StoneTor

How To I/O?

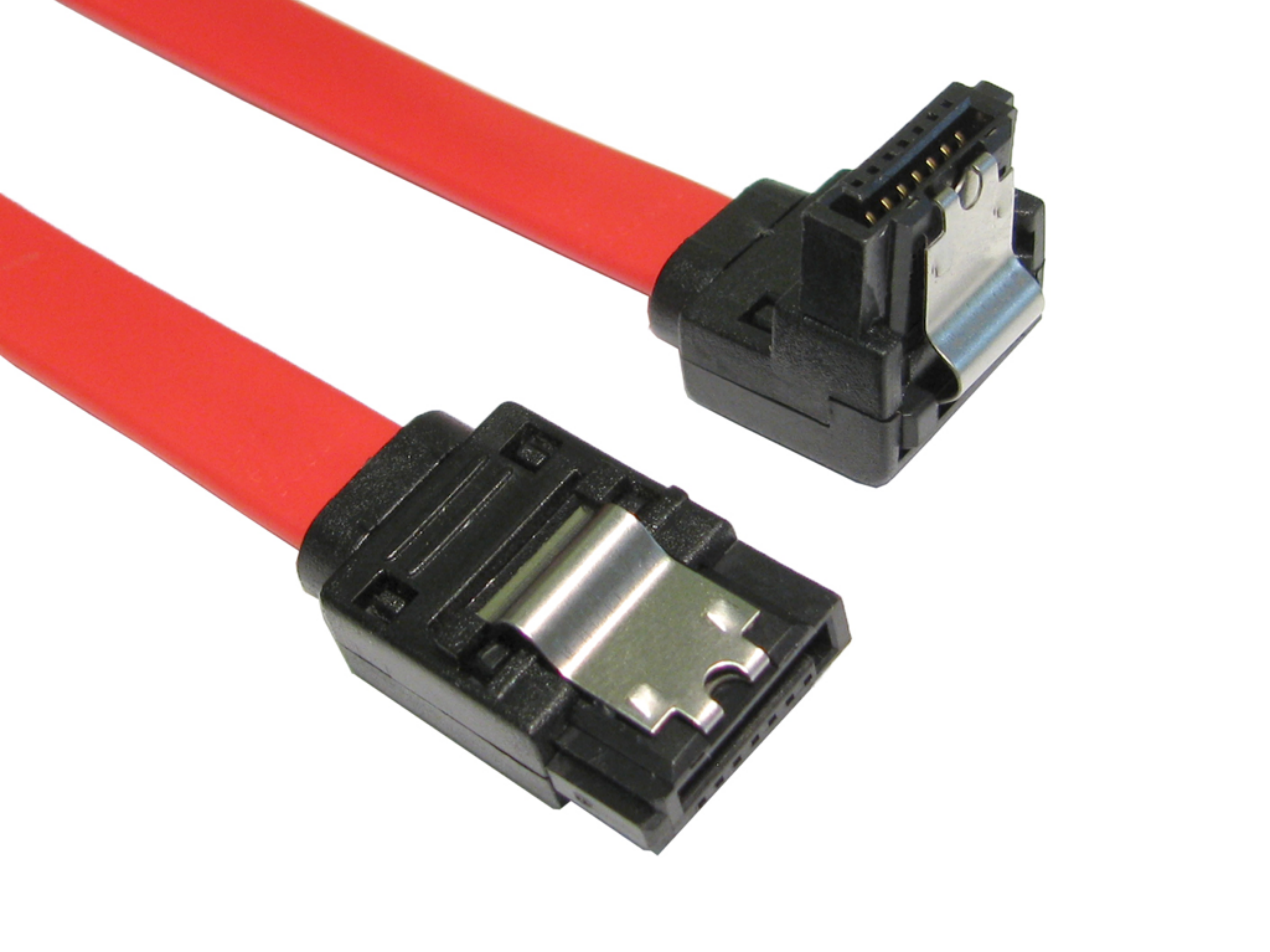
Todd L. Montgomery
@toddlmontgomery

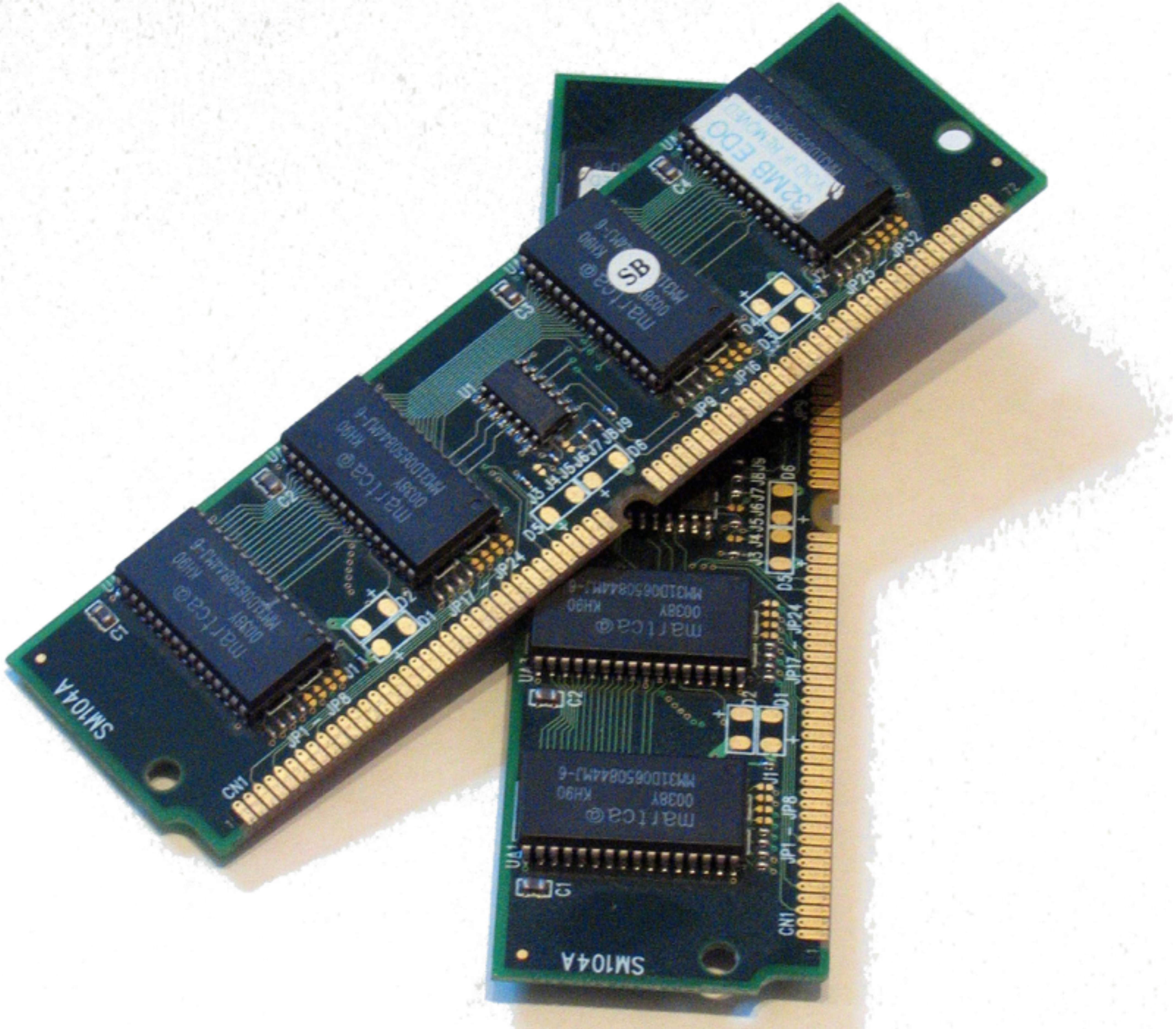
QCon
SAN FRANCISCO

- ☐ *I/O? Really?*
- ☐ *What used to be true*
- ☐ *… is still true*
- ☐ *Except when it isn’t*
- ☐ *Case Study: Aeron*
- ☐ *Takeaways*

☑ *I/O? Really?*

Western Digital
WD Caviar
Enhanced IDE Hard Drive
WD1000
100.0 GB
Product of Malaysia
DO NOT COVER ANY DRIVE HOLES
FRAGILE




32MB EDO
SB
marica
SM104A
SM104A

M.2
DDRSSD
PCIe - 3/4
100 GbE
…
OmniPath

CPUs
Cache / Memory

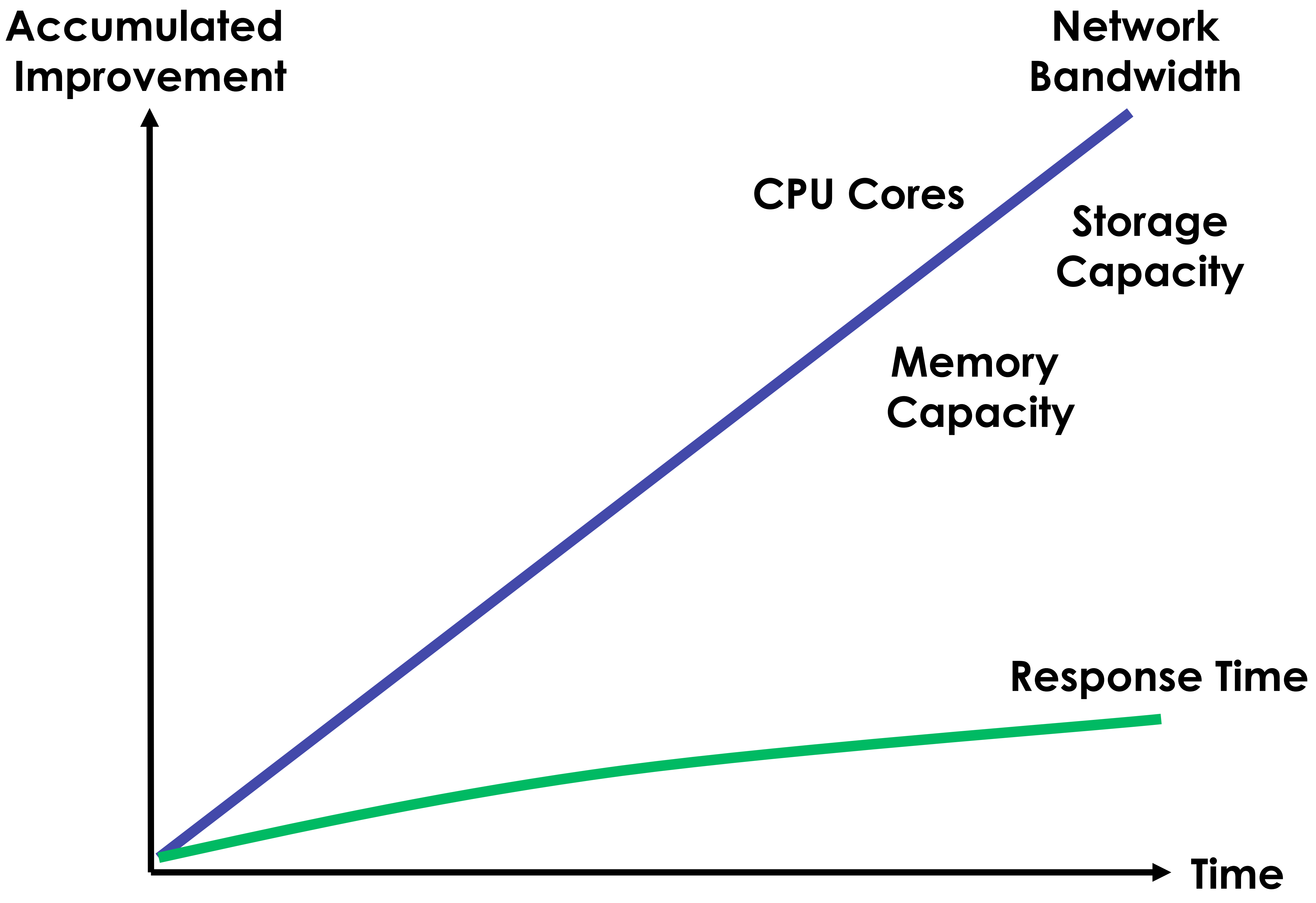
Fast networks - I/O-“ish"

Storage

700+ MBps

Network

10Gbps
<15us latency


Accumulated Improvement
Network Bandwidth
CPU Cores
Storage Capacity
Memory Capacity
Response Time
Time

It’s all good…

nothing to worry about…

right?

