Building a Time-Series Database for Financial Market Data

QuestDB

Nicolas Hourcard, co-founder/ceo

nic@questdb.io

Background

Team with decades of experience building low-latency trading software and large-scale databases

Fastest growing open source time-series database   ★ Starred 14.4k   Contributors 146

Raised funding from top US Venture Capital firms who previously invested in: *Redis Labs, GitLab, Docker, Airbnb, Stripe, Dropbox, Cruise, Mulesoft*

Database Popularity (DB-Engines.com)

Oct 2024	Oct 2023	DBMS	Database Model	Oct 2024	Oct 2023
1.	1.	InfluxDB ➕	Time Series, Multi-model ℹ️	22.39	-7.35
2.	2.	Kdb ➕	Multi-model ℹ️	7.72	-0.67
3.	3.	Prometheus	Time Series	7.31	+0.03
4.	⬆️5.	Graphite	Time Series	5.00	-0.22
5.	⬇️4.	TimescaleDB	Time Series, Multi-model ℹ️	3.79	-1.58
6.	⬆️10.	QuestDB	Time Series, Multi-model ℹ️	2.83	+0.68
7.	7.	Apache Druid	Multi-model ℹ️	2.79	-0.31
8.	⬇️6.	DolphinDB	Multi-model ℹ️	2.67	-1.27

← QuestDB growing meaningfully YoY

Source: https://db-engines.com/en/ranking/time+series+dbms

Thousands of users

↵ in Financial Services

Our focus: financial markets
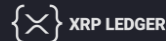
Global banks	MIZUHO	Laser Digital NOMURA Group	HDFC BANK	TD
Hedge funds	BREVAN HOWARD	WORLDQUANT	WELWING CAPITAL GROUP	
Crypto	OKX	HiddenRoad	ONE TRADING	XRP LEDGER
Financial exchanges	[B]³ BRAZILIAN EXCHANGE AND OTC	aquis stock exchange		
Central banks	Banco de la República We are the Central Bank of Colombia			
Data intelligence	S&P Global	Beeks		

Aerospace	AIRBUS
Space	isar aerospace
Telco	airtel
Technology	yahoo!
Energy	BOSCH

QuestDB under the hood

    ↰ Built from scratch for performance

Built from scratch for market data

Ideal data layout for fast queries:
Columnar store & time partitioned

Relational model: Tables, ASOF JOINs, SQL

Heavy parallelism: SIMD instructions, SQL JIT
Compiler

trades			
ts	price	symbol	qty
ts01	178.08	JPM	1000
ts02	148.66	V	400
ts03	424.86	JNJ	5000
ts04	178.09	AMD	100
ts05	505.08	BAC	2500
ts06	394.14	KO	2000
…	…	…	…

Open formats and tiered storage

Streaming

Columnar
format

Data
X hours old

Parquet

Object store / NAS

Streaming on the way in
5M rows/sec/server

Open format storage on
the way out

Benefits of QuestDB

No vendor lock-in: Open formats (Parquet & Arrow), open source

Lower TCO: Easy access via SQL, cost-efficient data retention, low Maintenance

Performance for market data: Fast ingestion, millisecond queries, efficient handling of order book data

Lightning-fast queries

↰ with SQL

SAMPLE BY

```sql
SELECT
    timestamp,
    vwap(price, amount) AS vwap_price
FROM trades
WHERE symbol = 'BTC-USD'
SAMPLE BY 15m
```

| timestamp | vwap_price |
timestamp	double
2024-04-17T14:00:00.000000Z	62160.61020321509
2024-04-17T14:15:00.000000Z	61458.34099643154
2024-04-17T14:30:00.000000Z	61216.872183932705
2024-04-17T14:45:00.000000Z	61505.64346573607
2024-04-17T15:00:00.000000Z	60970.16210306167
2024-04-17T15:15:00.000000Z	60800.45658818344
2024-04-17T15:30:00.000000Z	60215.96241438603
2024-04-17T15:45:00.000000Z	60391.896294865386
2024-04-17T16:00:00.000000Z	60284.43016018901
2024-04-17T16:15:00.000000Z	60686.007354288726
2024-04-17T16:30:00.000000Z	60366.15824438131
2024-04-17T16:45:00.000000Z	60170.87155928168

