AQUAQ Analytics

Kdb+ 4.0

Experts in fast data solutions

for demanding environments

AQUAQ Analytics

Introduction

Kdb+ 4.0 became a production release on 2020.03.17 – formerly known as 3.7t, Kx decided the changes merited a major version increase to 4.0 (plus it makes for a better sounding press release)

Features

- Multithreaded primitives
- Data-At-Rest Encryption (DARE)
- Optane AppDirect support
- Profiler

AQUAQ Analytics

Multithreaded primitives

The following primitives now use multiple threads where appropriate:

```
atomics:    abs acos and asin atan ceiling cos div exp floor
            log mod neg not null or reciprocal signum sin sqrt
            tan within xbar xexp xlog + - * % & | < > = >= <= <>
aggregate: all any avg cor cov dev max min scov sdev sum svar var wavg
lookups*:  ?(Find) aj asof bin binr ij in lj uj
index:     @(Apply At) select .. where delete
misc:      $(Cast) #(Take) _(Drop/Cut) ,(Join) deltas differ distinct
           next prev sublist til where xprev
           select ... by**
```

AQUAQ Analytics

Multithreaded primitives

```
q)/ check it works with a long vector
q)x:100000000?10f
q)\s 0
q)\t avg x
83
q)\s 4
q)\t avg x
40
q)/ what about shorter lists?
q)x:10000?10f
q)\s 0
q)\t:100000 avg x
154
q)\s 4
q)\t:100000 avg x
156
```

AQUAQ Analytics

Multithreaded primitives

```
q)f:{[r;s]system"s ",string s;floor med system each 3#enlist"t:",string[r]," avg x"}
q)g:{[n;s]x::n?10f;r:1|`long$1e9%n;f[r]each s}
q)s:0 2 4 8
q)l:1e3*prds 6#10
q)show t:([]l)!flip(`$"s",'string s)!flip g[;s] each `long$l
l       | s0  s2  s4  s8
--------| ---------------
10000   | 144 145 144 144
100000  | 148 148 148 148
1000000 | 302 195 120 101
1e+07   | 559 403 197 140
1e+08   | 632 508 250 193
1e+09   | 949 518 272 186
q)`long$100*t%'exec s0 from t
l       | s0  s2  s4  s8
--------| ---------------
10000   | 100 101 100 100
100000  | 100 100 100 100
1000000 | 100 65  40  33
1e+07   | 100 72  35  25
1e+08   | 100 80  40  31
1e+09   | 100 55  29  20
```

AQUAQ Analytics

Multithreaded primitives

- Multithreading only kicks in when vector is >=1m length
- Given lots of kdb queries involve grouping by sym, many typical aggregation queries will not benefit from multithreaded primitives as the lists aren't long enough
- Nesting data by sym and using peach will give better query parallelization in that case – at the cost of breaking the simple flat table structure
- Where clauses on unindexed columns will benefit – e.g. select count i from t where price=max price

AQUAQ Analytics

Data-At-Rest Encryption (DARE)

- Kdb+4.0 supports Data-At-Rest Encryption (DARE), using AES256CBC
- Requires OpenSSL 1.1.1 to generate key
- Intel AES hardware encryption instruction set on modern chips allows for extremely fast encryption and decryption – probably not viable to use DARE performance wise without this
- Encryption and decryption requires both a keyfile and password – these are expected to be kept separately
- Each kdb process can only use 1 key at a time

AQUAQ Analytics

Data-At-Rest Encryption (DARE)

```
jgrant@homer:~$ openssl rand 32 | openssl aes-256-cbc -md SHA256 -salt -pbkdf2 -iter 50000 -out kek1.key
enter aes-256-cbc encryption password:
Verifying - enter aes-256-cbc encryption password:
jgrant@homer:~$ q
KDB+ 4.0 2020.03.30 Copyright (C) 1993-2020 Kx Systems
l64/ 24()core 128387MB jgrant homer 127.0.1.1 EXPIRE 2020.06.30 AquaQ #55345

q)-36!(`:kek1.key;"abc123") / load key with our password
q).z.zd:17 16 6 / set file writes to use aes256cbc encryption automatically
q)\mkdir cryptdb
q)`:cryptdb/trade/ set .Q.en[`:cryptdb]([]time:asc 100?0t;sym:100?`3;price:100?100f);
q)`:cryptdb/quote/ set .Q.en[`:cryptdb]([]time:asc 200?0t;sym:200?`3;bid:200?100f;ask:200?100f);
q)\\
jgrant@homer:~$ q cryptdb/ # can't load encrypted folder directly as hdb
KDB+ 4.0 2020.03.30 Copyright (C) 1993-2020 Kx Systems
l64/ 24()core 128387MB jgrant homer 127.0.1.1 EXPIRE 2020.06.30 AquaQ #55345

'sym. no key loaded for encrypted file sym
jgrant@homer:~$ q
KDB+ 4.0 2020.03.30 Copyright (C) 1993-2020 Kx Systems
l64/ 24()core 128387MB jgrant homer 127.0.1.1 EXPIRE 2020.06.30 AquaQ #55345

q)-36!(`:kek1.key;"abc123") / load key with our password
q)\l cryptdb
q)select max price by sym from trade where sym like "a*"
sym| price
---| --------
aaj| 10.86824
afk| 25.60658
```