☑ *What used to be true*

Synchronous Read/Write

```java
final RandomAccessFile file = new RandomAccessFile(FILENAME, "w");

file.write(payload);
```

```java
final Socket socket = new Socket(SOME_HOST, SOME_PORT);

socket.getOutputStream().write(payload);
```

```java
final ByteBuffer buffer = file.getChannel()
    .map(FileChannel.MapMode.READ_WRITE, 0, file.length());

buffer.put(payload);
```

Streaming Read/Write

Striding

not just for memory

VM
Storage
RDMA

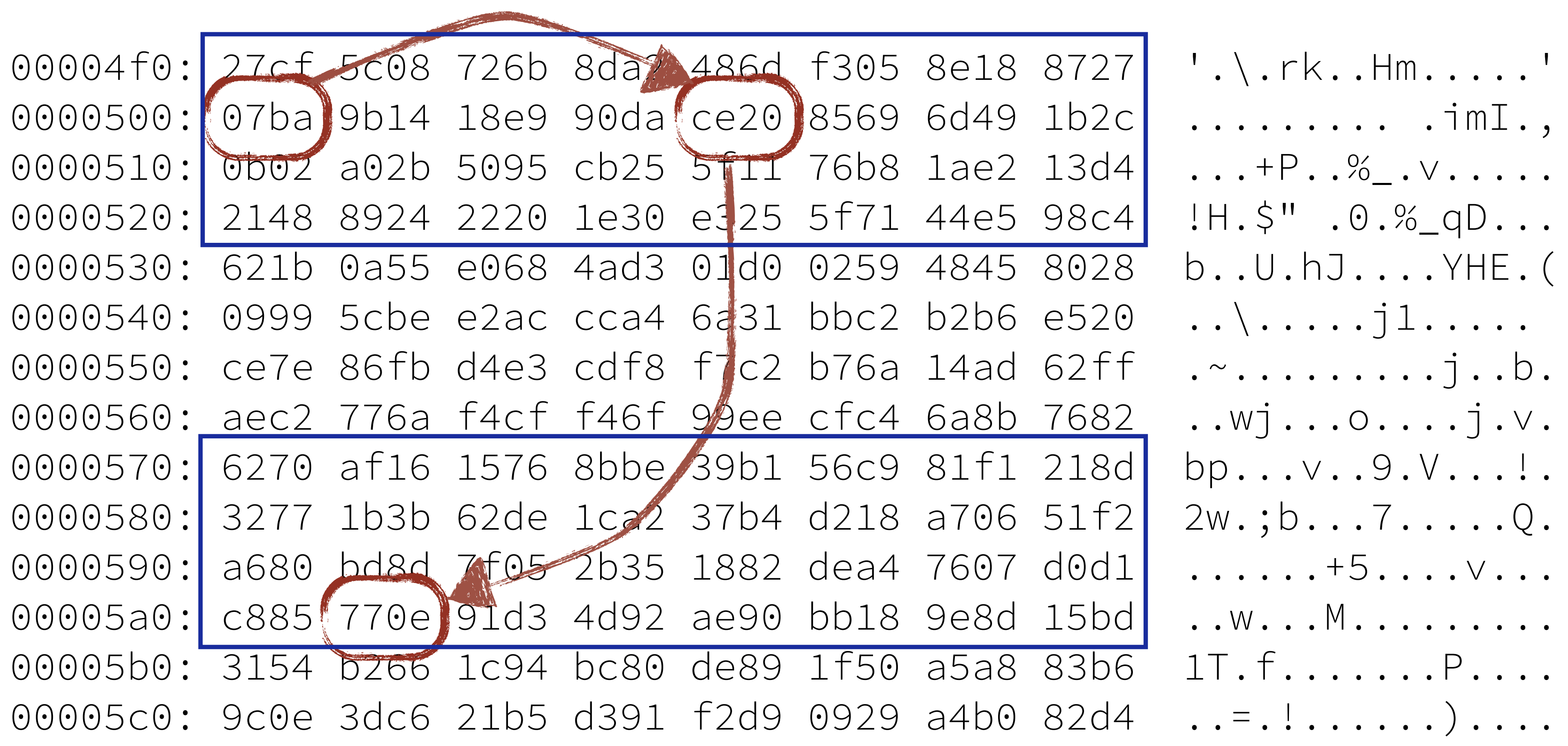
```
00004f0: 27cf 5c08 726b 8da2 486d f305 8e18 8727  '.\.rk..Hm.....'
0000500: 07ba 9b14 18e9 90da ce20 8569 6d49 1b2c  ......... .imI.,
0000510: 0b02 a02b 5095 cb25 5f11 76b8 1ae2 13d4  ...+P..%_.v.....
0000520: 2148 8924 2220 1e30 e325 5f71 44e5 98c4  !H.$" .0.%_qD...
0000530: 621b 0a55 e068 4ad3 01d0 0259 4845 8028  b..U.hJ....YHE.(
0000540: 0999 5cbe e2ac cca4 6a31 bbc2 b2b6 e520  ..\.....j1..... 
0000550: ce7e 86fb d4e3 cdf8 f7c2 b76a 14ad 62ff  .~.........j..b.
0000560: aec2 776a f4cf f46f 99ee cfc4 6a8b 7682  ..wj...o....j.v.
0000570: 6270 af16 1576 8bbe 39b1 56c9 81f1 218d  bp...v..9.V...!.
0000580: 3277 1b3b 62de 1ca2 37b4 d218 a706 51f2  2w.;b...7.....Q.
0000590: a680 bd8d 7f05 2b35 1882 dea4 7607 d0d1  ......+5....v...
00005a0: c885 770e 91d3 4d92 ae90 bb18 9e8d 15bd  ..w...M.........
00005b0: 3154 b266 1c94 bc80 de89 1f50 a5a8 83b6  1T.f.......P....
00005c0: 9c0e 3dc6 21b5 d391 f2d9 0929 a4b0 82d4  ..=.!......)....
```

```
00004f0: 27cf 5c08 726b 8da2 486d f305 8e18 8727  '.\.rk..Hm.....'
0000500: 07ba 9b14 18e9 90da ce20 8569 6d49 1b2c  .......... .imI.,
0000510: 0b02 a02b 5095 cb25 5f11 76b8 1ae2 13d4  ...+P..%_.v.....
0000520: 2148 8924 2220 1e30 e325 5f71 44e5 98c4  !H.$" .0.%_qD...
0000530: 621b 0a55 e068 4ad3 01d0 0259 4845 8028  b..U.hJ....YHE.(
0000540: 0999 5cbe e2ac cca4 6a31 bbc2 b2b6 e520  ..\.....j1.....
0000550: ce7e 86fb d4e3 cdf8 f7c2 b76a 14ad 62ff  .~.........j..b.
0000560: aec2 776a f4cf f46f 99ee cfc4 6a8b 7682  ..wj...o....j.v.
0000570: 6270 af16 1576 8bbe 39b1 56c9 81f1 218d  bp...v..9.V...!.
0000580: 3277 1b3b 62de 1ca2 37b4 d218 a706 51f2  2w.;b...7.....Q.
0000590: a680 bd8d 7f05 2b35 1882 dea4 7607 d0d1  ......+5....v...
00005a0: c885 770e 91d3 4d92 ae90 bb18 9e8d 15bd  ..w...M.........
00005b0: 3154 b266 1c94 bc80 de89 1f50 a5a8 83b6  1T.f.......P....
00005c0: 9c0e 3dc6 21b5 d391 f2d9 0929 a4b0 82d4  ..=.!......)....
```

```
00004f0: 27cf 5c08 726b 8da2 486d f305 8e18 8727  '.\.rk..Hm.....'
0000500: 07ba 9b14 18e9 90da ce20 8569 6d49 1b2c  ......... .imI.,
0000510: 0b02 a02b 5095 cb25 5f11 76b8 1ae2 13d4  ...+P..%_.v.....
0000520: 2148 8924 2220 1e30 e325 5f71 44e5 98c4  !H.$" .0.%_qD...
0000530: 621b 0a55 e068 4ad3 01d0 0259 4845 8028  b..U.hJ....YHE.(
0000540: 0999 5cbe e2ac cca4 6a31 bbc2 b2b6 e520  ..\.....j1.....
0000550: ce7e 86fb d4e3 cdf8 f7c2 b76a 14ad 62ff  .~.........j..b.
0000560: aec2 776a f4cf f46f 99ee cfc4 6a8b 7682  ..wj...o....j.v.
0000570: 6270 af16 1576 8bbe 39b1 56c9 81f1 218d  bp...v..9.V...!.
0000580: 3277 1b3b 62de 1ca2 37b4 d218 a706 51f2  2w.;b...7.....Q.
0000590: a680 bd8d 7f05 2b35 1882 dea4 7607 d0d1  ......+5....v...
00005a0: c885 770e 91d3 4d92 ae90 bb18 9e8d 15bd  ..w...M.........
00005b0: 3154 b266 1c94 bc80 de89 1f50 a5a8 83b6  1T.f.......P....
00005c0: 9c0e 3dc6 21b5 d391 f2d9 0929 a4b0 82d4  ..=.!......)....
```

```
00004f0: 27cf 5c08 726b 8da2 486d f305 8e18 8727  '.\.rk..Hm.....'
0000500: 07ba 9b14 18e9 90da ce20 8569 6d49 1b2c  ......... .imI.,
0000510: 0b02 a02b 5095 cb25 5f11 76b8 1ae2 13d4  ...+P..%_.v.....
0000520: 2148 8924 2220 1e30 e325 5f71 44e5 98c4  !H.$" .0.%_qD...
0000530: 621b 0a55 e068 4ad3 01d0 0259 4845 8028  b..U.hJ....YHE.(
0000540: 0999 5cbe e2ac cca4 6a31 bbc2 b2b6 e520  ..\.....j1.....
0000550: ce7e 86fb d4e3 cdf8 f7c2 b76a 14ad 62ff  .~.........j..b.
0000560: aec2 776a f4cf f46f 99ee cfc4 6a8b 7682  ..wj...o....j.v.
0000570: 6270 af16 1576 8bbe 39b1 56c9 81f1 218d  bp...v..9.V...!.
0000580: 3277 1b3b 62de 1ca2 37b4 d218 a706 51f2  2w.;b...7.....Q.
0000590: a680 bd8d 7f05 2b35 1882 dea4 7607 d0d1  ......+5....v...
00005a0: c885 770e 91d3 4d92 ae90 bb18 9e8d 15bd  ..w...M.........
00005b0: 3154 b266 1c94 bc80 de89 1f50 a5a8 83b6  1T.f.......P....
00005c0: 9c0e 3dc6 21b5 d391 f2d9 0929 a4b0 82d4  ..=.!......)....
```

```
00004f0: 27cf 5c08 726b 8da2 486d f305 8e18 8727  '.\.rk..Hm.....'
0000500: 07ba 9b14 18e9 90da ce20 8569 6d49 1b2c  ......... .imI.,
0000510: 0b02 a02b 5095 cb25 5f11 76b8 1ae2 13d4  ...+P..%_.v.....
0000520: 2148 8924 2220 1e30 e325 5f71 44e5 98c4  !H.$" .0.%_qD...
0000530: 621b 0a55 e068 4ad3 01d0 0259 4845 8028  b..U.hJ....YHE.(
0000540: 0999 5cbe e2ac cca4 6a31 bbc2 b2b6 e520  ..\.....j1..... 
0000550: ce7e 86fb d4e3 cdf8 f7c2 b76a 14ad 62ff  .~.........j..b.
0000560: aec2 776a f4cf f46f 99ee cfc4 6a8b 7682  ..wj...o....j.v.
0000570: 6270 af16 1576 8bbe 39b1 56c9 81f1 218d  bp...v..9.V...!.
0000580: 3277 1b3b 62de 1ca2 37b4 d218 a706 51f2  2w.;b...7.....Q.
0000590: a680 bd8d 7f05 2b35 1882 dea4 7607 d0d1  ......+5....v...
00005a0: c885 770e 91d3 4d92 ae90 bb18 9e8d 15bd  ..w...M.........
00005b0: 3154 b266 1c94 bc80 de89 1f50 a5a8 83b6  1T.f.......P....
00005c0: 9c0e 3dc6 21b5 d391 f2d9 0929 a4b0 82d4  ..=.!......)....
```



```
00004f0: 27cf 5c08 726b 8da2 486d f305 8e18 8727  '.\.rk..Hm.....'
0000500: 07ba 9b14 18e9 90da ce20 8569 6d49 1b2c  ......... .imI.,
0000510: 0b02 a02b 5095 cb25 5f11 76b8 1ae2 13d4  ...+P..%_.v.....
0000520: 2148 8924 2220 1e30 e325 5f71 44e5 98c4  !H.$" .0.%_qD...
0000530: 621b 0a55 e068 4ad3 01d0 0259 4845 8028  b..U.hJ....YHE.(
0000540: 0999 5cbe e2ac cca4 6a31 bbc2 b2b6 e520  ..\.....j1.....
0000550: ce7e 86fb d4e3 cdf8 f7c2 b76a 14ad 62ff  .~.........j..b.
0000560: aec2 776a f4cf f46f 99ee cfc4 6a8b 7682  ..wj...o....j.v.
0000570: 6270 af16 1576 8bbe 39b1 56c9 81f1 218d  bp...v..9.V...!.
0000580: 3277 1b3b 62de 1ca2 37b4 d218 a706 51f2  2w.;b...7.....Q.
0000590: a680 bd8d 7f05 2b35 1882 dea4 7607 d0d1  ......+5....v...
00005a0: c885 770e 91d3 4d92 ae90 bb18 9e8d 15bd  ..w...M.........
00005b0: 3154 b266 1c94 bc80 de89 1f50 a5a8 83b6  1T.f.......P....
00005c0: 9c0e 3dc6 21b5 d391 f2d9 0929 a4b0 82d4  ..=.!......)....
```

