Politechnika Warszawska

Crypto Options vs. Rates

Analiza Korelacji: Rynki Predykcyjne (Polymarket) vs. Ceny Spot (Binance)

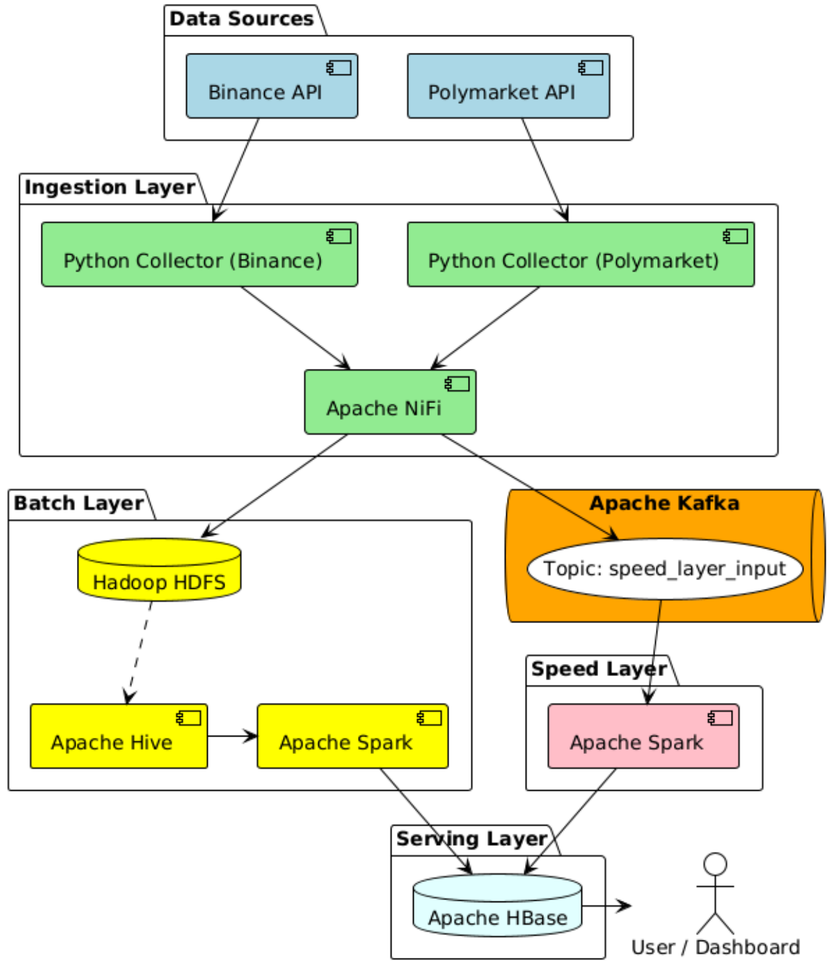
Jan Skwarek, Sebastian Pergała, Aleksander Malinowski