AQUAQ Analytics

Data-At-Rest Encryption (DARE)

- DARE extends the compression functionality (-19!, .z.zd) by adding aes256cbc as an algorithm
- It can be combined with compression by adding the encryption algorithm value to the compression algo
- Kx state there is a small additional storage cost with encrypted data

```
q)-36!(`:kek1.key;"abc123")
q)nt:10000000;trade:update`p#sym from`sym xasc([]time:asc .z.d+nt?1D0;sym:nt?`3;price:50+0.01*nt?5000;size:10+10*nt?100)
q)nq:50000000;quote:update`p#sym from`sym xasc([]time:asc
.z.d+nt?1D0;sym:nt?`3;bid:50+0.01*nt?5000;ask:50+0.01*nt?5000;bsize:10+10*nt?10;asize:10+10*nt?10)
q)f:{.z.zd:y;s x;}
q)s:{[db]{(` sv x,y,`) set .Q.en[x]get y}[db] each tables[]}
q)f'[`:db_none`:db_comp`:db_enc`:db_cenc;(17 0 6;17 2 6;17 16 6;17 18 6)];
q)\\
jgrant@homer:~$ du -hs db*
232M    db_cenc
232M    db_comp
764M    db_enc
764M    db_none
```

AQUAQ Analytics

Data-At-Rest Encryption (DARE)

- Performance suffers, but assuming the right hardware support, penalty is less than that of adding compression

```
q)-36!(`:kek1.key;"abc123")
q)\l /home/jgrant/db_none
q)\t:100 select sum size, avg price by sym from trade where sym in `aaa`bbb`ccc`ddd`eee
25
q)\l /home/jgrant/db_comp
q)\t:100 select sum size, avg price by sym from trade where sym in `aaa`bbb`ccc`ddd`eee
542
q)\l /home/jgrant/db_enc
q)\t:100 select sum size, avg price by sym from trade where sym in `aaa`bbb`ccc`ddd`eee
235
q)\l /home/jgrant/db_cenc
q)\t:100 select sum size, avg price by sym from trade where sym in `aaa`bbb`ccc`ddd`eee
577
```

AQUAQ Analytics

Data-At-Rest Encryption (DARE)

Errors

```
q)get`:db_enc/sym / read encrypted data without loading key
'db_enc/sym. no key loaded for encrypted file db_enc/sym
  [0]  get`:db_enc/sym / read unecrypted data without loading key
       ^
q)-36!(`:kek1.key;"abc1234") / wrong password
'Invalid password for kek1.key
  [0]  -36!(`:kek1.key;"abc1234") / wrong password
         ^
q)-36!(`:kek2.key;"123abc")
q)get`:db_enc/sym / wrong key
'db_enc/sym. wrong key or bad HMAC for db_enc/sym
  [0]  get`:db_enc/sym / wrong key
       ^
```

AQUAQ Analytics

Data-At-Rest Encryption (DARE)

