Introduction

KEVIN HOUSTOUN,

RAPID ADDITION

Biography: Kevin Houstoun

Kevin has over 20 year's experience of working within financial markets technology including Robert Fleming, LSE, Salomon Brothers and HSBC.

Kevin helped to establish the FIX protocol in Europe. His first FIX engine was given away as a learning aid that helped many enterprises further their knowledge of FIX. Kevin is the designer of the FIX Repository for FIX Protocol Limited, Co-chairs the Global Technical Committee, and is an active member of the Global Steering Committee. He is the lead Expert Group member, UK Government Foresight Committee looking at the Future of Computer Based Trading.

RUPERT SMITH,

CONTRIBUTOR TO THE APACHE QPID IMPLEMENTATION OF AMQP

Biography: Rupert Smith

Rupert Smith is a Java programmer, who started out tinkering at the low-level end of things; Assembler and C. He first worked on messaging software as a contributor to the Apache Qpid implementation of AMQP. He is currently working for Rapid Addition which, with its low-latency focus, is a fertile place to mix ideas from his background. He studied Computer Science at Cambridge University and maintains an interest in compiler development, particularly for logic based languages. His main hobbies are camping and nature conservation.

Introduction

Rationale
Origin
Road to Gen Zero
Details

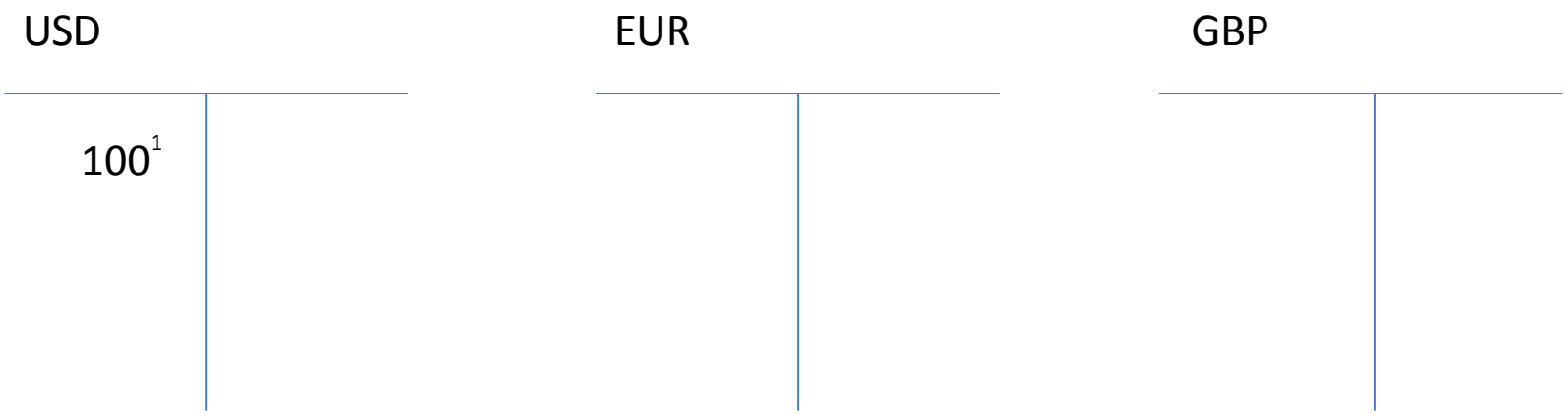
Box Trade - $100

USD/EUR = 1.3158

EUR/GBP = 1.1875

USD/GBP = 1.5625

Ignoring spreads, commissions and transaction costs



Box Trade - $100

USD/EUR = 1.3158

EUR/GBP = 1.1875

USD/GBP = 1.5625

USD		EUR		GBP	
100^{1}	100^{2}	76^{2}			

Box Trade - $100

USD/EUR = 1.3158

EUR/GBP = 1.1875

USD/GBP = 1.5625

USD		EUR		GBP	
100^{1}	100^{2}	76^{2}	76^{3}	64^{3}	

Box Trade - $100

USD/EUR = 1.3158

EUR/GBP = 1.1875

USD/GBP = 1.5625

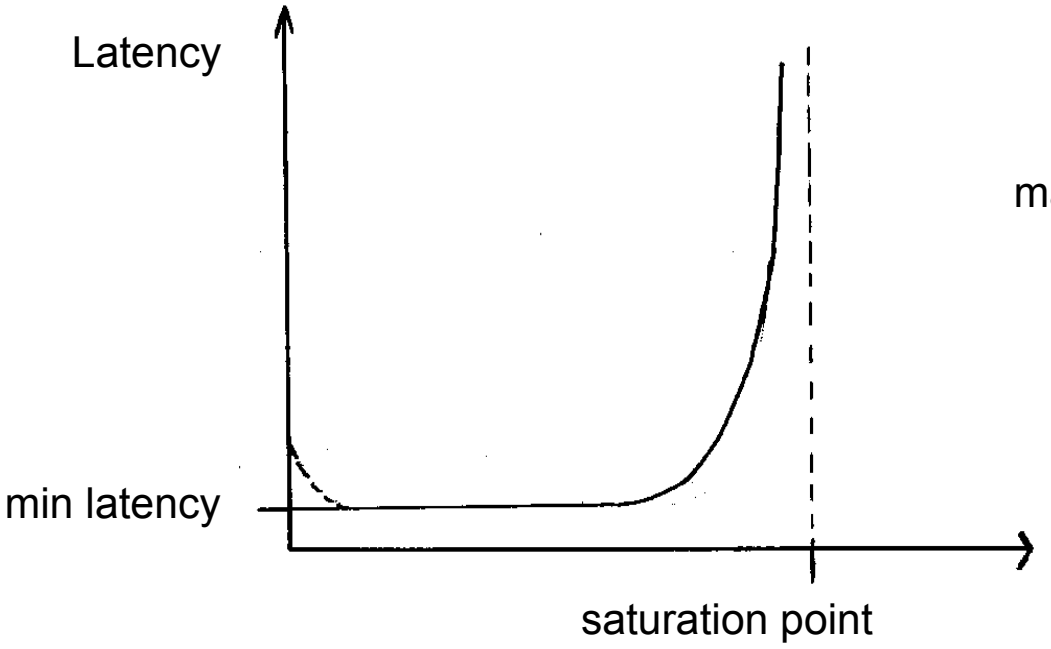
Profit = $1

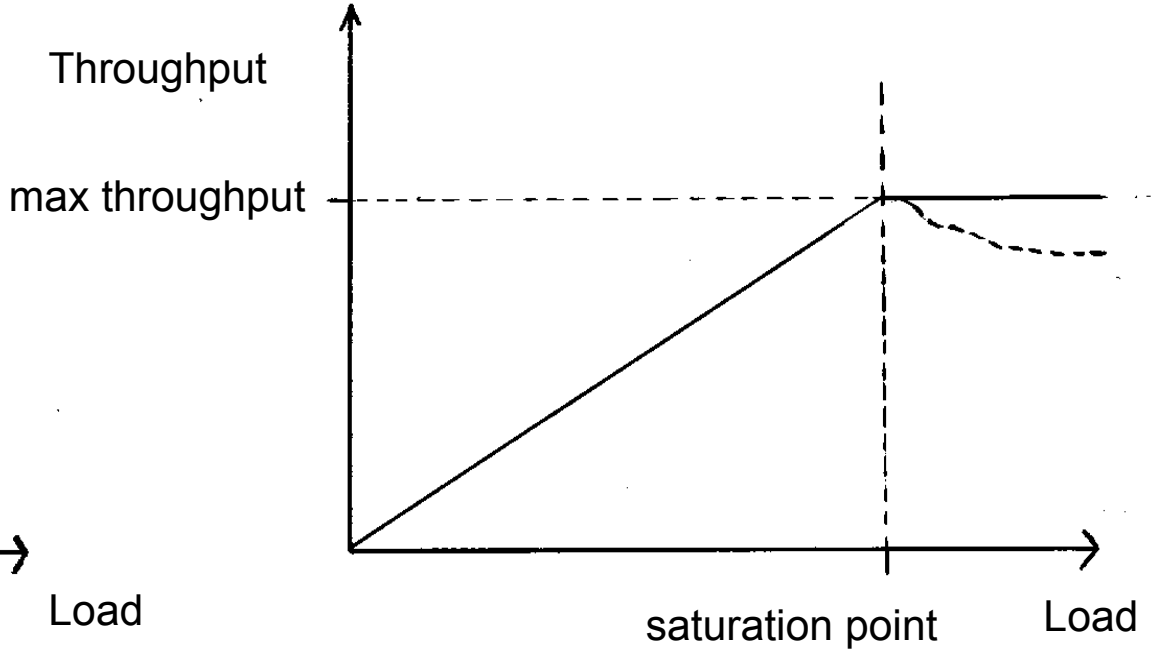
USD		EUR		GBP	
100^{1} 101^{4}	100^{2}	76^{2}	76^{3}	64^{3}	64^{4}

Origin

Wrote a repository driven engine initially used with BizTalk
What is the repository?
Looked a code generation to improve
Already one of the fastest engines around
GC identified as major hurdle
.Net initiative to remove

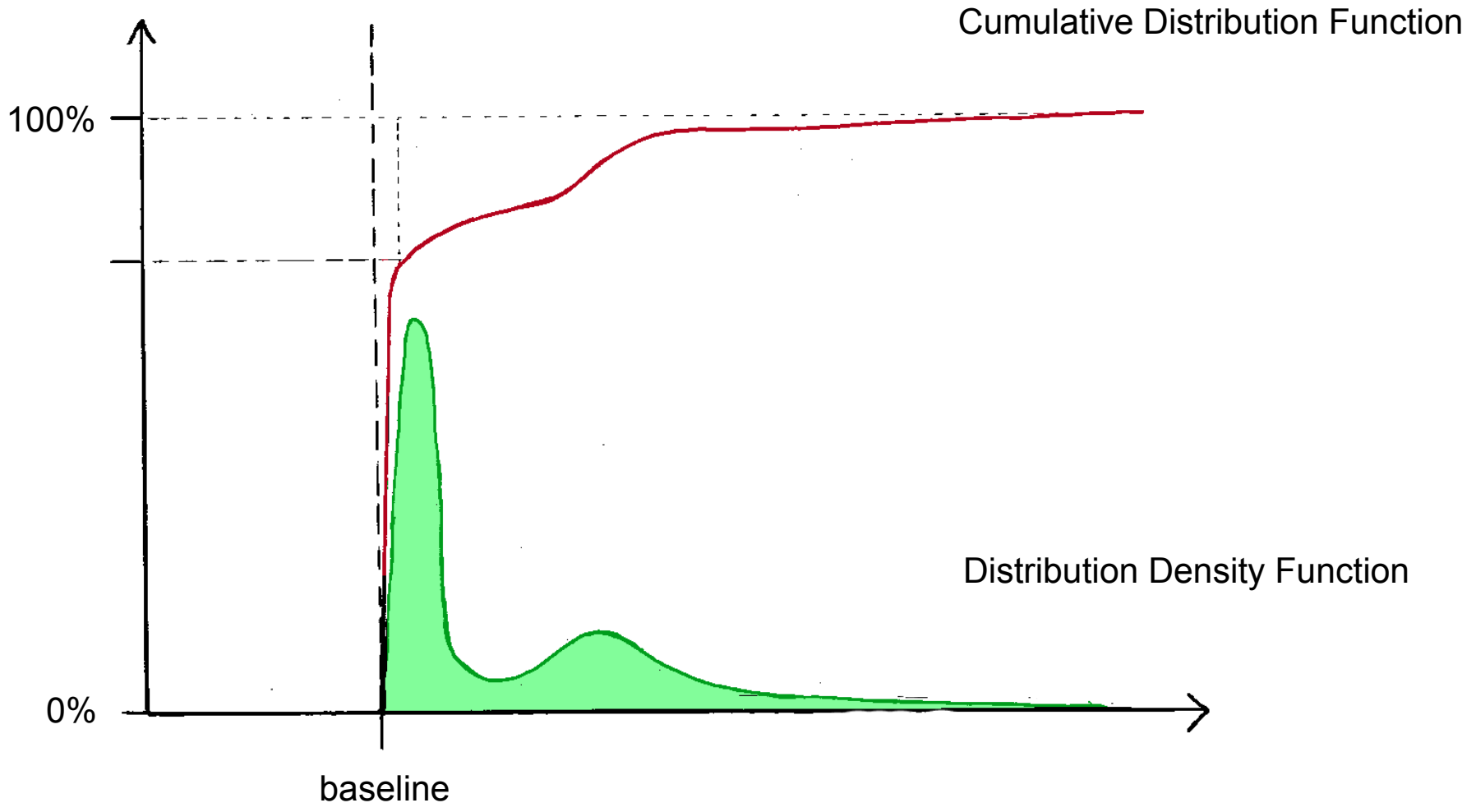
Latency vs Load and Throughput vs Load





If you want *good latency*,
you must have *excess capacity*.