14.01.2026

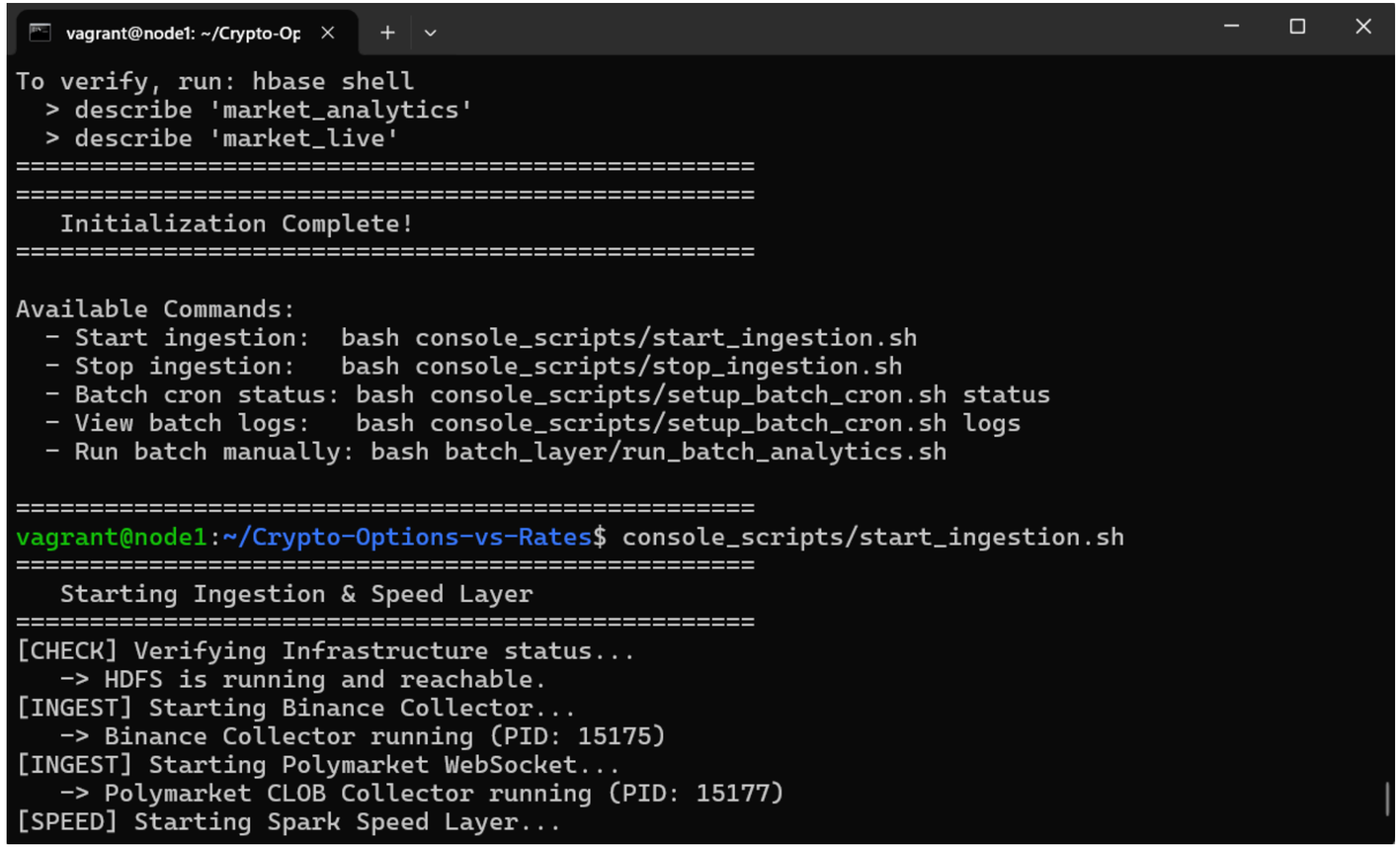
Analiza: Sentyment vs. Rzeczywistość

Cel: Badanie korelacji między zakładami (Polymarket) a kursem kryptowalut (Binance)
Źródła: Dane z rynków predykcyjnych oraz rzeczywiste ceny spot.
Podejście: Wykorzystanie "mądrości tłumu" do zrozumienia ruchów cenowych.
Technologia: Skalowalne przetwarzanie w architekturze Lambda.

