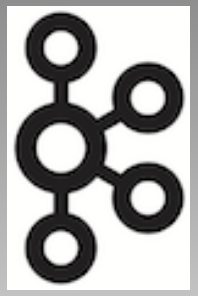


High throughput kafka for science

Testing Kafka’s limits for science

J Wyngaard, PhD
wyngaard@jpl.nasa.gov

OUTLINE

- Streaming Science Data
- Benchmark Context
- Tests and Results
- Conclusions

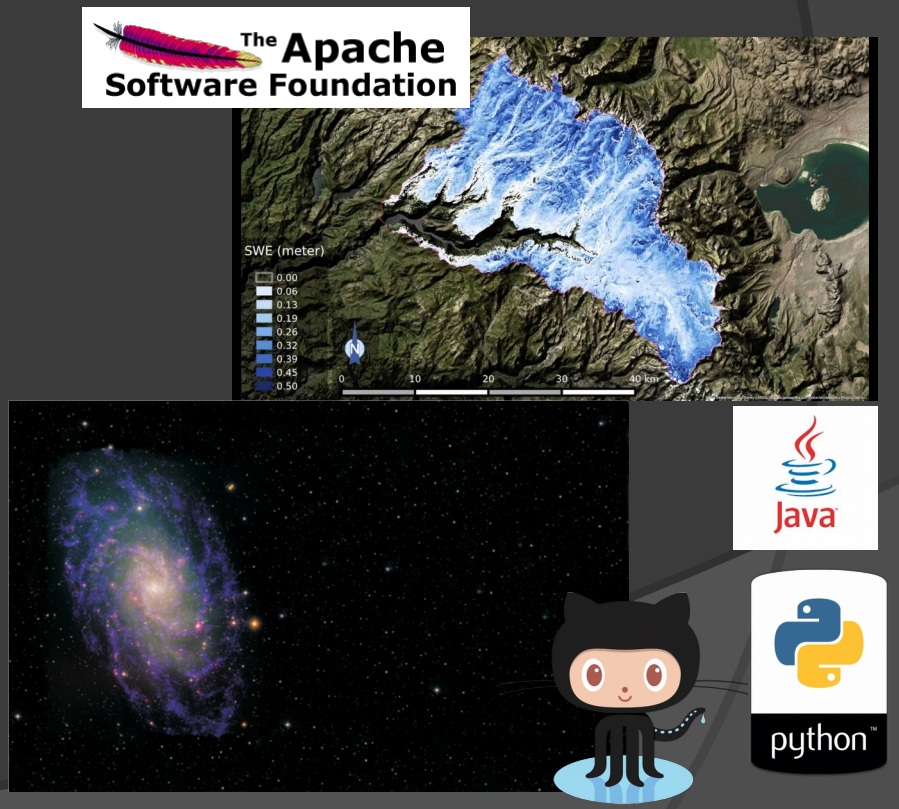
- Streaming Science Data
- Benchmark Context
- Tests and Results
- Conclusions

Streaming Science Data

SOODT, Kafka, Science data streams

DIA GROUP

- Using open source tools extensively, to enable JPL scientists to handle their big data.
  - Apache OODT
  - Apache Tika
  - Apache Hadoop
  - Apache Kafka
  - Apache Mesos
  - Apache Spark
  - ...so many more...