Password entry prompt

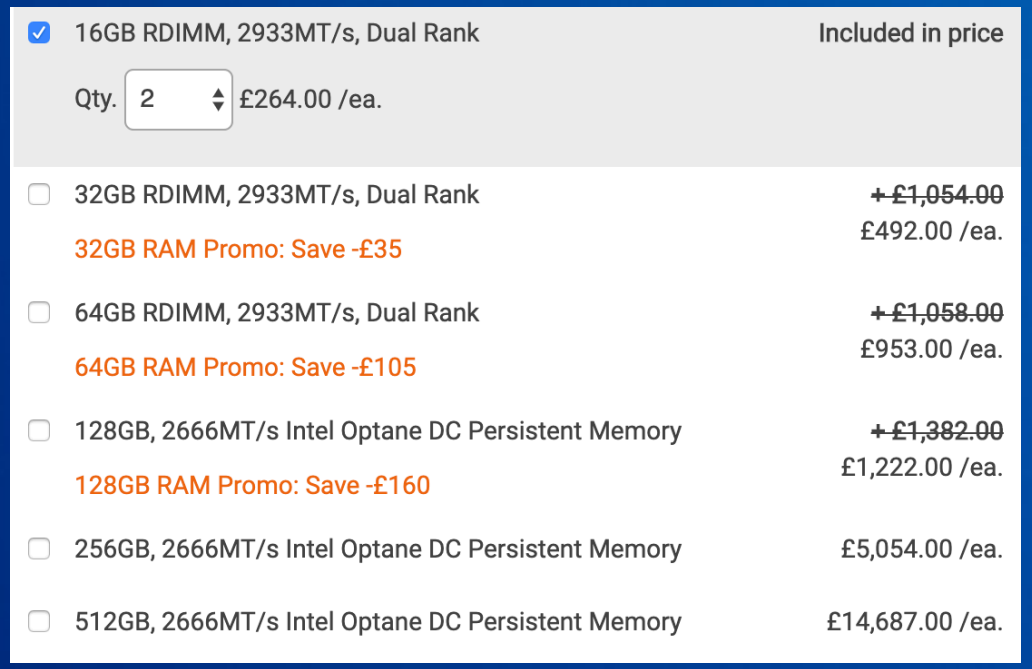
```
jgrant@homer:~$ cat edb.q
1"enter password: ";
-36!(hsym `$.z.x 1;read0 0);
system"l ",.z.x 0
jgrant@homer:~$ q edb.q db_enc/ kek1.key
KDB+ 4.0 2020.03.30 Copyright (C) 1993-2020 Kx Systems
l64/ 24()core 128387MB jgrant homer 127.0.1.1 EXPIRE 2020.06.30 AquaQ #55345

enter password: abc123
q)tables[]!count each value each tables[]
quote| 10000000
trade| 10000000
```

AQUAQ Analytics

Optane AppDirect support

What is Optane?

- New type of RAM –NVRAM (Non-Volatile) or persistent RAM
- Unlike DRAM, contents persist after machine restart – more like disk
- Slower than DRAM, much faster than SSD
- Cheaper than DRAM and maximum size per chip much higher