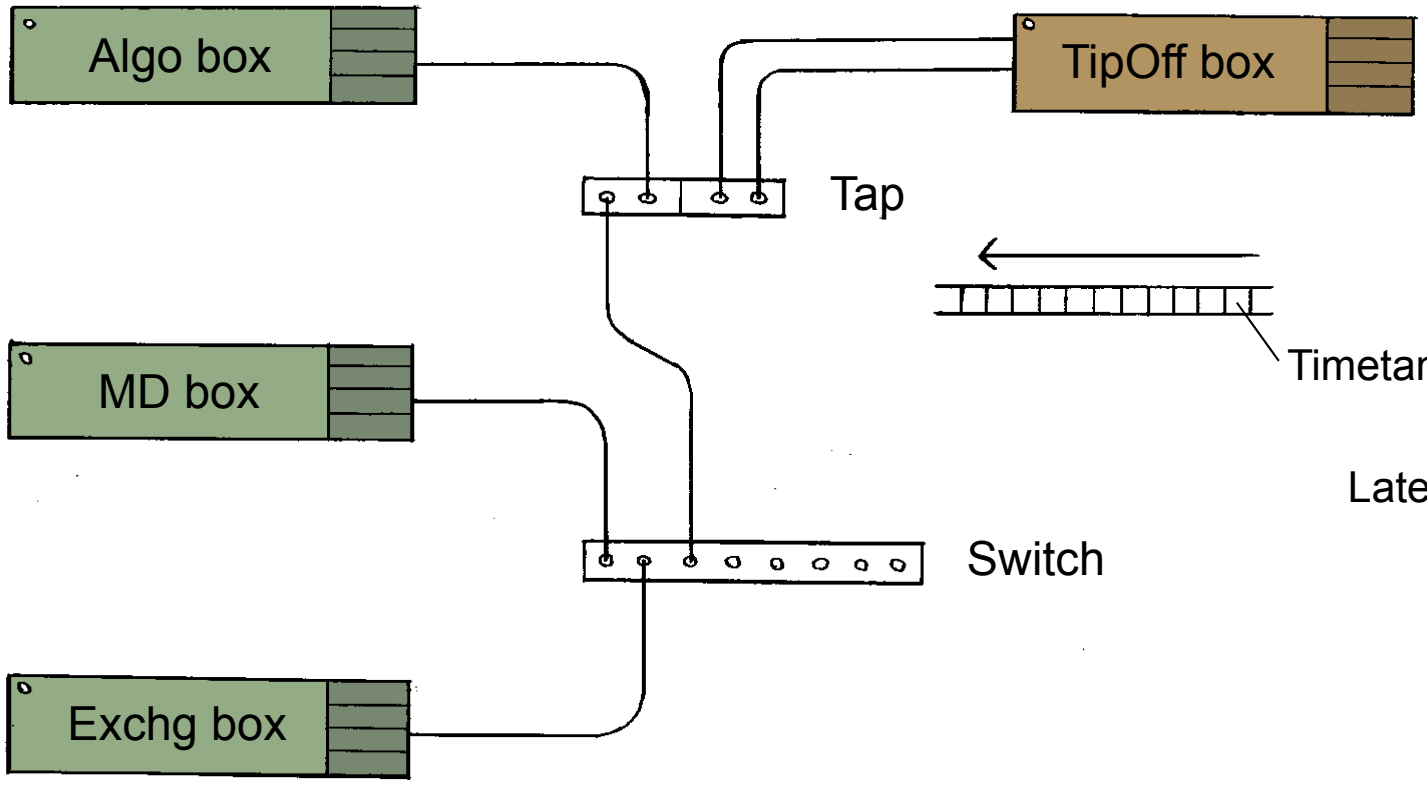
Distribution of Latencies



Lower the baseline.
Tighten the tail.

The 'Algo' Trading Test

fix0_n Market Data and Executions
fix1_n Orders



Timetamp first bit (+ constant delay)

Latency = T2 - T1
= (T2 + C) - (T1 + C)

5 'stocks'
5 ticks/cycle

1001.000 -> BUY 1
1002.000 -> BUY 2
...
1001.001
...
1001.005

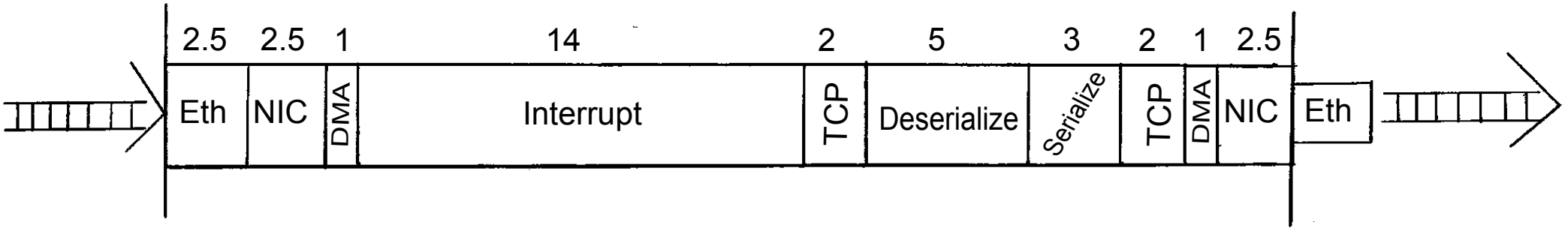
2001.000 -> BUY 1
...

Why?