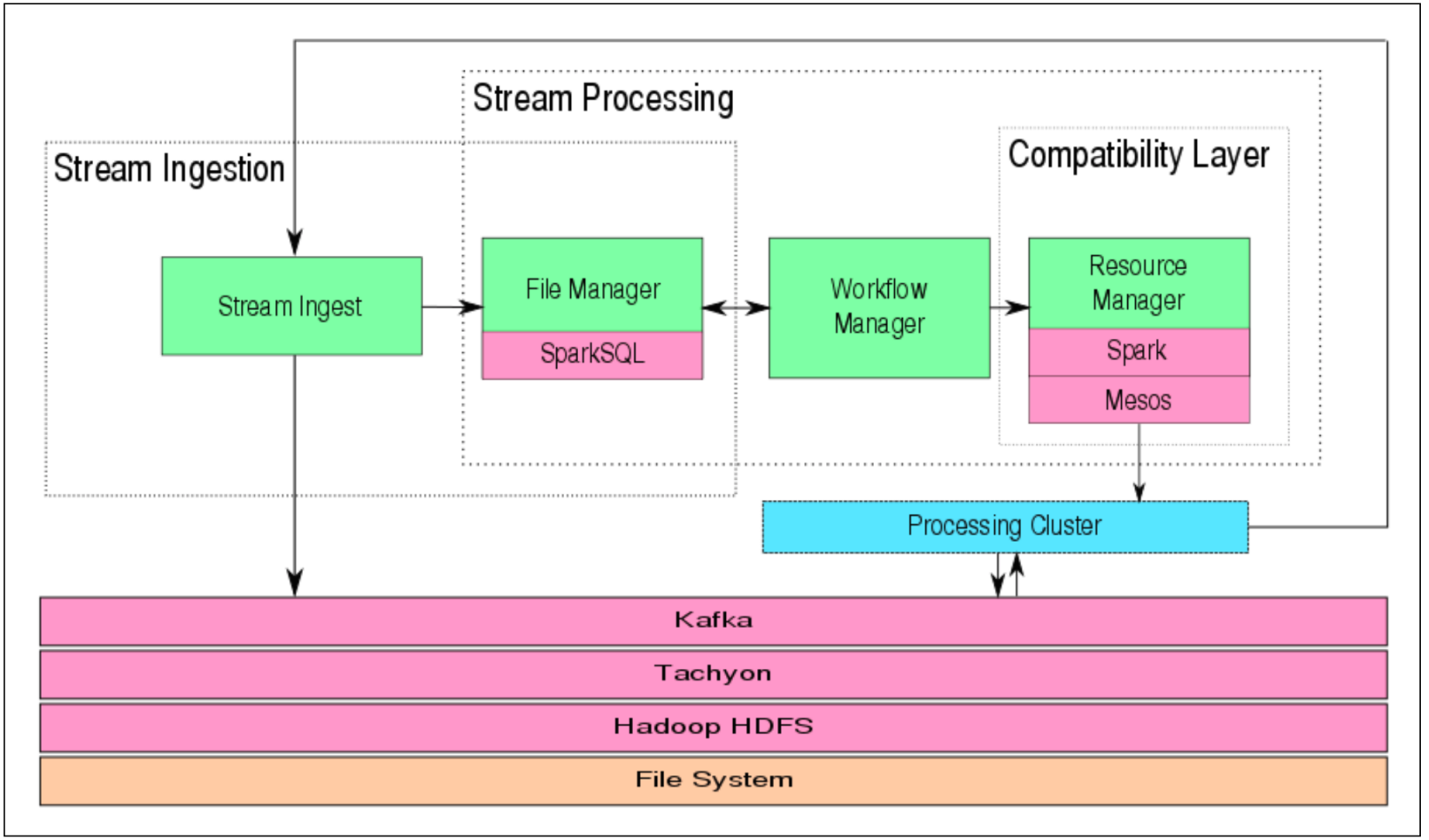


SCIENCE DATA

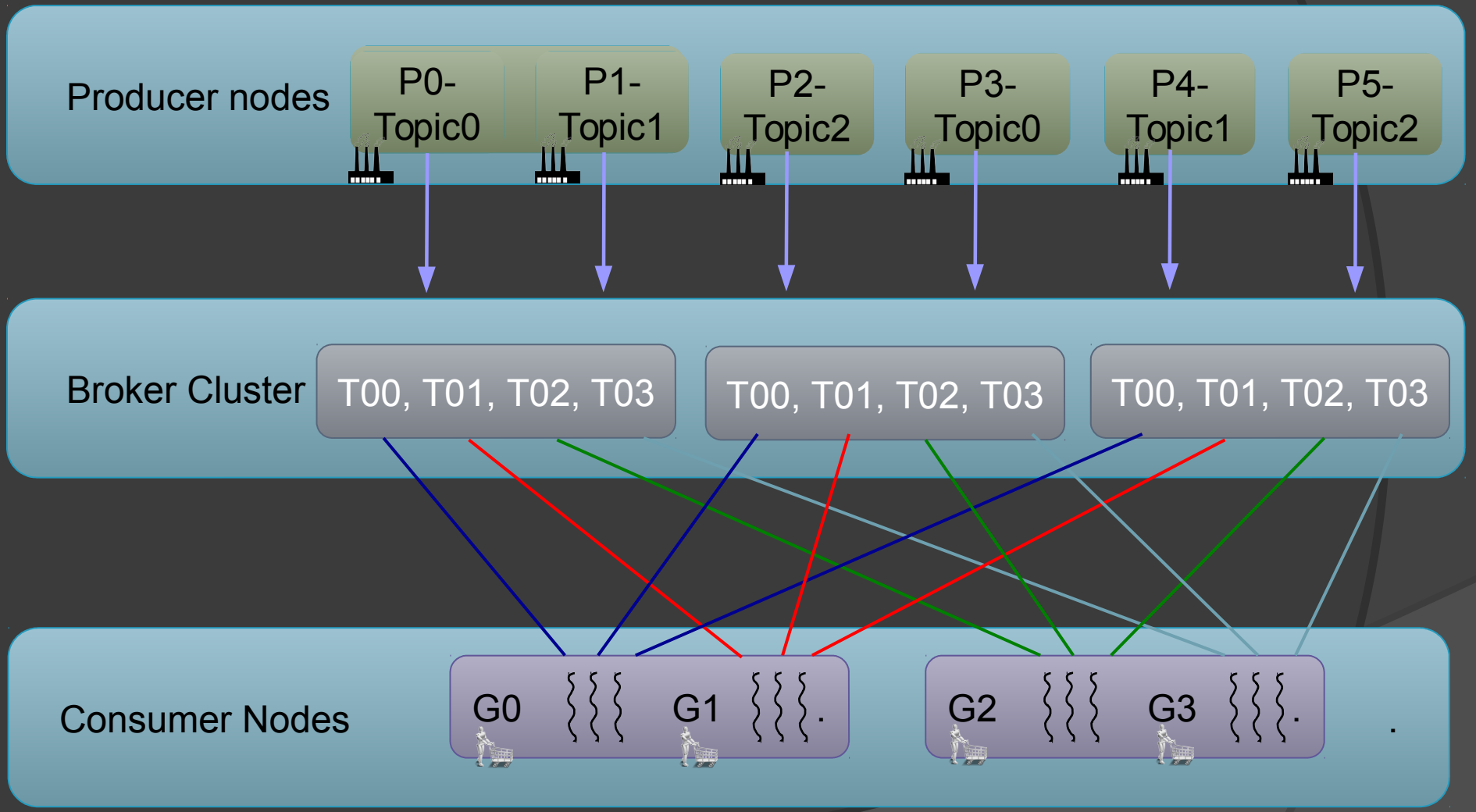
- Earth Science
  - Satellite data ~5GB/day
- Radio Astronomy
  - Antenna arrays ~4Tbps >>1K 10Gbps
- Airborne missions
  - ~5GB files, 0.5TB per flight
- Bioinformatics

STREAMING SOODT

Streaming OODT Conceptual Layout



APACHE KAFKA



10G X 1024 - ?

- Low Frequency Apature array:
  - 0.25M antennas
    - 1024 stations
      - 16 Processing modules
- = 4Tbps from 1024 stations at 10Gbps each

Artists' impression of LFAA, SKA image