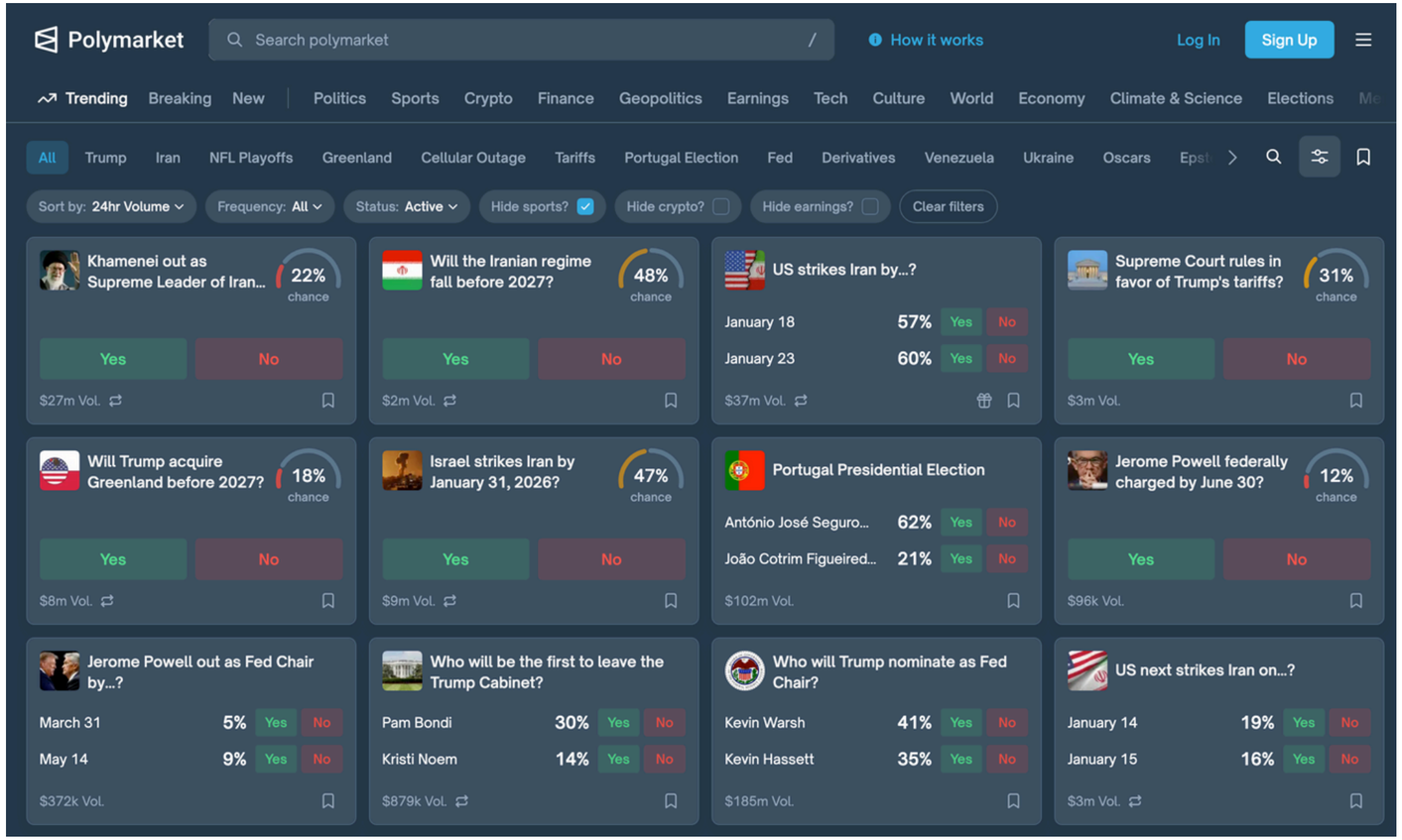
Architektura systemu



Środowisko i automatyzacja



Źródła danych



Źródło danych - Binance

- **Backfill: Binance API → JSON → Parquet (HDFS) - zaciągamy automatycznie 1-minutowe świece (klines) z ostatnich 24h, żeby mieć dane do analityki już na starcie (zapobiegamy ‘cold start’)**
- **Stream: WebSocket → Bufor → Parquet (Live) - połączenie przez WebSocket dla bieżących aktualizacji świec co minutę.**

Standaryzacja danych

- Rejestrujemy pełne dane OHLCV
- Składujemy pliki Parquet, do których dane są zapisywane z partycjonowaniem po dacie i symbolu waluty, żeby optymalizować odczyt.

Niezawodność

- Mechanizm self-healing dla połączeń
- Deduplikacja danych przed zapisem do strefy Raw w HDFS

Źródło danych - Polymarket

Polymarket

Charakterystyka danych

Typ: Zdecentralizowany rynek predykcyjny.
Zakres: Krótkoterminowe rynki 15-minutowe (opcje binarne góra/dół).
Aktywa: ETH, SOL, XRP oraz BNB.
Wiarygodność: Wyniki rozliczane przez wyrocznie Chainlink.