SSDs
RDMA

Random Access is OK!?…

☑ *… is still true*

Striding

still works well

Striding

still works well
+ more patterns

Random Access

incurs a penalty

Random Access

incurs a ***PENALTY***

Random Access

-10%, -10x, -100x*

Streaming Read/Write

still true

☑ *Except when it isn’t*

Synchronous
Read/Write

never really was true

[Incorrect] *Assumption*

Oh.. You’re doing I/O, you don’t care about being fast

Scheduling Jitter
Locks

LOCKS!!!

It's more likely you are blocked on locks than on the I/O device itself

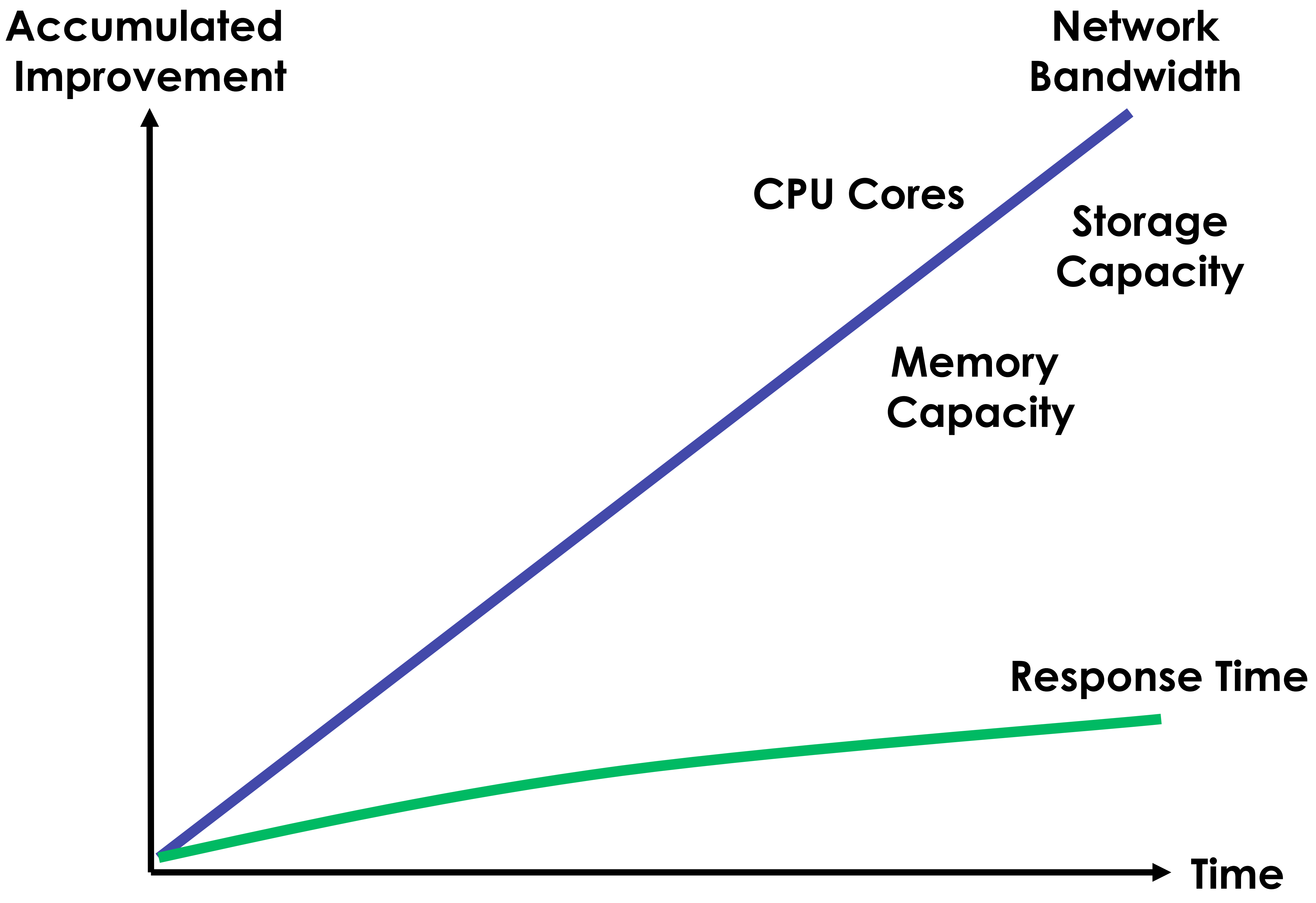
Most I/O is so fast, that the price of locking can overshadow it

But it’s not just locking…

Data Formats (binary?)
Algorithms
Protocols
…

It is <u>highly</u> doubtful that you are being held back by the network or storage

The reason(s)


Accumulated Improvement
Network Bandwidth
CPU Cores
Storage Capacity
Memory Capacity
Response Time
Time

