

Kx Streaming Analytics –



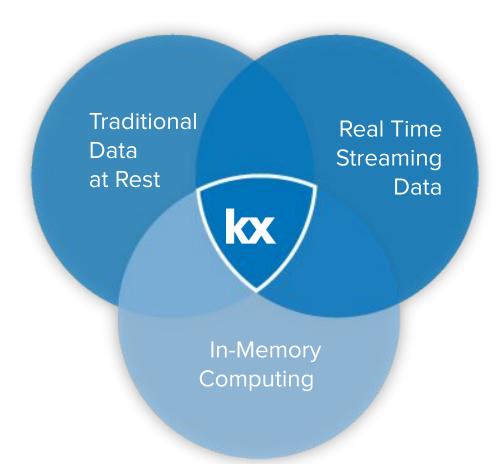
Ferenc Bodon Senior Architect, Kx

fbodon@kx.com









"Most of our clients are extremely aware of the fact that they need a really solid data architecture – part of that data architecture will absolutely require access to in-memory technology, and that's where our customers see us fitting in."

Gerry Buggy Chief Strategy Officer, Kx

Solid foundations – kdb+ and q



Simplicity and Elegance without compromising Performance



def my_agg(x):

data = {'NR': x.bucket.count(),

'TOTAL_QTY': x.qty.sum(),

'AVG_QTY': x.qty.sum(),

'AVG_RISK': x.risk'

'AVG_RISK': x.risk'

'AVG_RISK': x.risk'

'AVG_OTY: r

AVG_OTY: r

AVG_OTY: r

AVG_OTY: avg qty,

AVG_OTY: avg qty,

AVG_OTY: avg qty,

def executeQueryApply(t):

return t.groupby('bucket').apply(my_agg,

meta={'NR': 'i8', 'TOTAL_QTY': 'i8', 'AVG_QTY': 'f8', 'TOTAL_AVG',

'W_AVG_OTY': 'f8', 'W_AVG_RISK': 'f8'}).compute().astype({

'NR': 'int64', 'TOTAL_QTY': 'int64', 'TOTAL_RISK': 'int64'})

- Vector programming
- Roots in mathematics. Lists, maps, functions with the same syntax
- Advanced functional programming
- Tables are first class citizens

Inherent parallelisation

q-sql



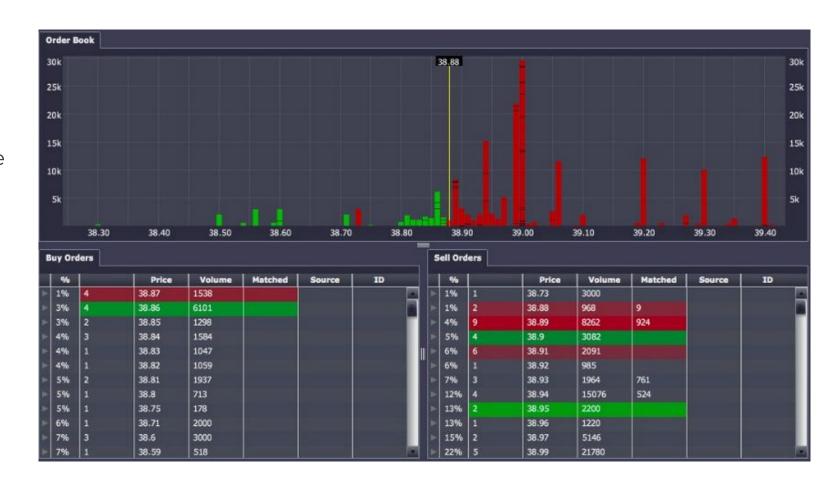
KDnuggets article: https://www.kdnuggets.com/2020/04/python-data-analysis-really-that-simple.html

Business Case: Build the order book in real-time, e.g. Dynamic TOB



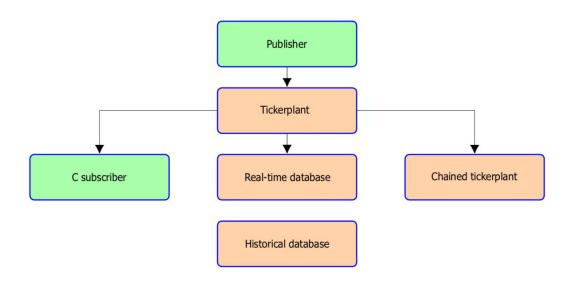
Kdb+ is highly optimized for **vector operations** so the framework uses this strength. For example, instead of re-running the same code multiple times, it will use vector operations. We filter data sources so multiple downstream consumers can see different views on the same quote landscape

This requirement to **aggregate across venues** is also increasingly becoming the case in the equity market, where the large exchanges are losing volume and liquidity is more fragmented across pools and alternative sources



Kdb+ tick plant





Open-source kdb+ tick plant offers streaming, in-memory and on-disk data analysis.

Customers' motivation for ingest benchmark

- Streaming and historical analysis require different code. A simple example: average price by stock.
- Several applications tolerate (sub) millisecond latency but they require flexibility of a query language

Fully integrated technology stack



View. Transform. Share. Your Advantage.

- Connect directly to kdb+ for instant visualization (Dashboards Direct)
- Hardware accelerated rendering for faster display time
- Bespoke components and SDK enable solutions for surveillance, financial charting and more



Instant stream visualization for actionable insights