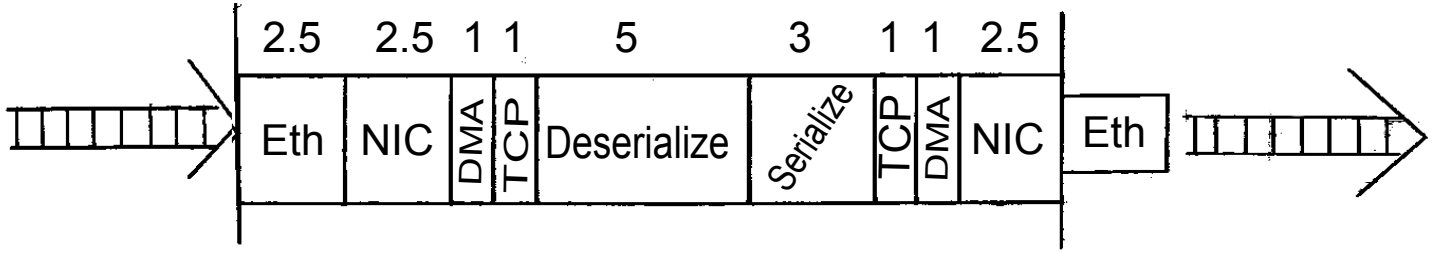
Order and Exec Report can be matched up by Client Order Id.
Market Data and Order have no built in correlation.

End to End Time Budget

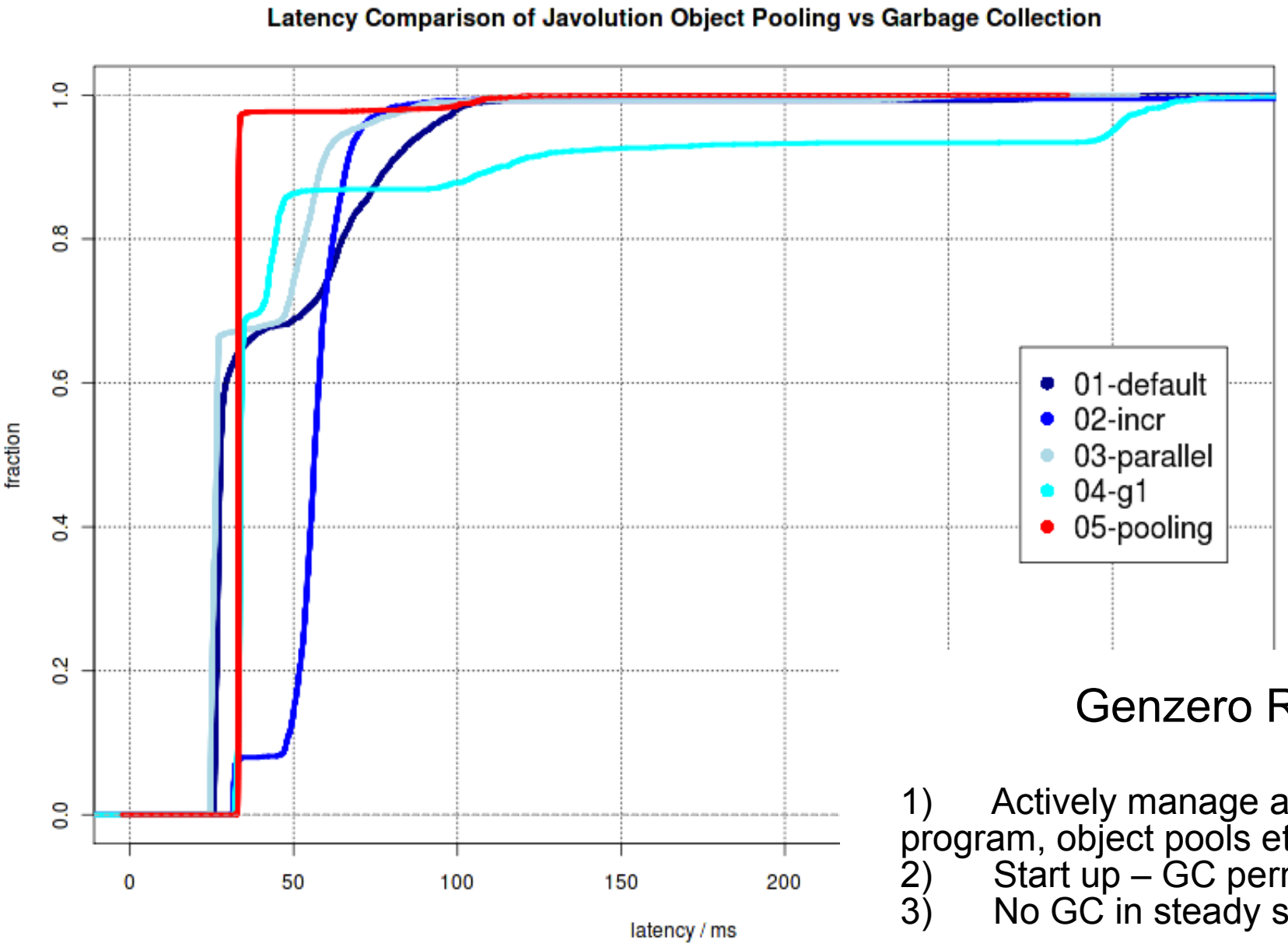
1G on 2.8 GHz Nehalem, built in NIC



1G on 2.8 GHz, SolarFlare and 'open onload'



Jitter and Garbage Free Code



JavaWorld article on Javolution

http://tinyurl.com/69m7xz

Compute FFT of Complex numbers. Requires lot of intermediate values; a source of garbage.

Genzero Rules:

1) Actively manage any resources used in program, object pools etc
2) Start up – GC permitted
3) No GC in steady state phase of program