The OS has locks
*The runtime has locks**
*Algorithms have coherence***

Algorithms Matter

Configuration that Outperforms a Single Thread

SSD + 1 thread of goodness

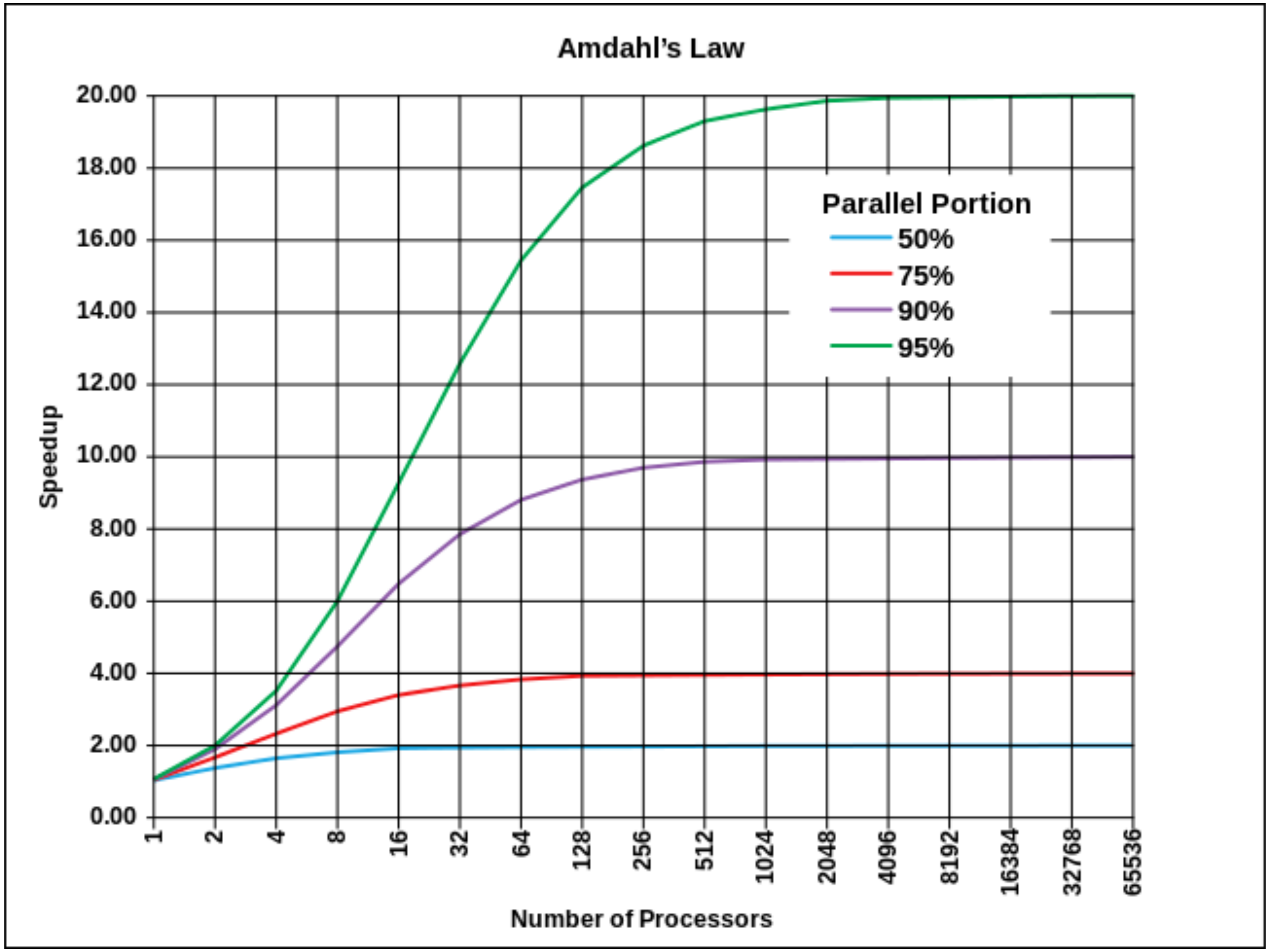
>

128 cores of so-so

http://blog.acolyer.org/2015/06/05/scalability-but-at-what-cost/

http://www.frankmcsherry.org/graph/scalability/cost/2015/01/15/COST.html

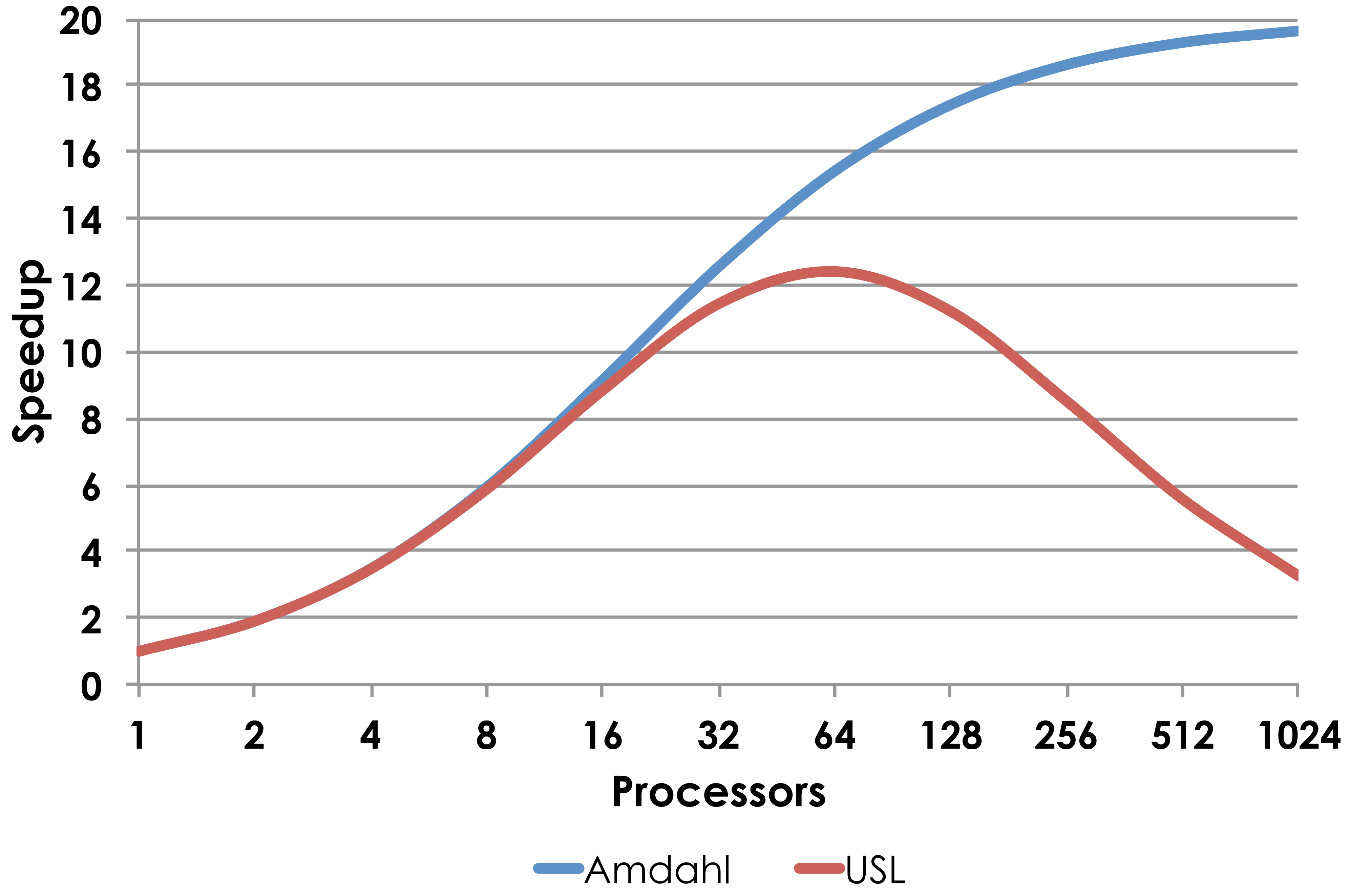
You can't escape the Math


Amdahl's Law
Speedup
20.00
18.00
16.00
14.00
12.00
10.00
8.00
6.00
4.00
2.00
0.00
1
2
4
8
16
32
64
128
256
512
1024
2048
4096
8192
16384
32768
65536
Number of Processors
Parallel Portion
50%
75%
90%
95%

Contention <u>isn't</u> the biggest enemy

<u>Coherence</u> is!