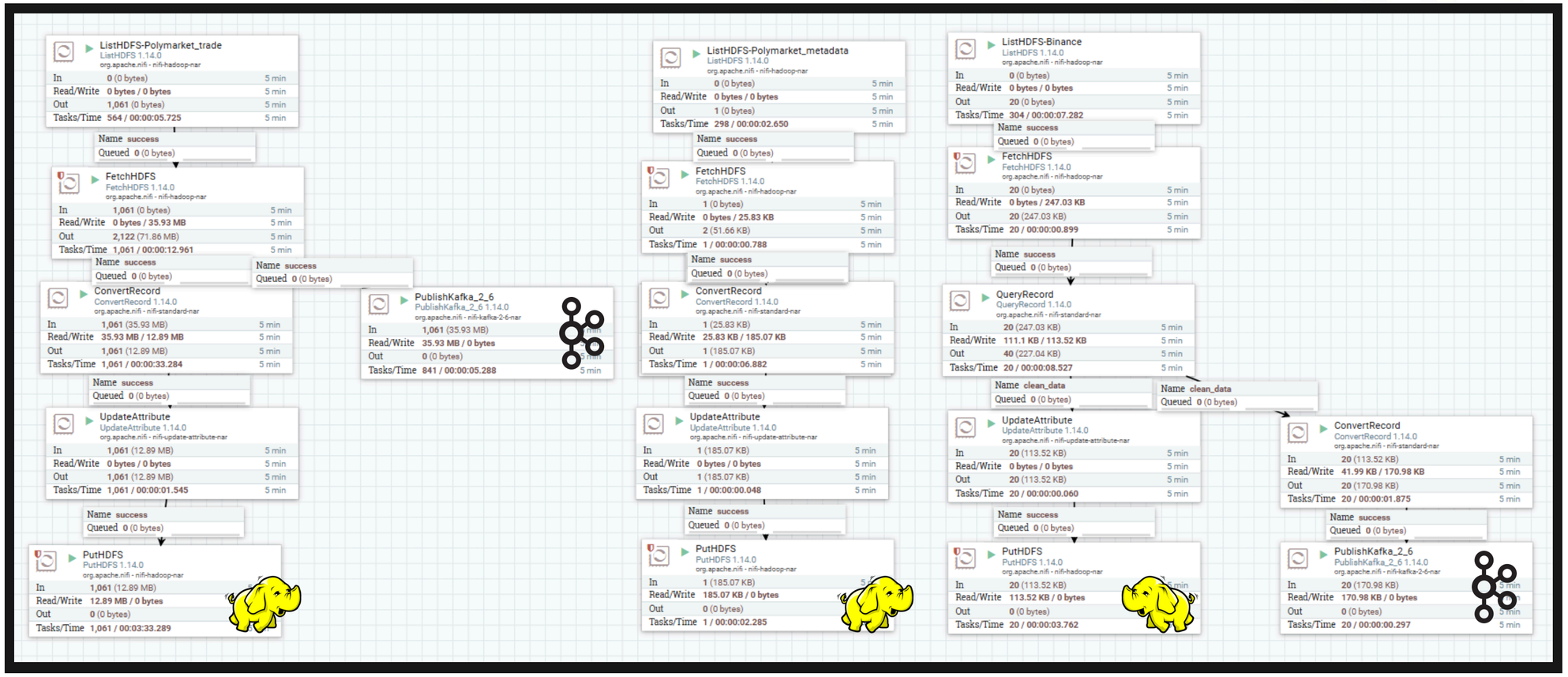
Metoda akwizycji

Gamma API (REST): Dynamiczne odkrywanie identyfikatorów aktywnych rynków.
CLOB API (WebSocket): Odczyt strumienia zdarzeń w czasie rzeczywistym.
Przetwarzane zdarzenia: Arkusz zleceń (book), zmiany cen oraz ostatnie transakcje.