https://www.skatelescope.org/multimedia/image/low-frequency-array-ska-wide-field/

- Streaming Science Data
- Benchmark Context
- Tests and Results
- Conclusions

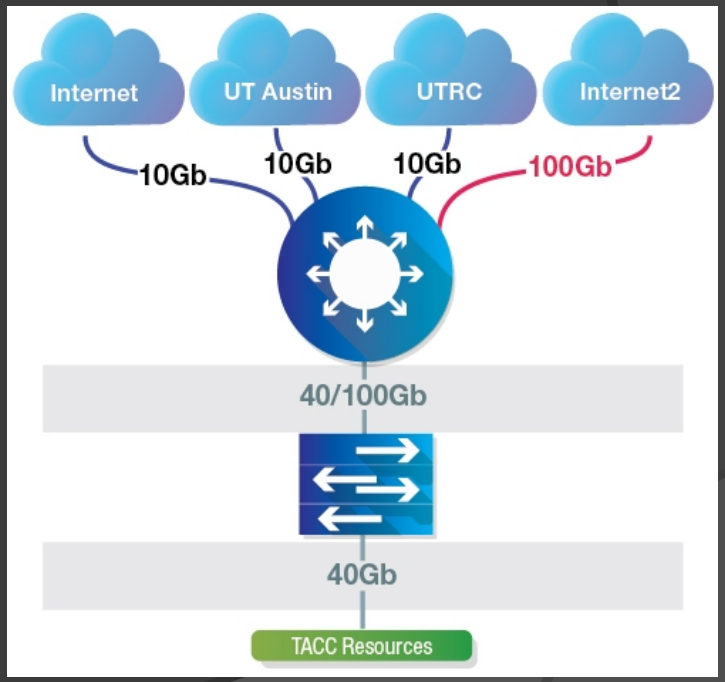
Benchmark Context

Reality check – kafka was not designed for this

TACC WRANGLER

- Primary system
  - 96 nodes
    - 24Core Haswells
    - 128GB RAM
  - Infniband FDR and 40 Gb/s Ethernet connectivity.
  - 0.5PB NAND Flash
    - 1 Tbps
    - >200 million IOPS.
- A 24 node replicate cluster resides at University of Indiana, connected by a 100 Gb/s link