Universal Scalability Law



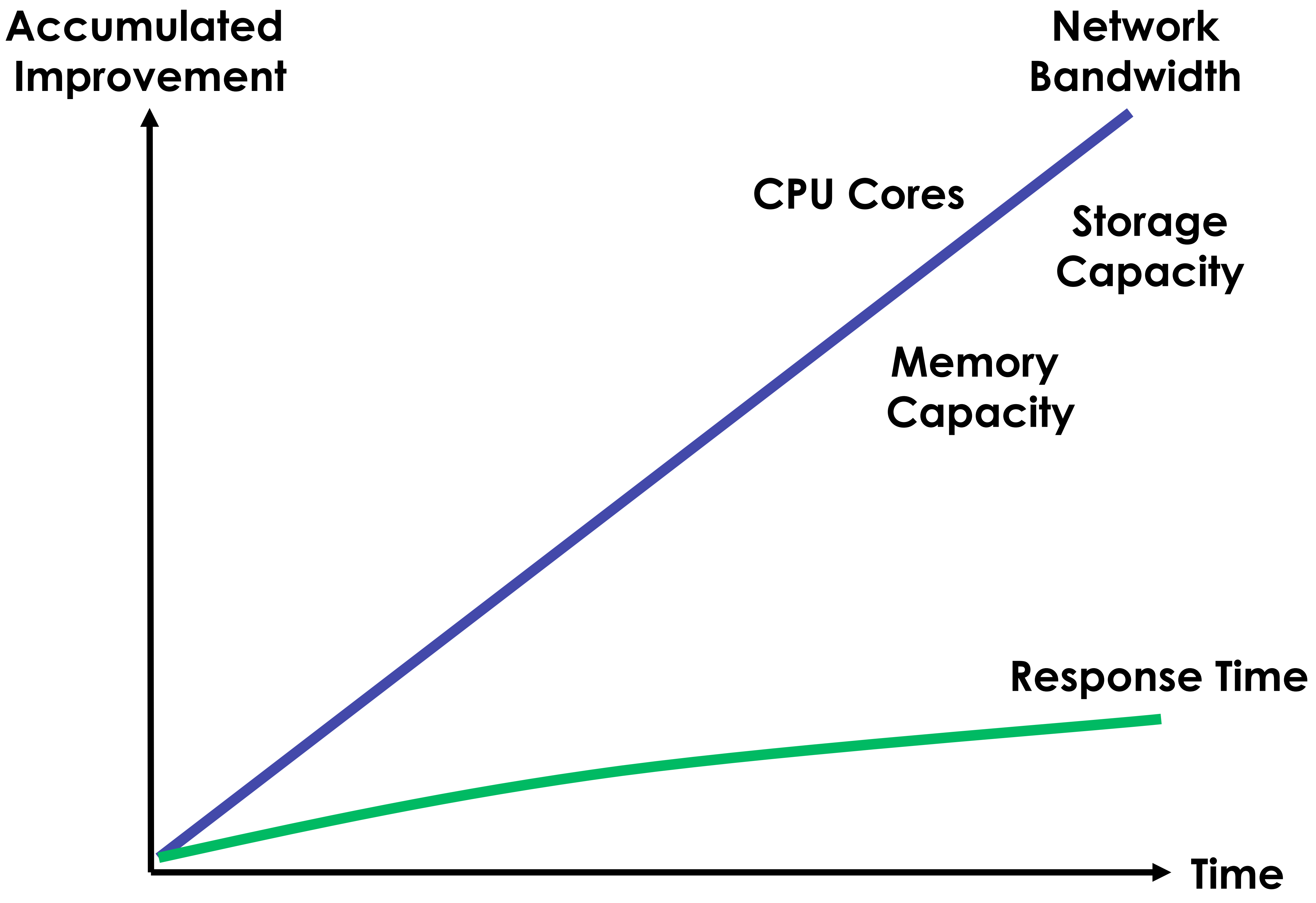
Also

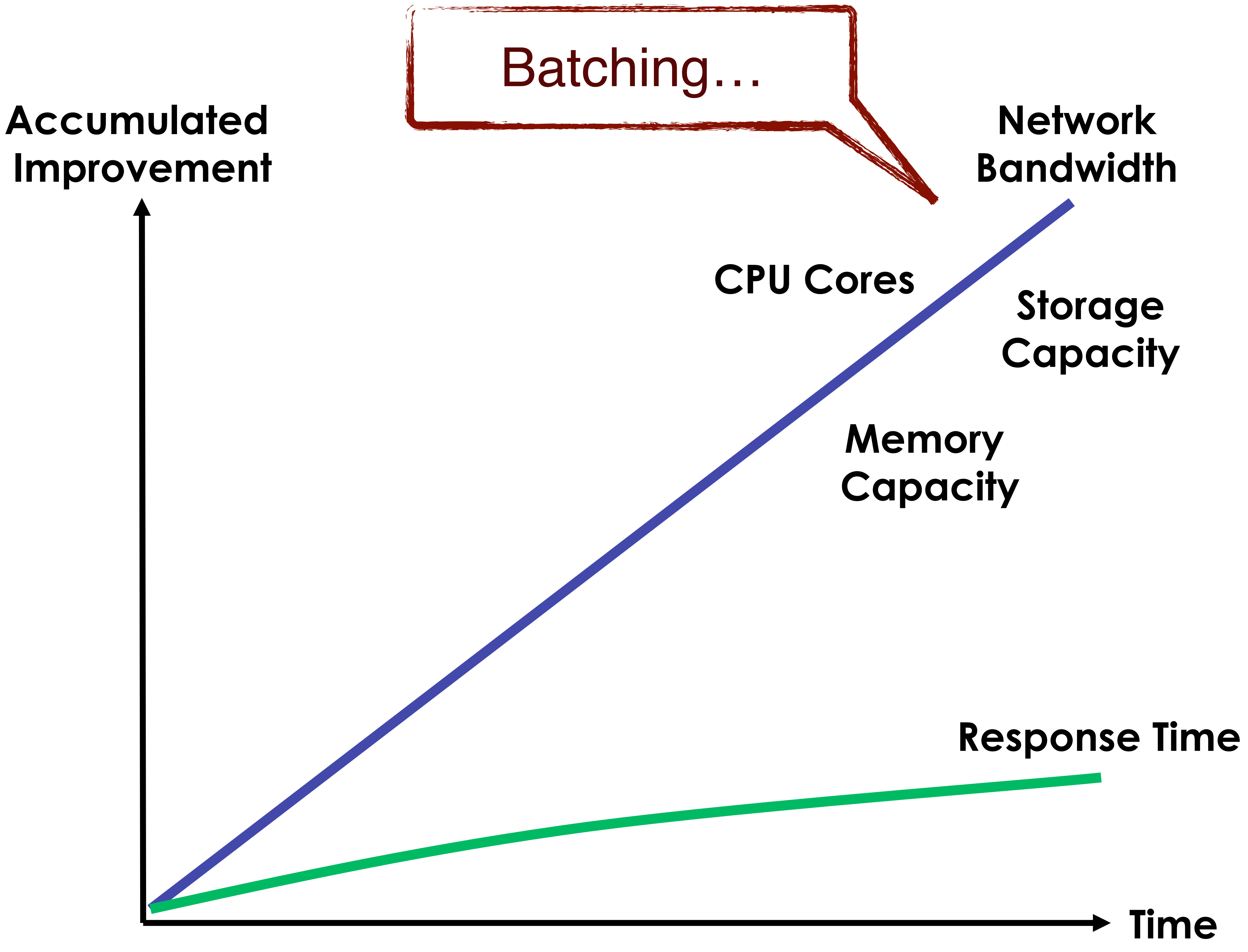
Coherence traffic eats up bandwidth

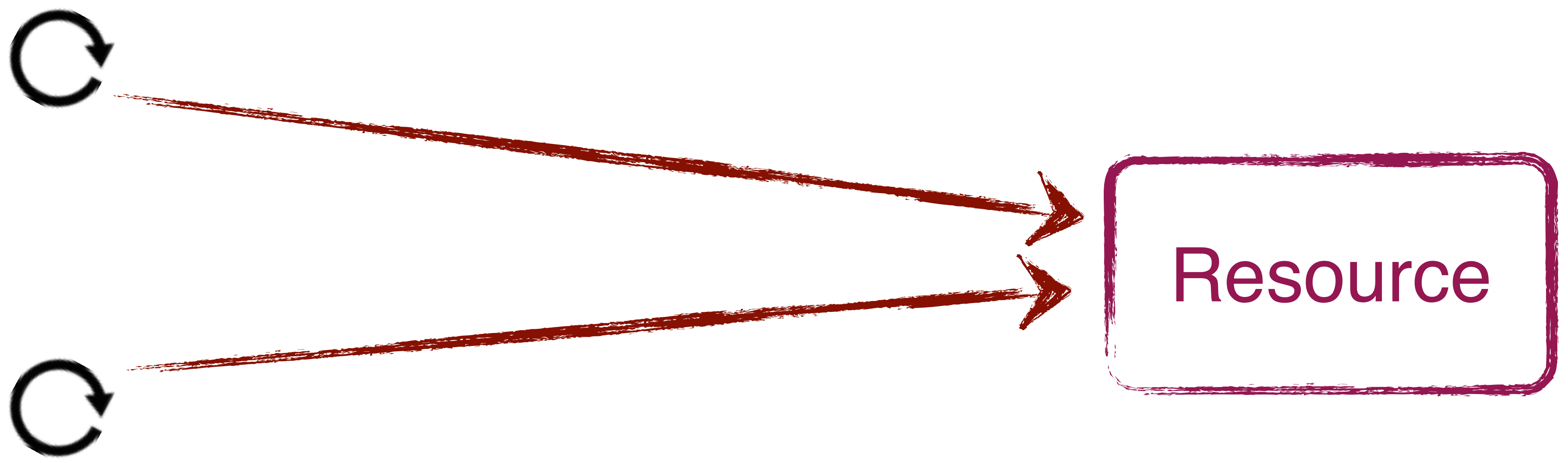
Defeating Contention

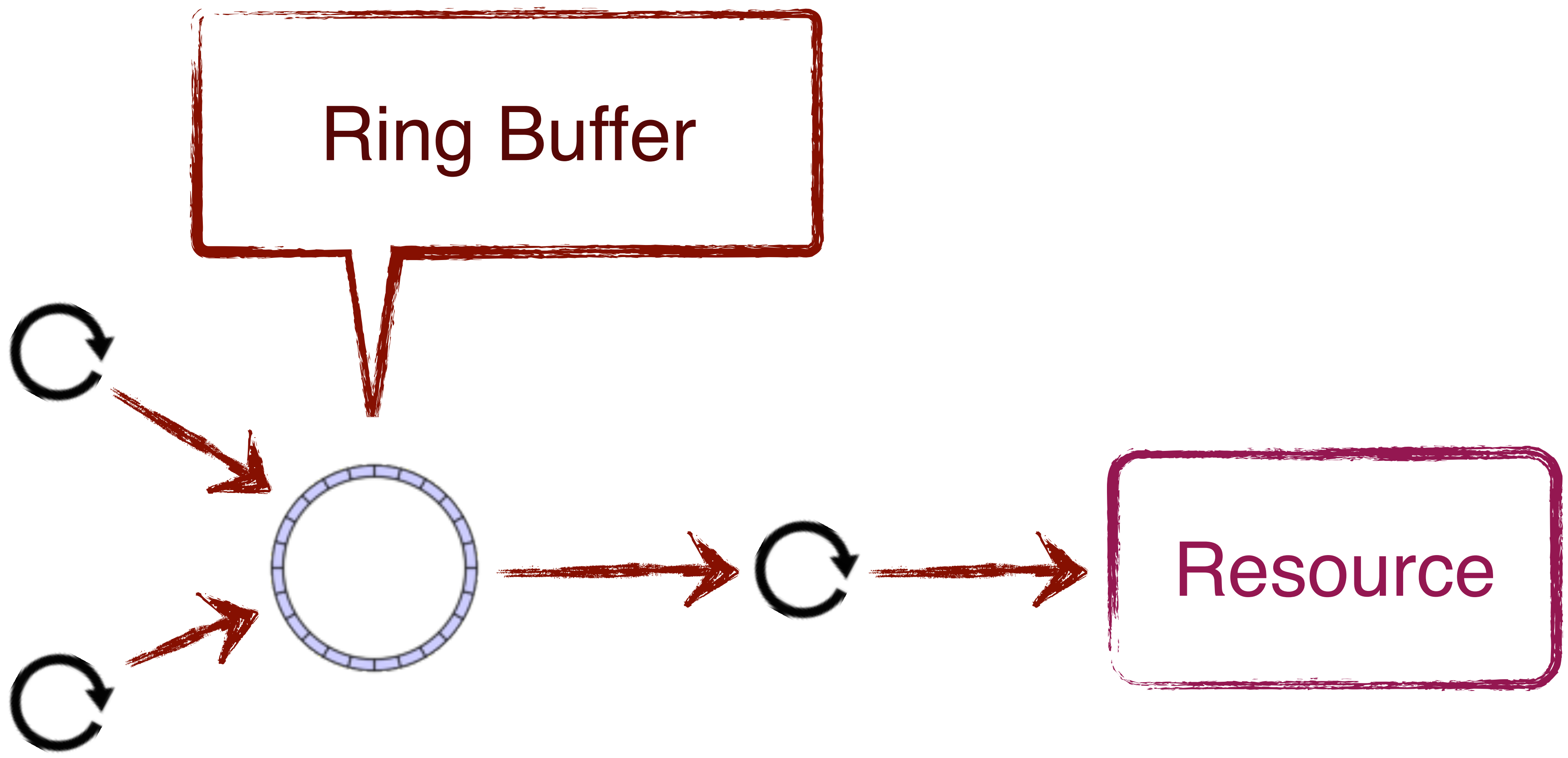
Smart Batching (Natural Batching)

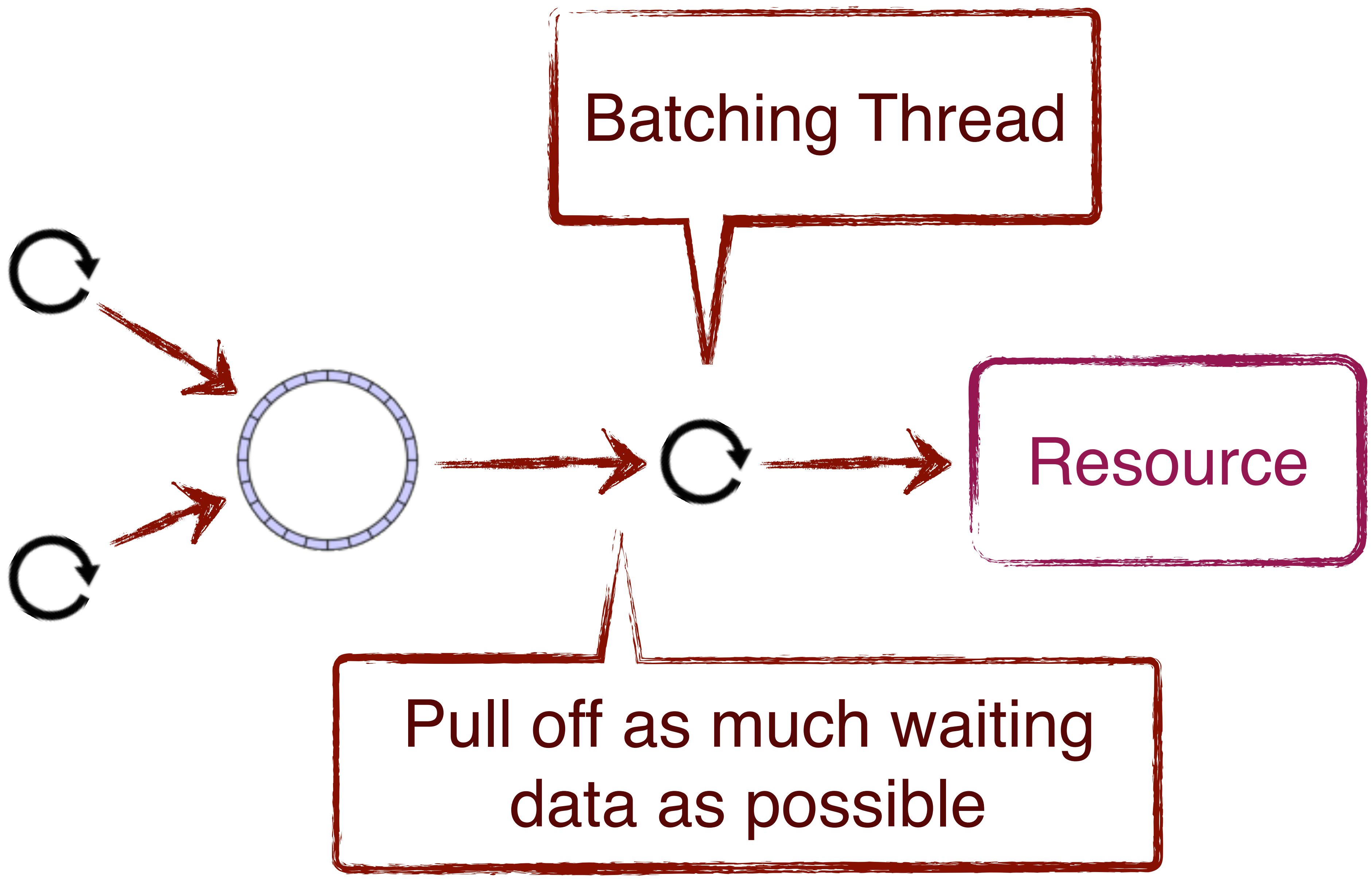
http://mechanical-sympathy.blogspot.com/2011/10/smart-batching.html


Accumulated Improvement
Network Bandwidth
CPU Cores
Storage Capacity
Memory Capacity
Response Time
Time


Batching…
Accumulated Improvement
Network Bandwidth
CPU Cores
Storage Capacity
Memory Capacity
Response Time
Time


Resource


Ring Buffer
Resource


Batching Thread
Resource
Pull off as much waiting data as possible

- ☑ *Single Writer Principle*
- ☑ *Avoid Resource Contention*
- ☑ *Batching only when needed*
- ☑ *Rate Decoupling*
- ☑ *Back Pressure*

Reading

sendfile / slice / transferTo

Read in (multiple) page size chunks

Reduce kernel calls

Async I/O

The cost of locks

DatagramChannelImpl

```
public int write(ByteBuffer buf)
{
    synchronized (writeLock) {
        synchronized (stateLock) { … } … }
}

public int read(ByteBuffer buf)
{
    synchronized (readLock) {
        synchronized (stateLock) { … } … }
}
```

`send` & `receive` are similar

http://grepcode.com/file/repository.grepcode.com/java/root/jdk/openjdk/8-b132/sun/nio/ch/DatagramChannelImpl.java#DatagramChannelImpl

Bias Locking

Same thread constructing, reading, & writing

= 1+ microsecond

Freedom!
Lock-Free, Wait-Free

http://en.wikipedia.org/wiki/Non-blocking_algorithm

FREEDOM!