FIX is ASCII

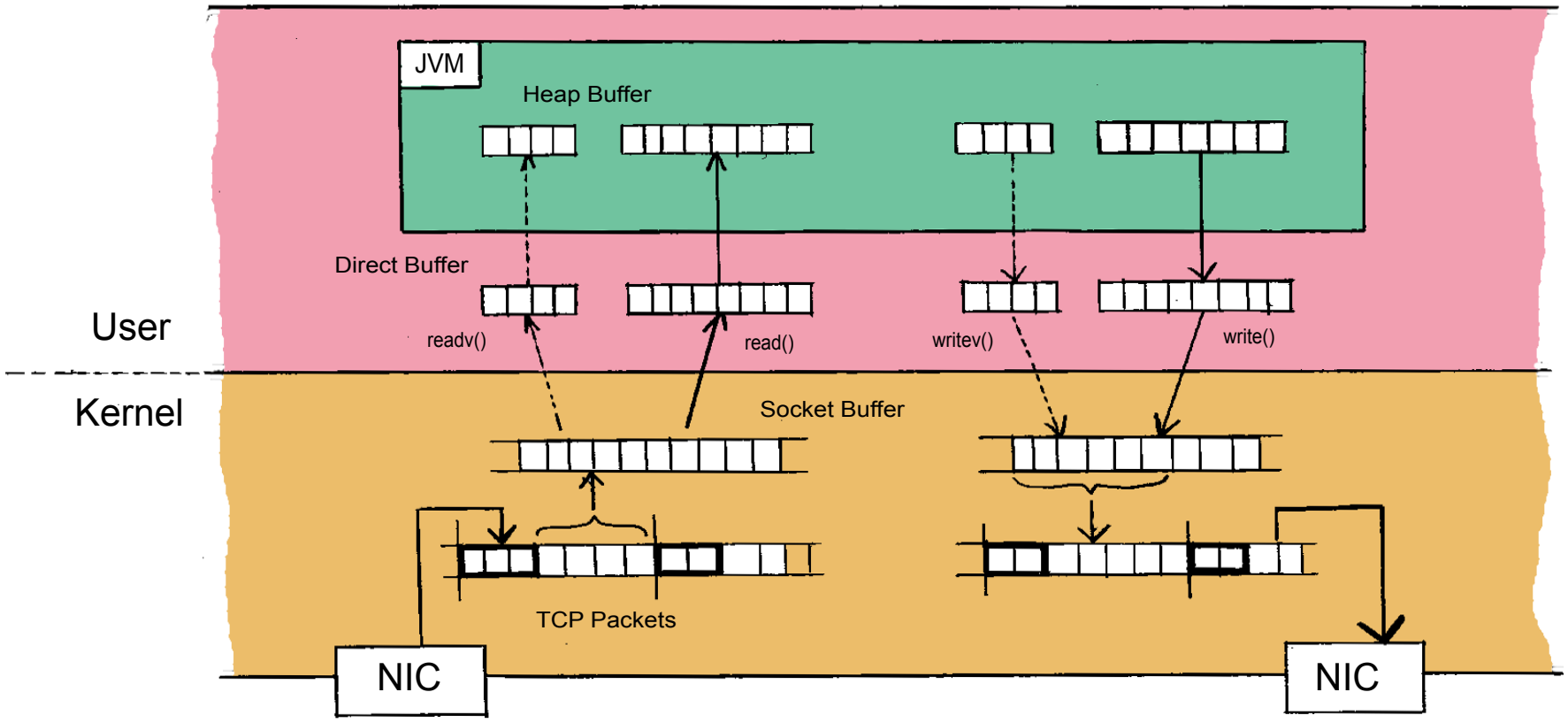
8=FIX.4.2 | 9=192 | 35=X | 49=FEED | 56=ALGO | 34=7 | 52=20120308-00:51:52.303 | 262=subscribe:A000 | 268=2 | 279=1 | 269=1 | 278=23 | 55=A000 | 270=0.001 | 271=1000 | 346=1 | 290=1 | 279=1 | 269=0 | 278=24 | 55=A000 | 270=0.101 | 271=1000 | 346=1 | 290=1 | 10=186 |

Subtract ASCII '0'
Multiply by 10

Divide by 10, take modulo.
Add ASCII '0'

Avoid using java.lang.String for string processing.

Zero Copy I/O, how real is it?

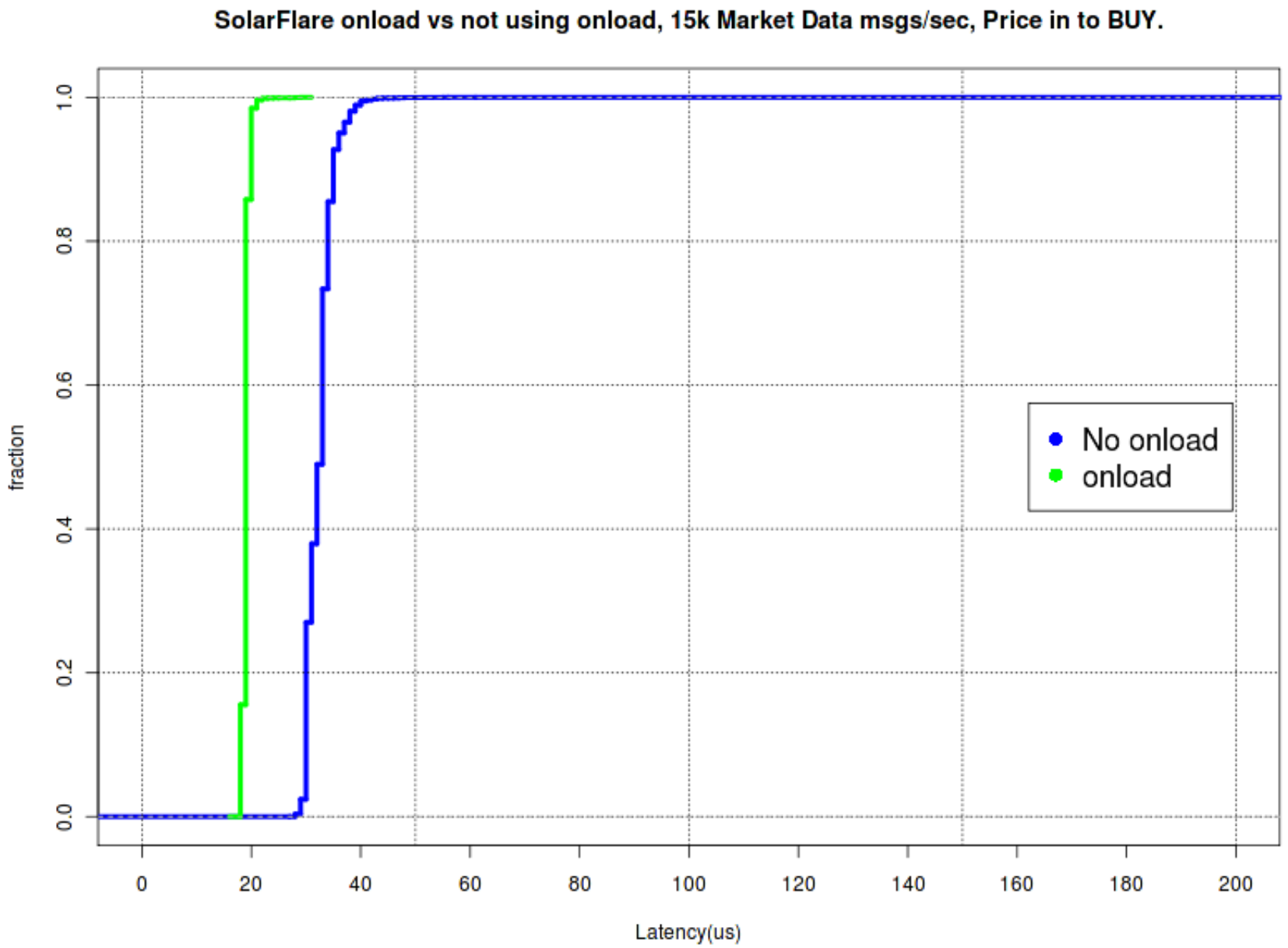


Java nio provides API for zero-copy vectored I/O.