TACC
TEXAS ADVANCED COMPUTING CENTER



"LAZY" BENCHMARKING

- "Lazy" being:
  - Off-the shelf cheap hardware
  - Untuned default configuration

https://engineering.linkedin.com/kafka/benchmarking-apache-kafka-2-million-writes-second-three-cheap-machines

6 CHEAP MACHINES

- OTS benchmark
  - 6 core 2.5GHz Xeons
  - ~ 100 IOPS harddrives
  - 1Gb Ethernet
- Wrangler nodes
  - 2x 12core 2.5GHz Xeons
  - >200 IOPS flash
  - 128GB RAM
  - 40Gb Ethernet

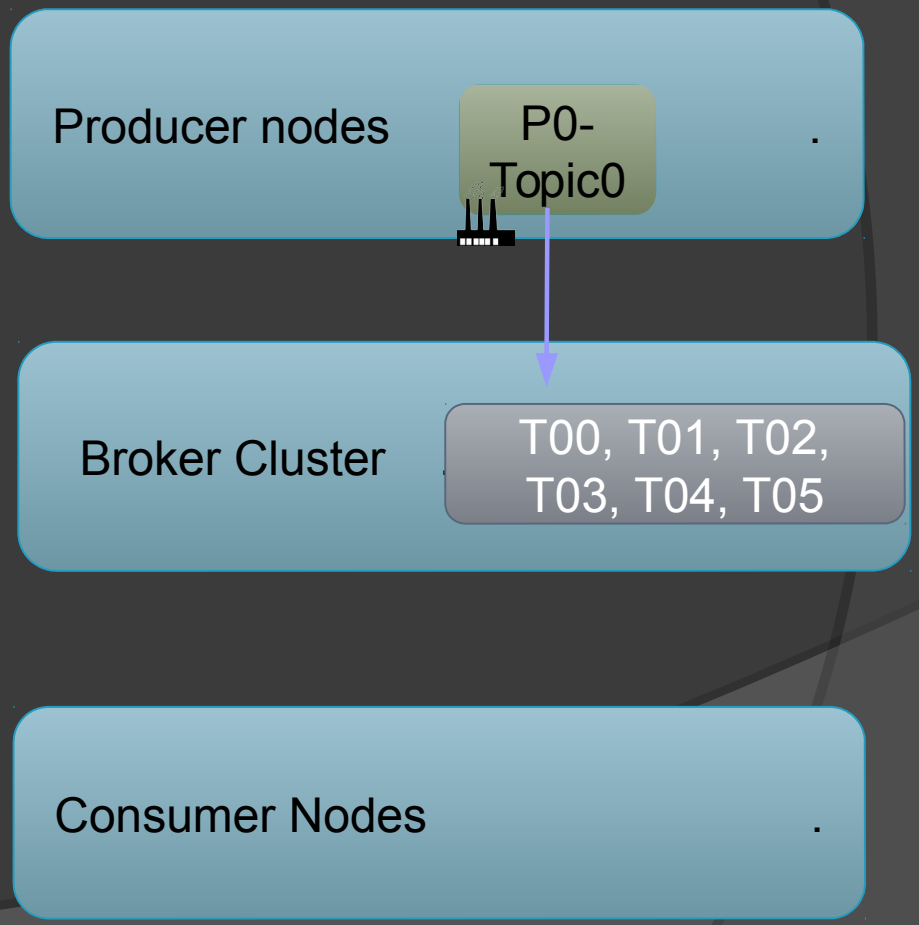
“LAZY” CONFIGURATION

- Kafka trunk 0.8.1
- New producer
- Default configurations
- Small messages

- Setup
  - 3 Broker nodes
  - 3 Zookeeper, Consumer, Producer nodes
- Kafka builtin performance tools

STRAIGHTLINE "LAZY" SPEED TEST

- 1 Producer
- 0 Consumer
- 1 Topic
- 6 partition
- 1 replicates (i.e 0)
- 50M 100B messages
  (small for worst case)