LATEST ON

```
select * from
  trades
latest on
  timestamp
partition by
  symbol, side
```

| symbol | side | price | amount | timestamp |
symbol	symbol	double	double	timestamp
ETH-DAI	buy	3444.96	0.00741366	2024-03-22T10:38:28.948385Z
ETH-DAI	sell	3437.17	0.0014	2024-03-22T10:51:29.994074Z
ADA-BTC	sell	0.0000096	80.78	2024-03-22T10:54:04.689191Z
MATIC-BTC	buy	0.00001522	119.7	2024-03-22T10:55:40.118164Z
ADA-BTC	buy	0.00000961	686	2024-03-22T10:55:44.238375Z
ADA-USDT	sell	0.62	139.07	2024-03-22T10:56:03.979027Z
MATIC-BTC	sell	0.00001522	498.4	2024-03-22T10:56:04.176845Z
UNI-BTC	sell	0.0001827	100	2024-03-22T10:56:38.972749Z
UNI-BTC	buy	0.0001827	93.1	2024-03-22T10:56:40.703348Z
XLM-BTC	buy	0.00000202	811	2024-03-22T10:57:45.555952Z

Order book functions

```
select timestamp, avg(l2price(
  5000,
  ask_sz_00, ask_px_00,
  [...]
  ask_sz_09, ask_px_09
) - ask_px_00)
from AAPL_orderbook
where timestamp IN '2023-08-25T13:30:00;1h'
sample by 10m;
```

timestamp	avg
timestamp	double
2023-08-25T13:30:00.000000Z	0.0409174107836
2023-08-25T13:40:00.000000Z	0.0416735977974
2023-08-25T13:50:00.000000Z	0.0400370085279
2023-08-25T14:00:00.000000Z	0.045853886332
2023-08-25T14:10:00.000000Z	0.0424633881882
2023-08-25T14:20:00.000000Z	0.0443940941203

Execution price for a buy order of a given size (5k shares) minus best ask price for a given time range

＋ Example queries

▶ Run ⌘ ↵   Search docs ⌘ K

Tables ⚙

Filter...

H AAPL_orderbook
D ethblocks_json
D trades
M trips
weather

SQL ＋

History

```sql
/* This query finds each buy event, and joins with the most
   recent sell event for the same symbol.
   Note that we are filtering buys since 2024 and sells since
   Dec 31st 2023. This is to make sure we always have a previous
   sell row for each buy event */

WITH trade_buys AS (
SELECT timestamp, symbol, price, side
FROM trades
WHERE side = 'buy' AND timestamp IN '2024'
), trade_sells AS (
        SELECT timestamp, symbol, price, side
FROM trades
WHERE side = 'sell' AND timestamp > '2023-12-31'
)
SELECT * from trade_buys ASOF JOIN trade_sells ON (symbol);
```

Log   [4:38:01 PM GMT+01:00] ✓ 239,442,009 rows in 384ms **Execute: 281.98ms Network: 102.02ms Total: 384ms**   Count: 250.61ms Authentication: 680ns Compile: 2ms

239,442,009 rows

timestamp (timestamp)	symbol (symbol)	price (double)	side (symbol)	timestamp1 (timestamp)	symbol1 (symbol)	price1 (double)	side1 (symbol)
2024-01-01T00:00:00.314299Z	AVAX-USD	38.56	buy	2023-12-31T23:59:46.690587Z	AVAX-USD	38.51	sell
2024-01-01T00:00:00.314299Z	AVAX-USD	38.56	buy	2023-12-31T23:59:46.690587Z	AVAX-USD	38.51	sell
2024-01-01T00:00:00.315867Z	AVAX-USD	38.56	buy	2023-12-31T23:59:46.690587Z	AVAX-USD	38.51	sell
2024-01-01T00:00:00.315867Z	AVAX-USD	38.56	buy	2023-12-31T23:59:46.690587Z	AVAX-USD	38.51	sell

Copyright © 2024 QuestDB   ● Connected QuestDB 8.1.2

https://demo.questdb.io/   https://github.com/questdb