Words Matter

Obstruction-Freedom

Partially completed operations aborted & changes made rolled back

Lock-Freedom

Individual thread may starve, but guaranteed system-wide throughput

Lock-Free is Obstruction-Free

Wait-Freedom

Starvation free and guaranteed system-wide throughput

Wait-Free is Lock-Free

These properties are awesome!

Who wouldn’t want them?

System-wide properties start at the lowest level

Essence

Just because we could take an action right now, doesn’t mean we should

PATIENCE

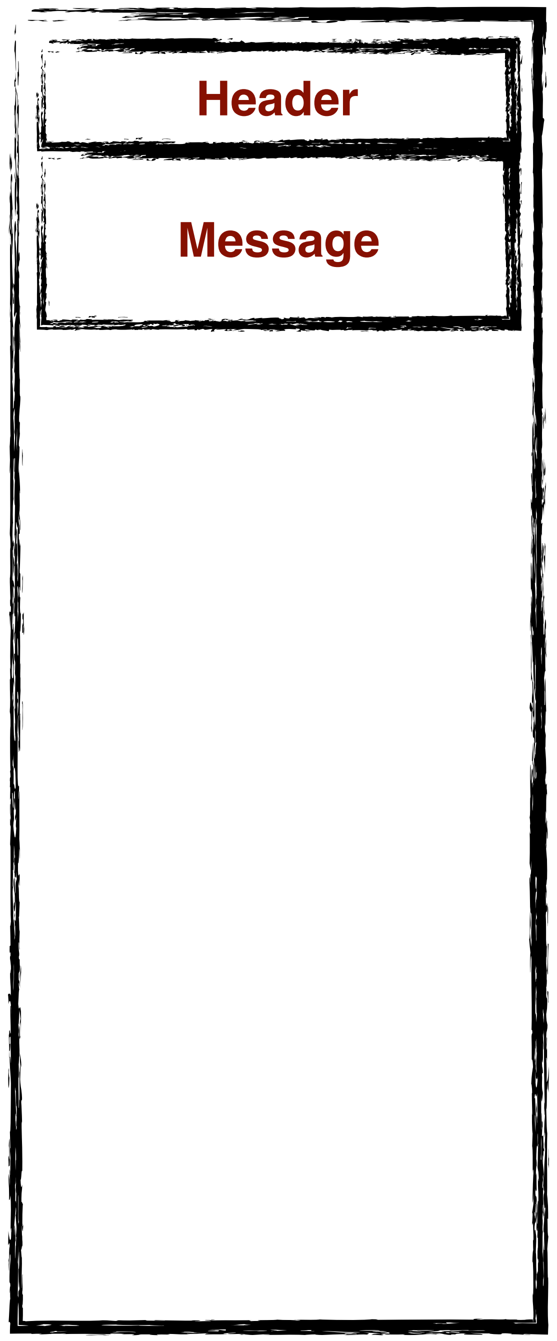
Because you know that someday, you'll be able to beat the #$%^ out of that cat.

☑ Case Study: Aeron

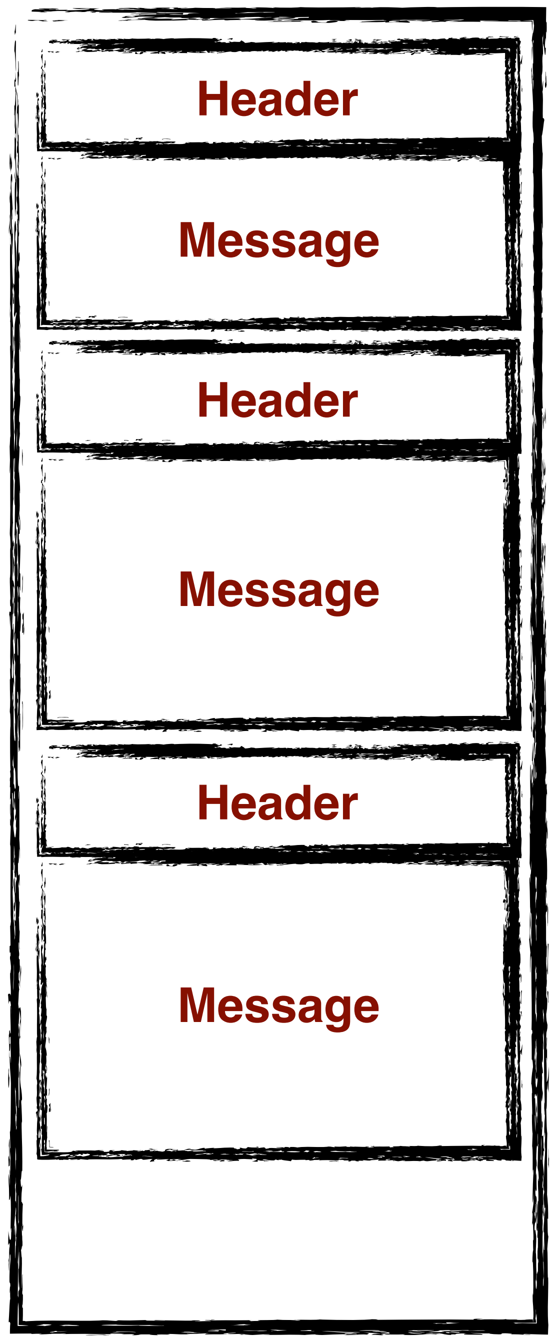
https://github.com/real-logic/Aeron

Append-only Data Structures

Log



Log

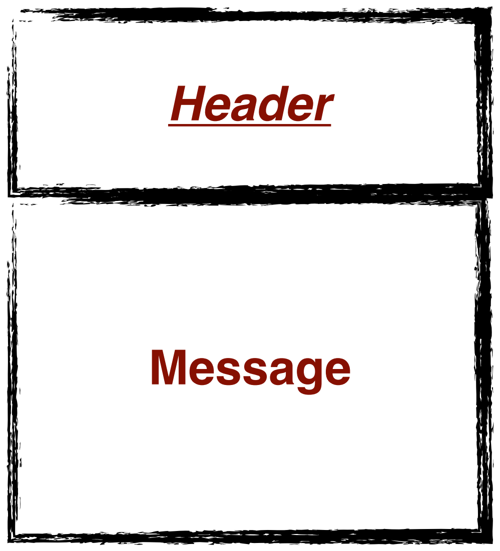


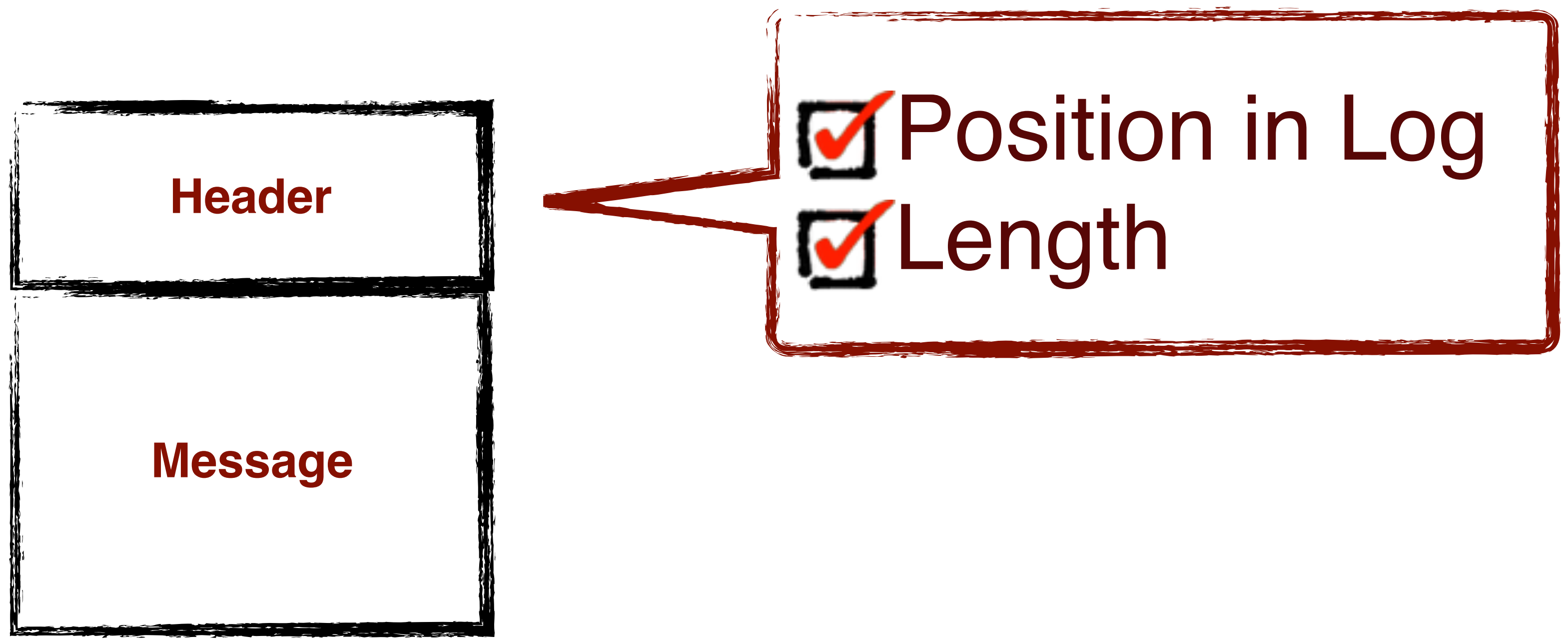
Efficiently Replicating an Append-only Log