STRAIGHTLINE "LAZY" SPEED TEST

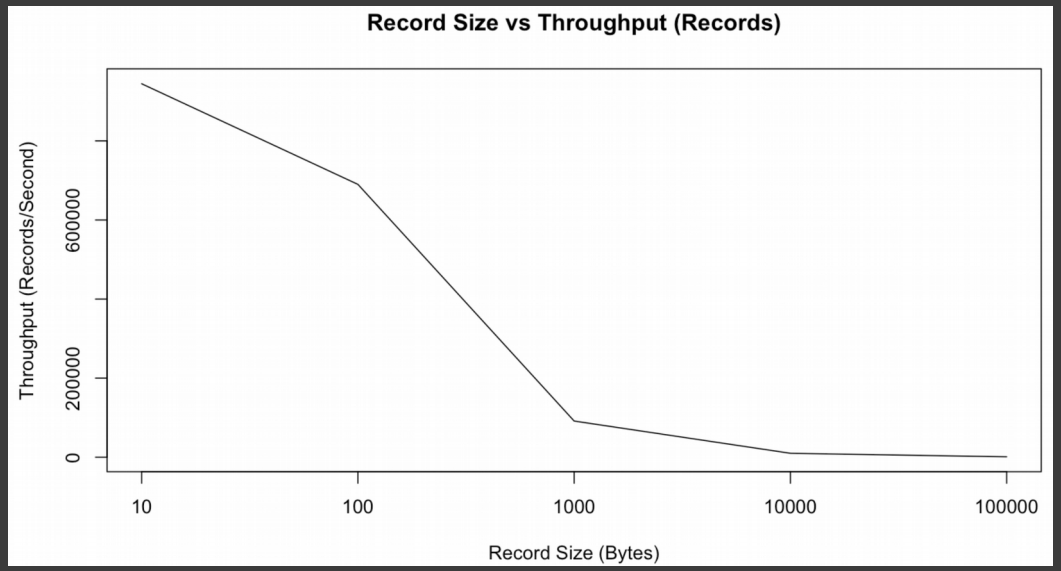
- 1 Producer
- 0 Consumer
- 1 Topic
- 6 partition
- 1 replicates (i.e 0)
- 50M 100B messages
  (small for worst case)

6 cheap machines
78.3MB/s*
(0.6Gbps)

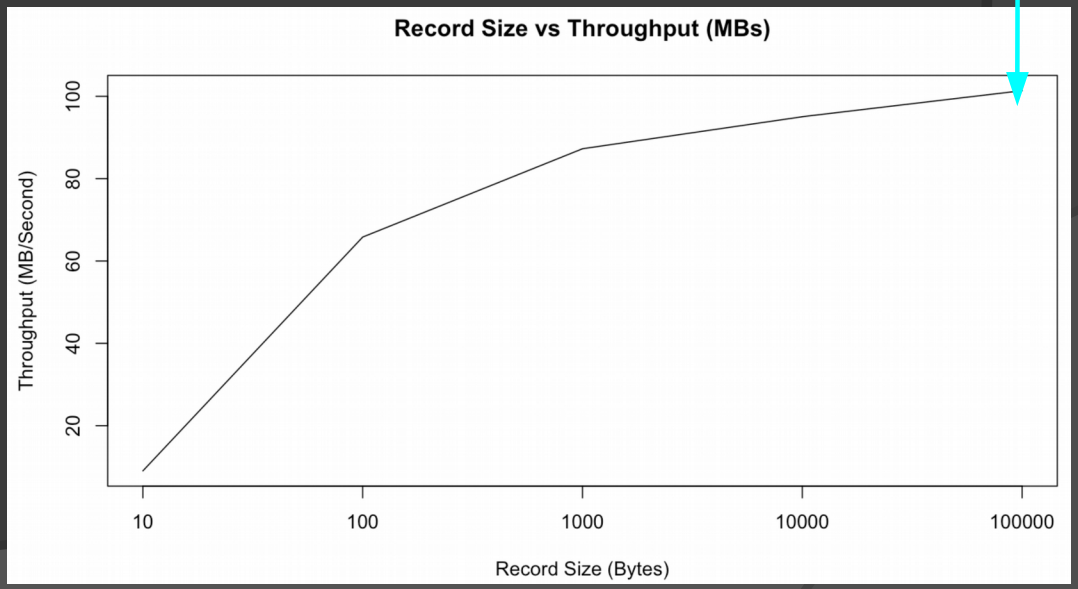
Wrangler
170.27 MB/sec*
(1.3Gbps)