Response


```
{
    "market": "0x5f65177b394277fd294cd75650044e32ba009a95022d88a0c1d565897d72f8f1",
    "price_changes": [
        {
            "asset_id": "71321045679252212594626385532706912750332728571942532289631
            "price": "0.5",
            "size": "200",
            "side": "BUY",
            "hash": "56621a121a47ed9333273e21c83b660cff37ae50",
            "best_bid": "0.5",
            "best_ask": "1"
        },
        {
            "asset_id": "52114319501245915516055106046884209969926127482827954674441
            "price": "0.5",
            "size": "200",
            "side": "SELL",
            "hash": "1895759e4df7a796bf4f1c5a5950b748306923e2",
            "best_bid": "0",
            "best_ask": "0.5"
        }
    ],
    "timestamp": "1757908892351",
    "event_type": "price_change"
}
```

Ingestion Layer - NiFi



Batch Layer - PySpark

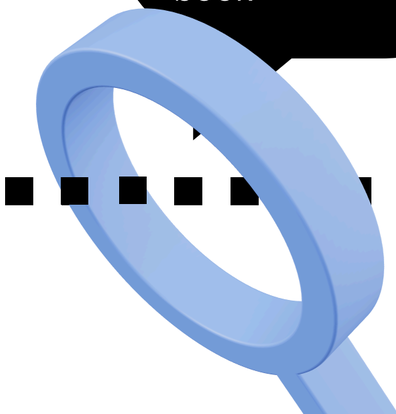
Przetwarzanie historyczne i analityka

- **Technologie:** Apache Spark (PySpark) oraz Apache Hive.
- **Harmonogram:** Cykliczne uruchamianie co 6 godzin.
- **Główne operacje:**
  - Agregacja danych do spójnych okien 15-minutowych.
  - Łączenie (Join) danych giełdowych z przewidywaniami rynku.
  - Obliczanie korelacji i klasyfikacja trafności sygnałów.
- **Wynik:** Pre-komputowane widoki zapisywane w HBase.

Speed Layer - Kafka

binance

Event type:
- last_trade_price
- price_change
- book



polymarket_trade

Speed Layer - Spark

binance

Binance

Column	Type
window	struct
symbol	string
current_price	double
current_sentiment	double
avg_price	double
min_price	double
max_price	double
volatility	double
total_usdt_volume	double
avg_sentiment	double
ticks	long

Polymarket

Column	Type
window	struct
market_id	string
current_prob	double
current_spread	double
current_imbalance	double
avg_prob	double
min_prob	double
max_prob	double
avg_imbalance	double
avg_spread	double
num_updates	long

book

polymarket_trade

last_trade_price

Column	Type
window	struct
market_id	string
total_shares_traded	double
num_trades	long

Serving Layer HBase - integracja architektury Lambda

Schemat Tabeli

- **market_analytics** (Warstwa Batch): Przechowuje w pełni przetworzone, zagregowane okna 15-minutowe
- **market_live** (Warstwa Speed): Przechowuje sub-sekundowe aktualizacje strumieniowe ze Spark Streaming.
- **Strategia RowKey**: Zastosowaliśmy odwrócone znaczniki czasu (Reverse Timestamps).
- **Widok Scalony** (Merge View): Mechanizm łączący historyczną dokładność (Batch) z bieżącą aktualnością (Speed) w spójną odpowiedź API.

cel →

Wymuszenie fizycznego składowania najnowszych danych na początku tabeli, co umożliwia natychmiastowe pobieranie ich przez analitykę i/lub dashboardy.

Analityka i Walidacja Rozwiązania

Logika etykietowania

- Korelacja Sentymentu Rynku (kursy zakładów) z Rzeczywistością (kurs giełdowy) w oknach 15-minutowych.
- Klasyfikacja wyników na: CORRECT_BULL (trafny wzrost), FAILED_BEAR (nietrafiony spadek), UNCERTAIN (niepewny).

Manualna Walidacja

- Przeprowadzono ręczną weryfikację konkretnych okien czasowych (np. 12:00-12:15).
- Potwierdzono, że okna z wysokim pstwem (>60%) zostały poprawnie oflagowane przez Spark Batch w momentach, gdy cena rynkowa podążyła za predykcją.

Automatyzacja i Testowanie

- Testy E2E: Zweryfikowano pełny przepływ danych: WebSocket Binance → HDFS -> Spark -> HBase.
- Integralność Danych: Osiągnięto pełną spójność rekordów między źródłem (Hive) a celem (HBase) podczas testów obciążeniowych.
- Unit testing: Każdy z modułów kodu został pokryty testami jednostkowymi, odizolowanymi od środowiska, na którym działa program.

Dziękujemy za uwagę