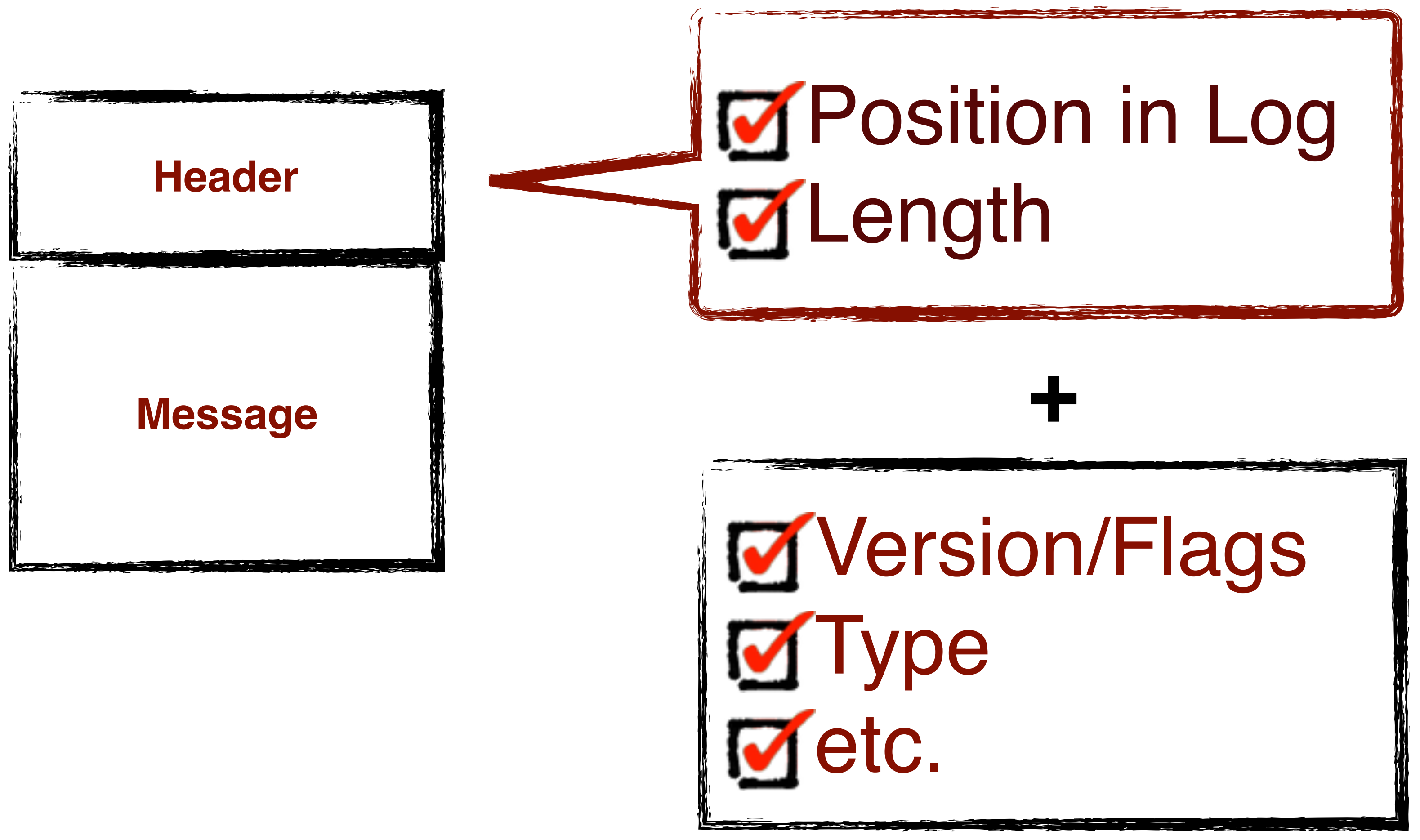
What If…?

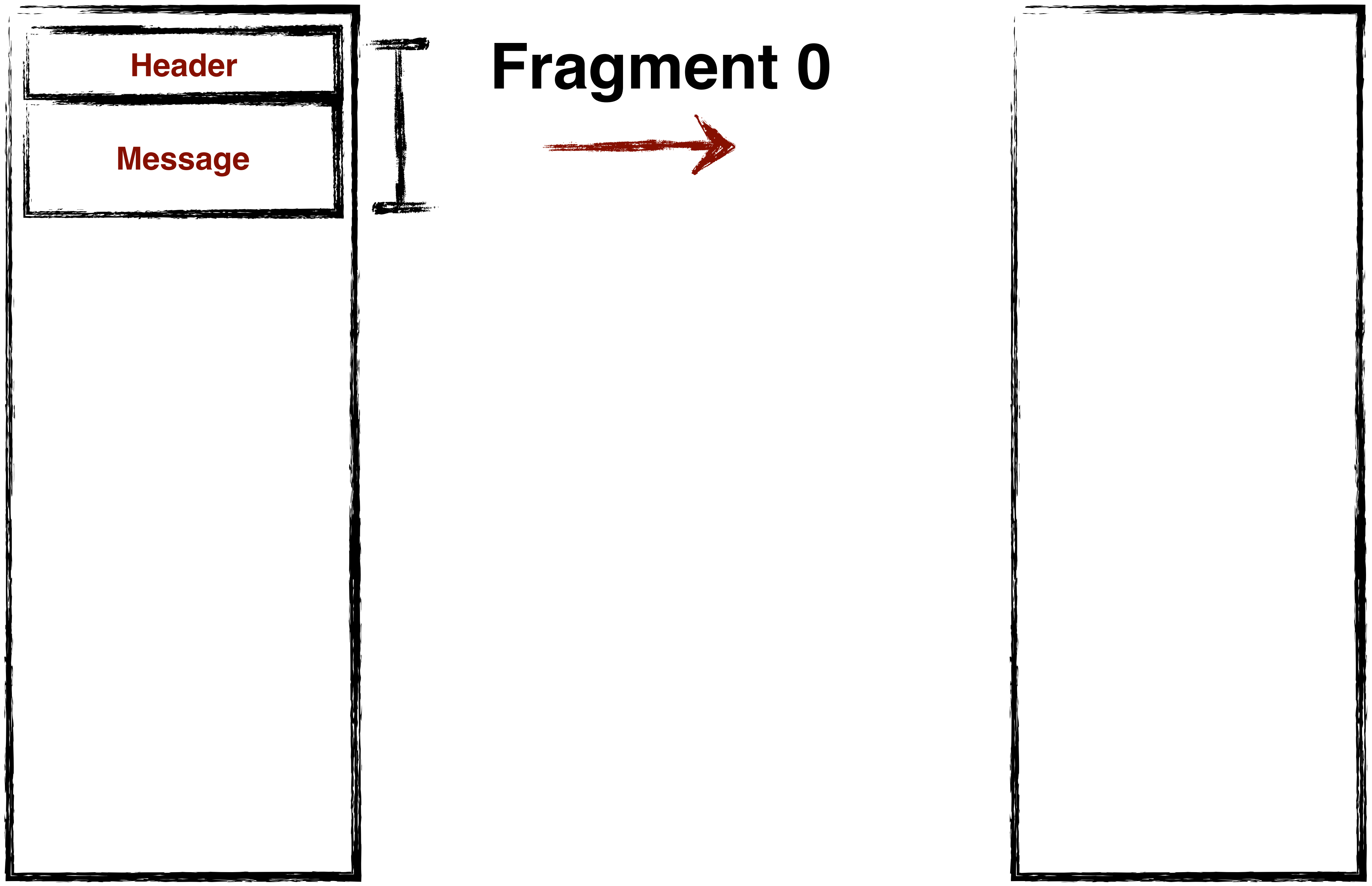
The Data Structure could be directly sent to the “network”?

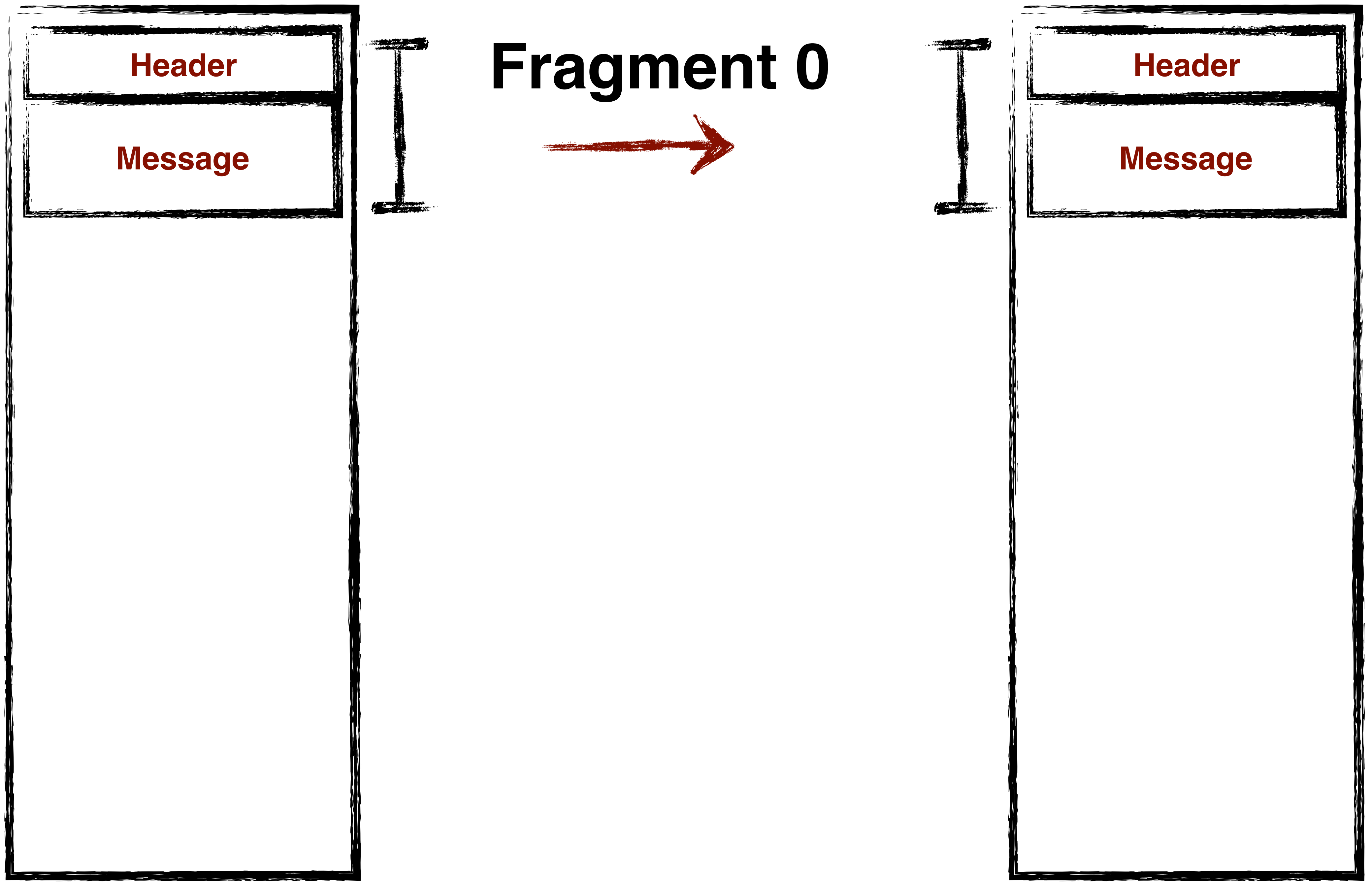
and saved to “storage”?