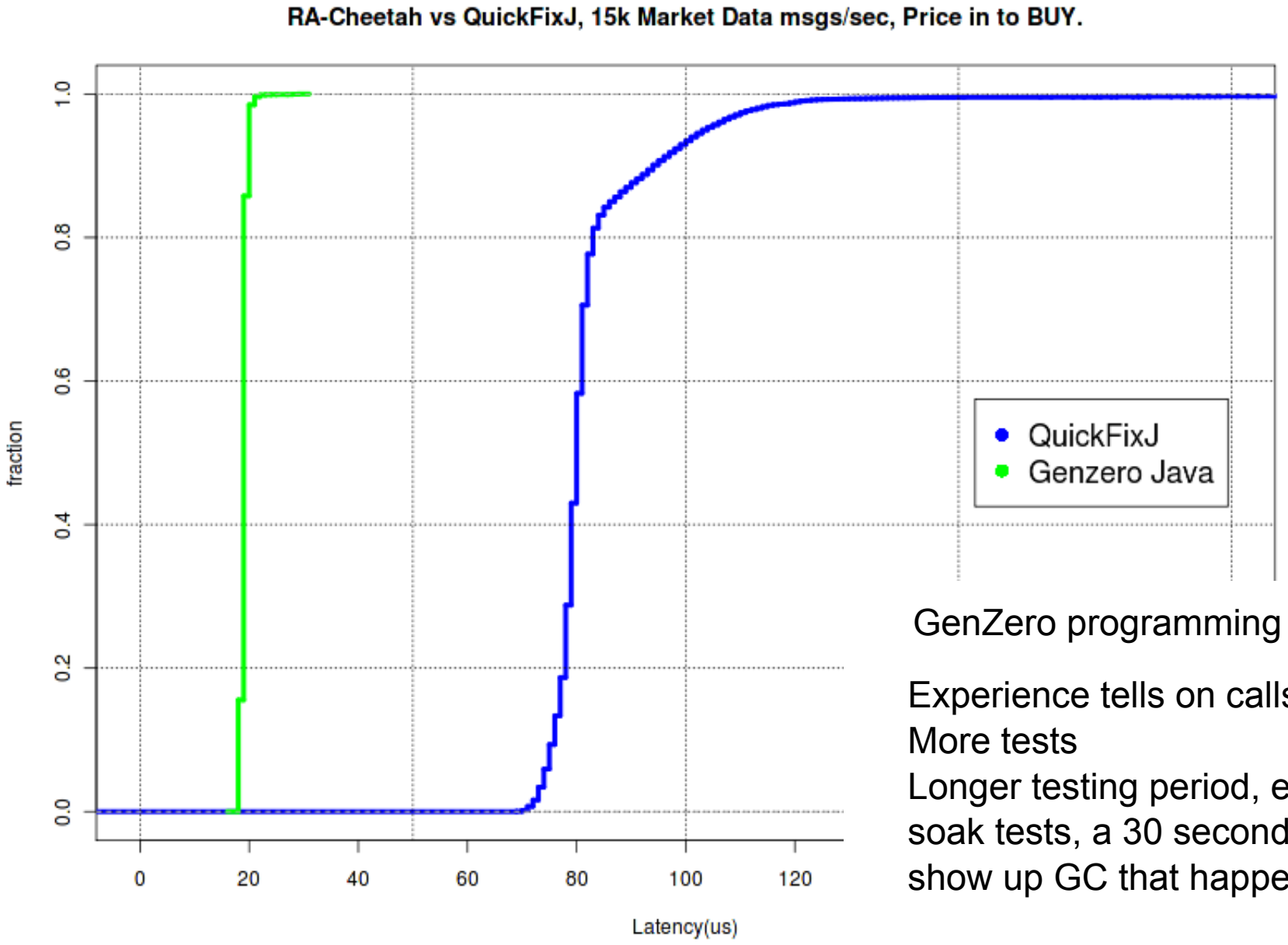
There may be copying.

Access to byte[] vs getByte()/setByte().

SolarFlare Effect of 'onload' (polling)