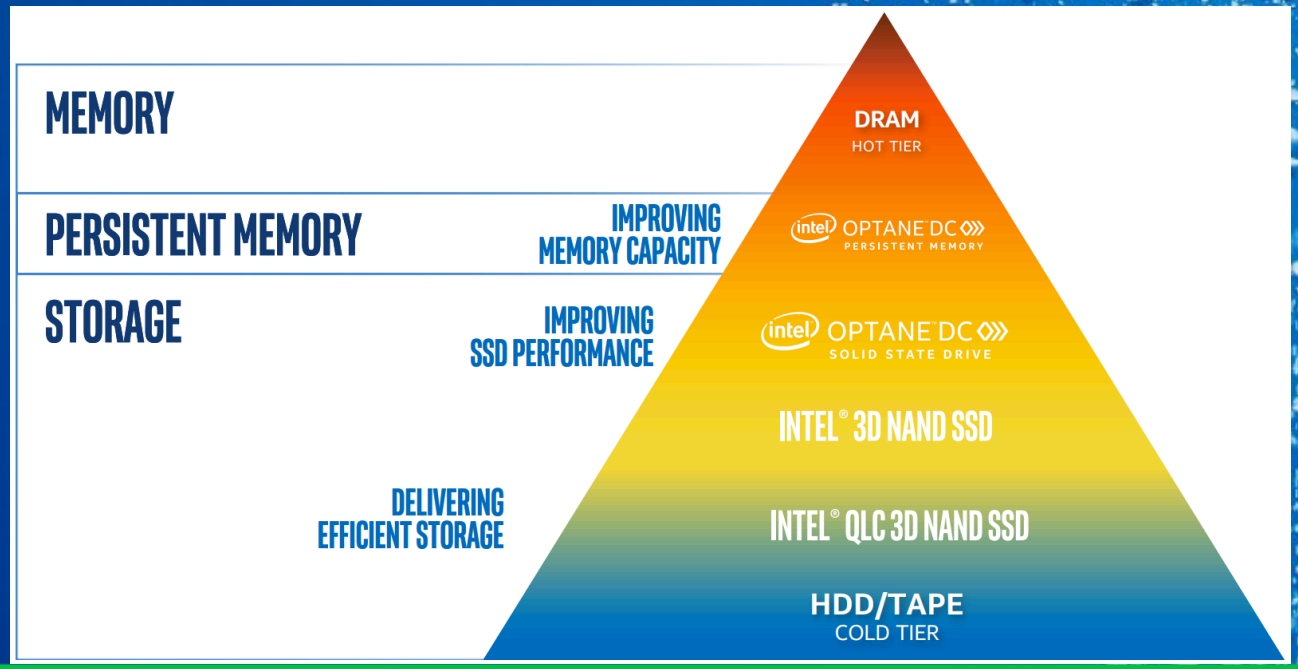


AQUAQ Analytics

Optane AppDirect support

Optane Performance

DDR4 memory accesses – 14ns
Optane DIMM – 350ns
NVMe Optane SSD access can take 10,000ns (10µs)
NVMe NAND SSD write – 30,000ns (30µs)
NVMe NAND SSD read – 120,000ns (120µs)
SATA NAND SSD read – 500,000ns (500µs or 0.5ms)
SATA NAND SSD write – 3,000,000ns (3,000µs or 3ms)
Disk drive seek – 100,000,000ns (100,000µs or 100ms)



AQUAQ Analytics

Optane AppDirect support

Optane Modes

- Storage mode – Optane presents as a disk
- Memory mode – Optane becomes main memory pool, DRAM is L4 cache
- AppDirect mode – DRAM and Optane present as separate memory pools to applications

AQUAQ Analytics

Optane AppDirect support

Storage Mode

- Applications talk to Optane through a file system interface
- Optane just behaves as very fast SSD
- File API impacts performance – not as fast as direct access
- Transparent to kdb+, compatible with all versions

AQUAQ Analytics

Optane AppDirect support

Memory Mode

- Optane becomes the main memory pool – so a machine with 256Gb of DRAM and 1Tb of Optane would appear to applications as having 1Tb RAM
- DRAM is managed as cache transparently by OS
- Frequently accessed objects in memory should remain in DRAM with occasional cache misses which have to hit Optane
- Compatible with all versions of kdb+ - could potentially have huge 1Tb+ RDBs if caching performance is suited to how kdb+ is accessing in memory data

AQUAQ Analytics

Optane AppDirect support

App Direct Mode

- Applications can see DRAM and Optane pools separately
- Software must be rewritten to take advantage of Optane memory
- Kdb+ 4.0 adds support for App Direct mode with compatible Intel processor (Cascade Lake - 2020)

AQUAQ Analytics

Optane AppDirect support

App Direct Mode – Kdb 4.0

- The namespace .m is now reserved for data stored in Optane memory
- Applications can maintain in-memory data either in regular namespaces (DRAM) or .m (Optane)
- Can also use a command line switch to switch domain globally
- Although in theory App Direct writes are persistent, Kx don't support recovery of .m contents after a process restart
- Could be part of a tiered storage system:

1d DRAM, 3d Optane, 1m SSD, 1m+ HDD

AQUAQ Analytics

Profiler

Kdb+ 4.0 includes an experimental built-in call-stack snapshot primitive that allows building a sampling profiler.

- A new function, .Q.prf0, returns a table representing a snapshot of the call stack at the time of the call in another kdb+ process.
- Takes the pid as an argument, doesn't require IPC
- Linux only – by default processes can only profile their direct children (e.g. started with system "q …")

column	description
name	assigned name of the function
file	path to the file containing the definition
line	line number of the definition
col	column offset of the definition, 0-based
text	function definition or source string
pos	execution position (caret) within text

AQUAQ Analytics

Profiler

DEMO

AQUAQ Analytics

References

https://code.kx.com/q/kb/mt-primitives

https://code.kx.com/q/kb/dare/

https://code.kx.com/q/kb/optane/

https://code.kx.com/q/kb/profiler/

AQUAQ Analytics

Thank you!

Q&A

Upcoming talks:

- Kdb+ in containers – 28th May
- Memverge Memory Machine – 4th June