DATA
EXPORT DATA
UPDATE
DELETE
25
1 - 2 of 2
_id: ObjectId('684d030f3d9ee9143bc7508d')
timestamp : 2025-06-14T04:43:00.000+00:00
symbol : "BTCUSDT"
window : "15m"
zscore_price : -1.3879421905213463
_id: ObjectId('684d04483d9ee9143bc750d7')
timestamp : 2025-06-14T04:45:00.000+00:00
symbol : "BTCUSDT"
window : "15m"
zscore_price : -0.4045334710694371


MongoDB Compass
Connections
Edit
View
Collection
Window
Help
Tue 17 Jun 14:28
Compass
My Queries
CONNECTIONS (2)
Search connections
localhost:27017
admin
config
crypto_analysis
btc-price-zscore-15m
btc-price-zscore-1h
btc-price-zscore-1m
btc-price-zscore-30m
btc-price-zscore-30s
btc-price-zscore-5m
local
mongodb.xpym2yf.mongodb.net
btc-price-zscore-15m
btc-price-zscore-30m
btc-price-zscore-1h
btc-price-zscore-30s
btc-price-zscore-1m
btc-price-zscore-5m
localhost:27017 > crypto_analysis > btc-price-zscore-30m
Open MongoDB shell
Documents 1
Aggregations
Schema
Indexes 1
Validation
Type a query: { field: 'value' } or Generate query
Explain
Reset
Find
Options
ADD DATA
EXPORT DATA
UPDATE
DELETE
25
1-1 of 1
_id: ObjectId('684d04483d9ee9143bc750d9')
timestamp : 2025-06-14T04:45:00.000+00:00
symbol : "BTCUSDT"
window : "30m"
zscore_price : -0.4121113491239654