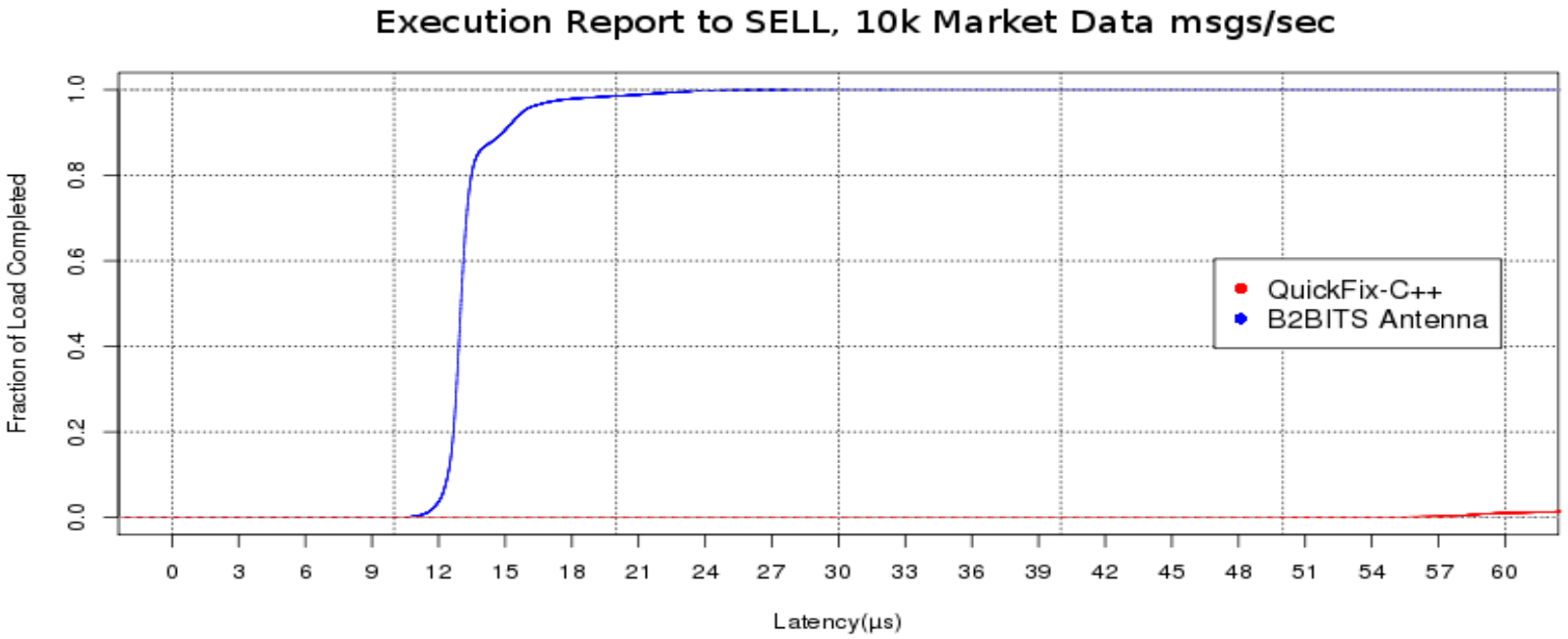
RA Cheetah vs QuickFixJ

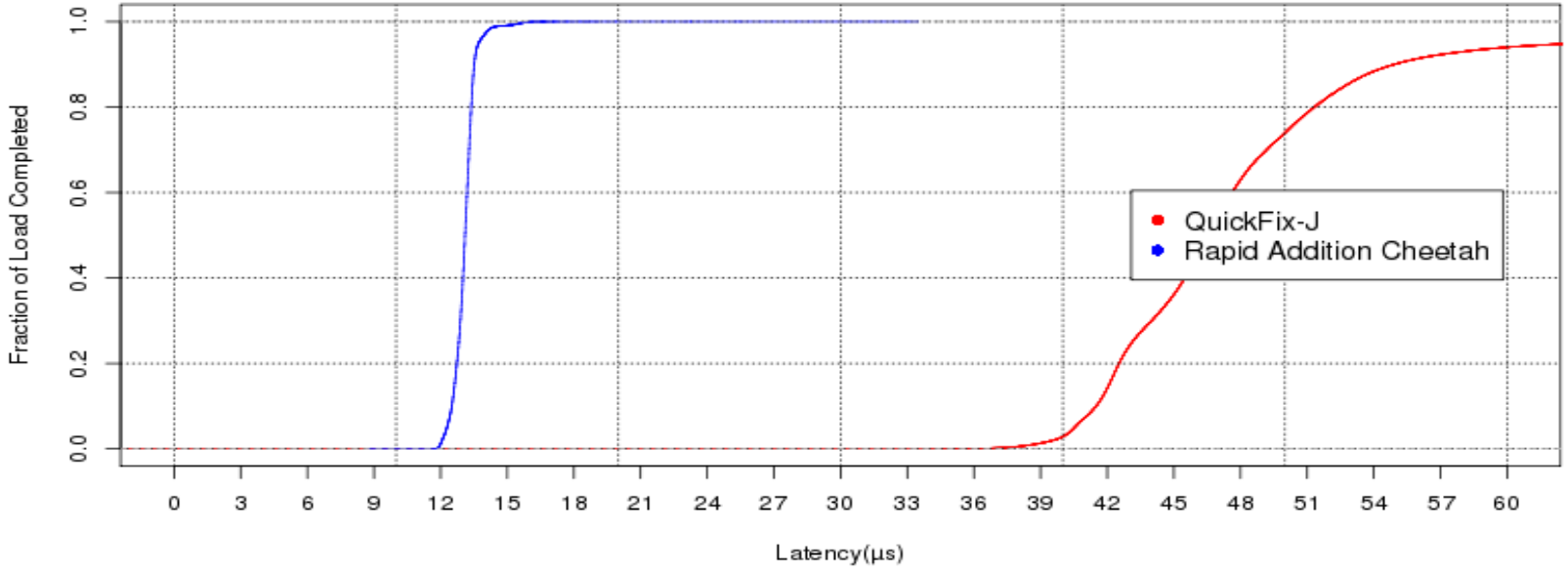


GenZero programming differences:

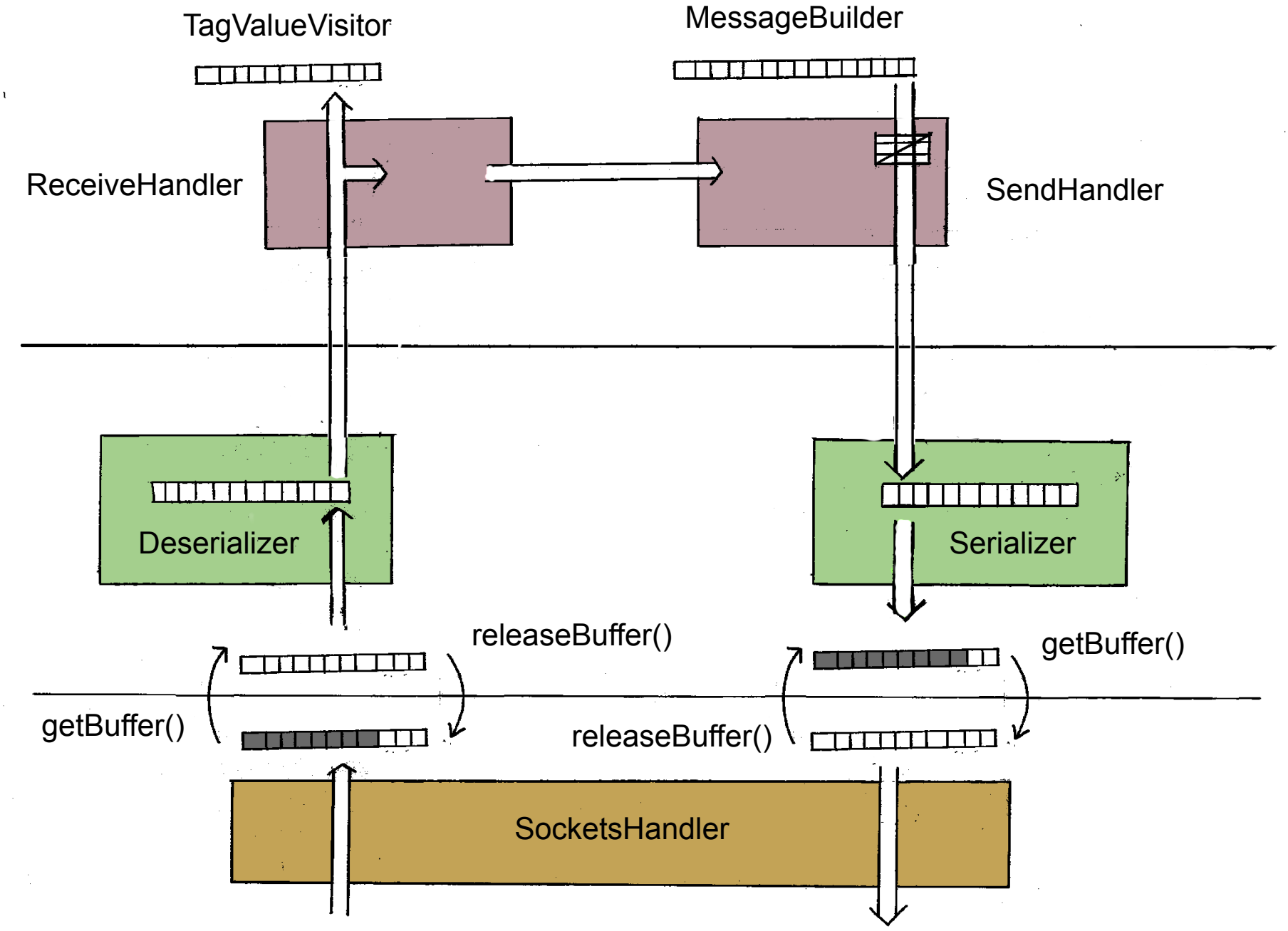
Experience tells on calls to avoid etc
More tests
Longer testing period, effectively soak tests, a 30 second test will not show up GC that happens every 120

Intel Testing, 10G SolarFlare and Dell Everest

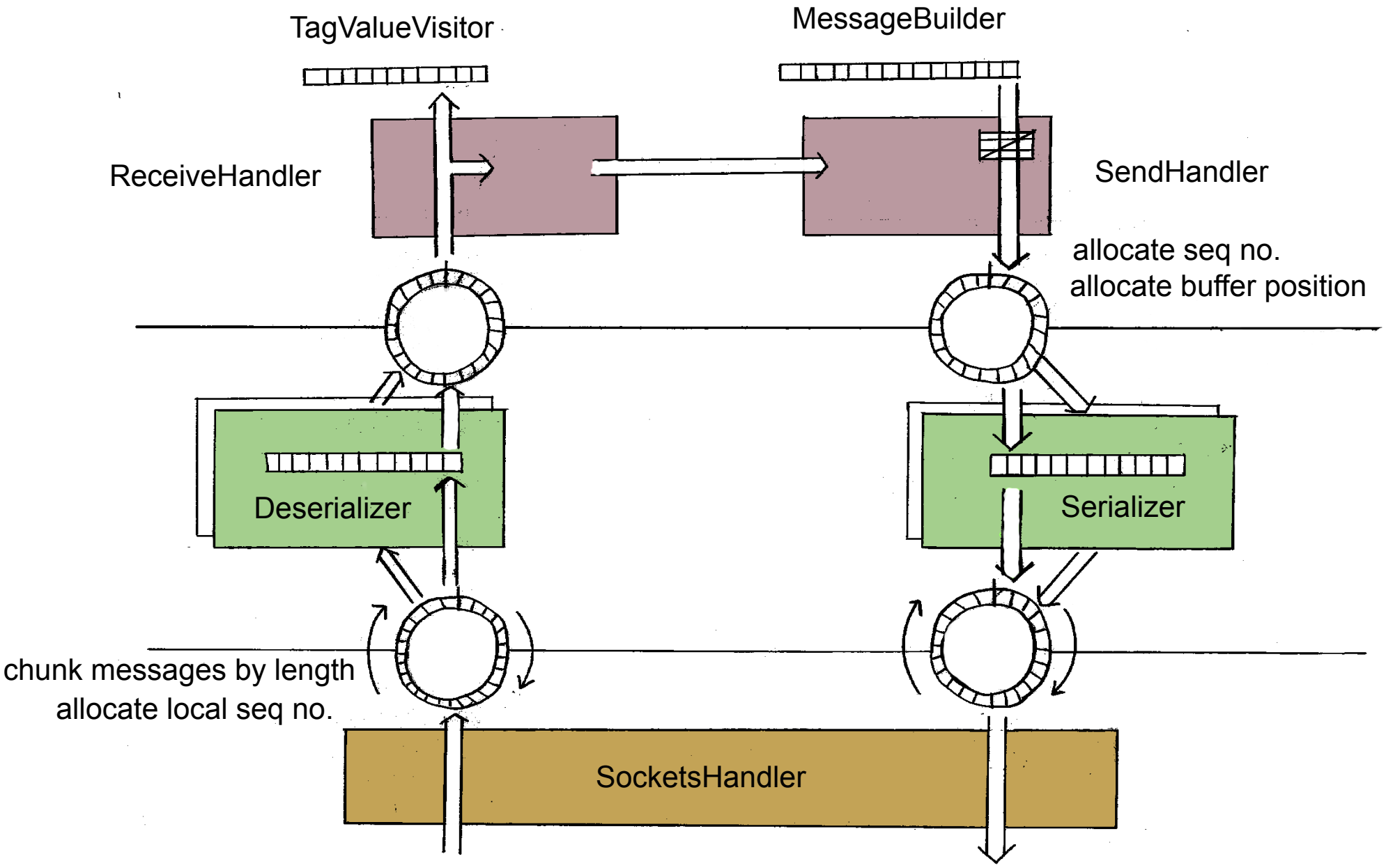




FIX Engine Overview



FIX Engine - With Internal Queues



FIX Engine - With Hardware Implementation