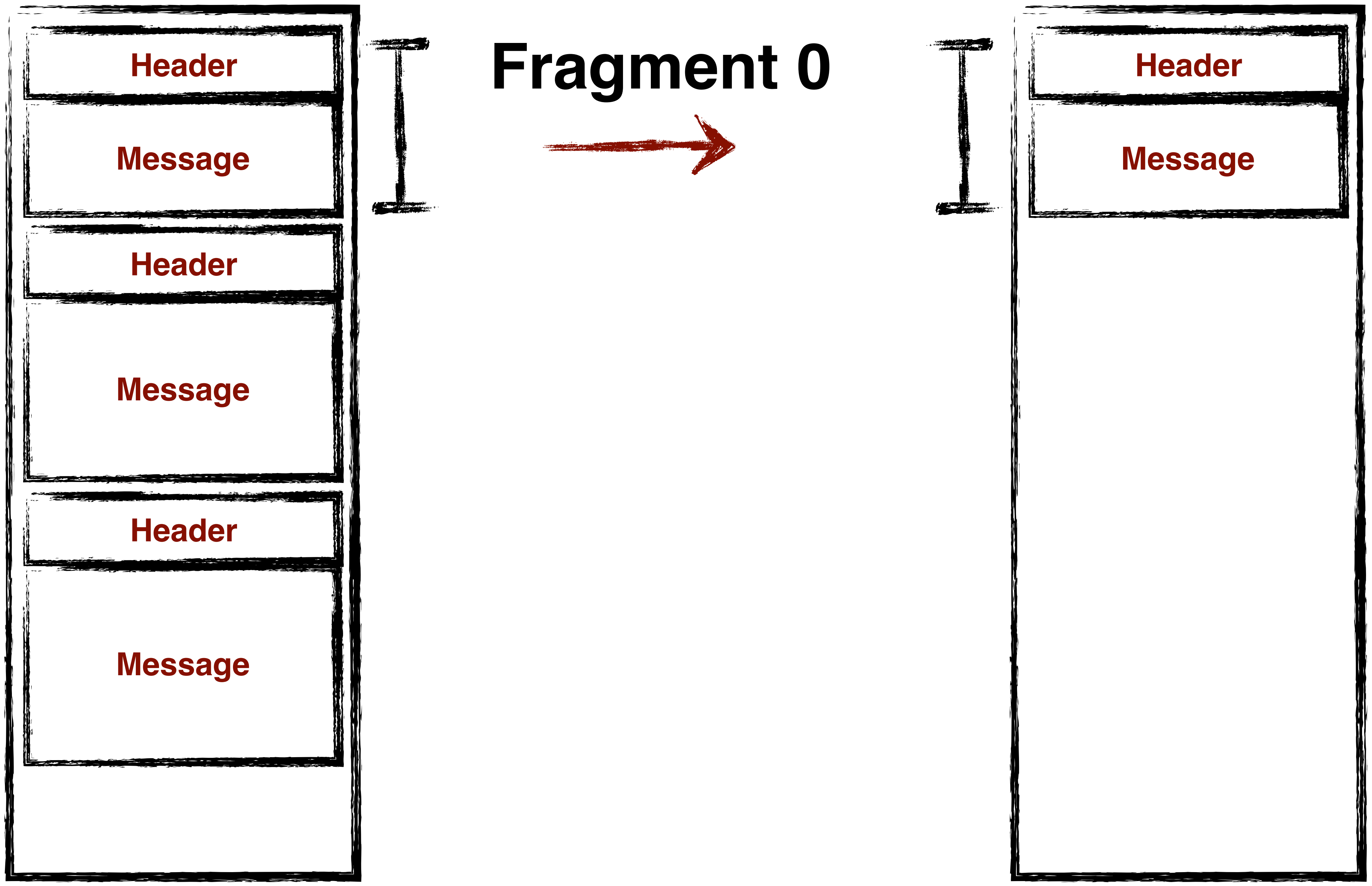

Header
Message

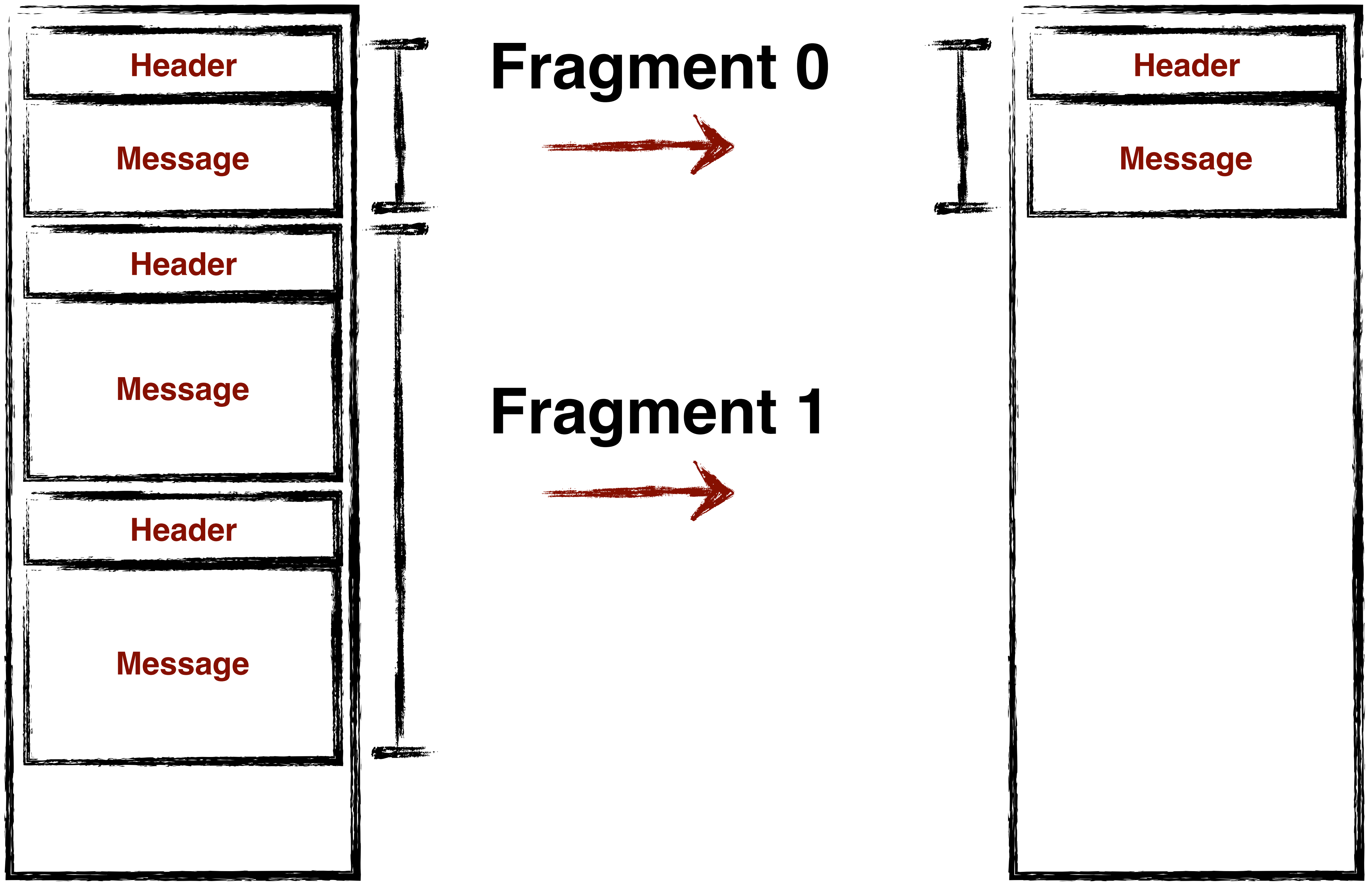

Header
Message
☑ Position in Log
☑ Length

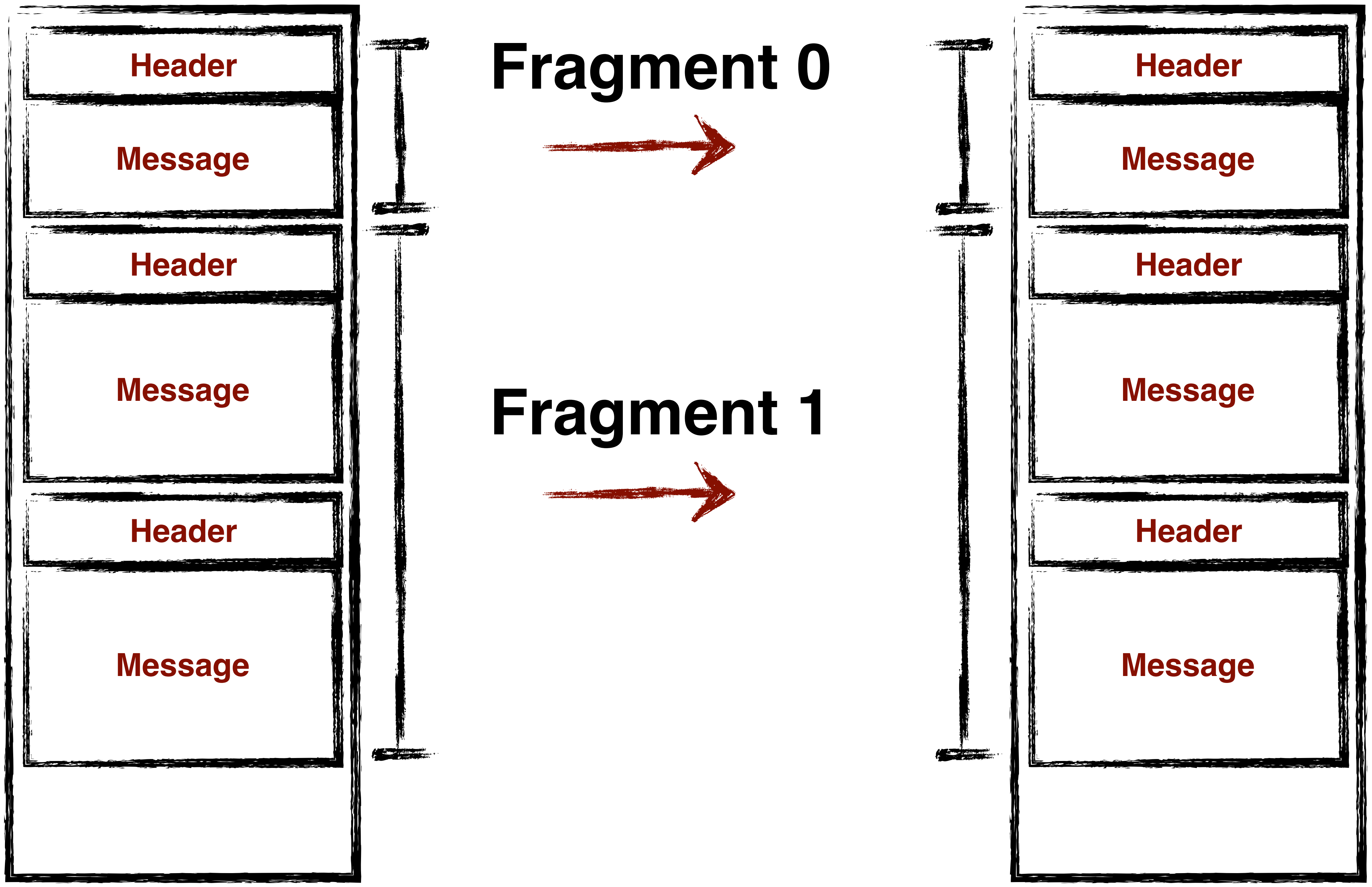

Header
Message
Position in Log
Length
+
Version/Flags
Type
etc.


Header
Message
Fragment 0


Header
Message
Fragment 0
Header
Message


Header
Message
Header
Message
Header
Message
Fragment 0
Header
Message


Header
Message
Header
Message
Header
Message
Fragment 0
Fragment 1
Header
Message


Header
Message
Header
Message
Header
Message
Fragment 0
Fragment 1
Header
Message
Header
Message
Header
Message

✓ *Takeaways*

We are loosing 30% memory bandwidth…*

Oh &^%(&^!

Observed by Martin Thompson

Stream over Data
Predictable Access
Batching
Algorithms
Avoid contention
*Avoid coherence**

Questions?

StoneTor

- Aeron *https://github.com/real-logic/Aeron*
- Twitter @toddlmontgomery

Thank You!