*Network overhead not accounted for

Δ MESSAGE SIZE



~100MBps at 100KB message sizesec



OTHER PARAMETER IMPACTS

- Replication:
  - Single producer thread, 3x replication, 1 partition
    - Asynchronous
      - 0.59Gbps
    - Synchronous
      - 0.31 Gbps
- Parallelism:
  - Three producers, 3x asynchronous replication
    - Independant machines
      - 1.51 MB/sec < 3*0.59 = 1.77

Reference straight line producer speed:
0.61Gbps

- Streaming Science Data
- Benchmark Setup
- Wrangler Performance
- Conclusions

Wrangler Performance

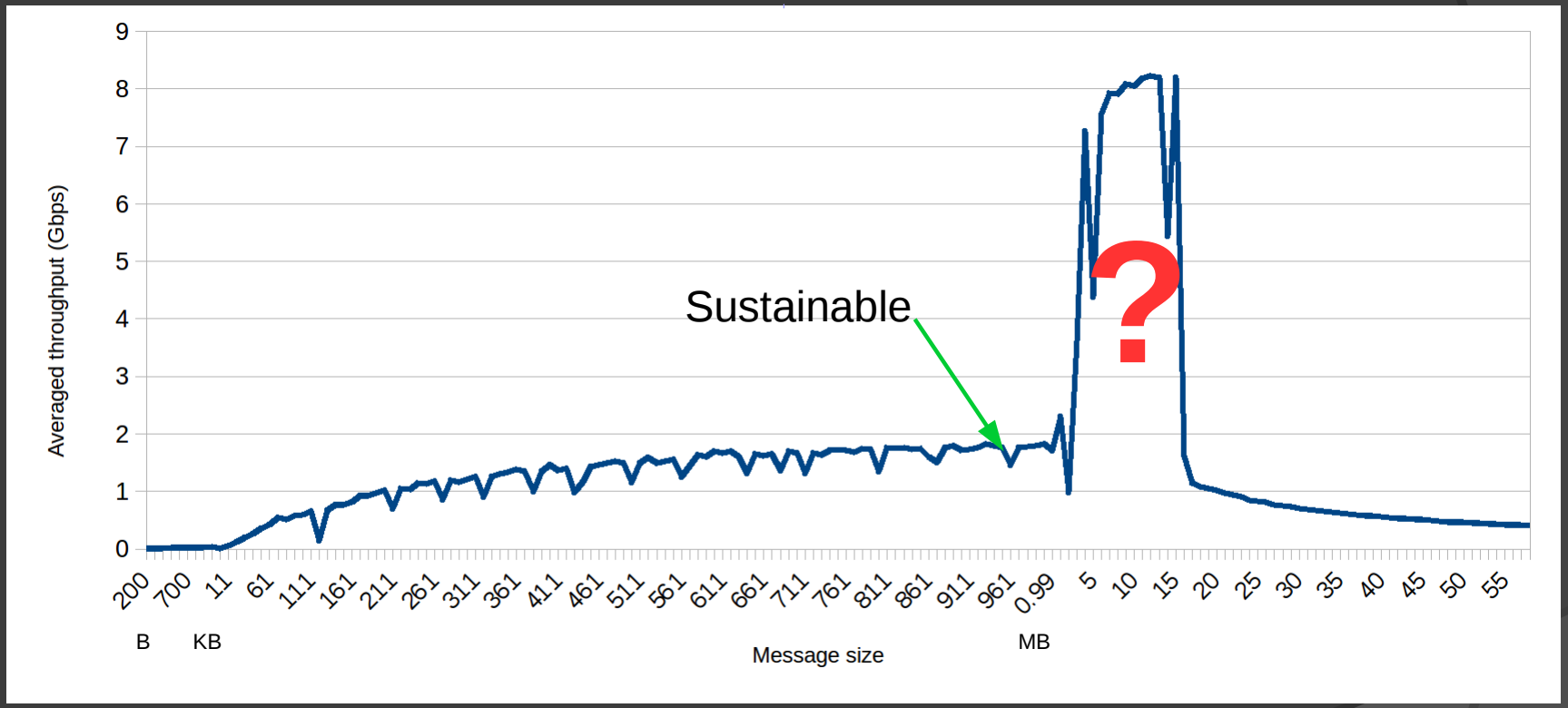
Limits

TARGETTING 10G

- 40x networks speed
- 4x core counts
- 2x IOPS
- 128x RAM

- Starting point
  - Bigger messages
  - No replication
  - In node paralleism
  - Big Buffers
  - Large Batches

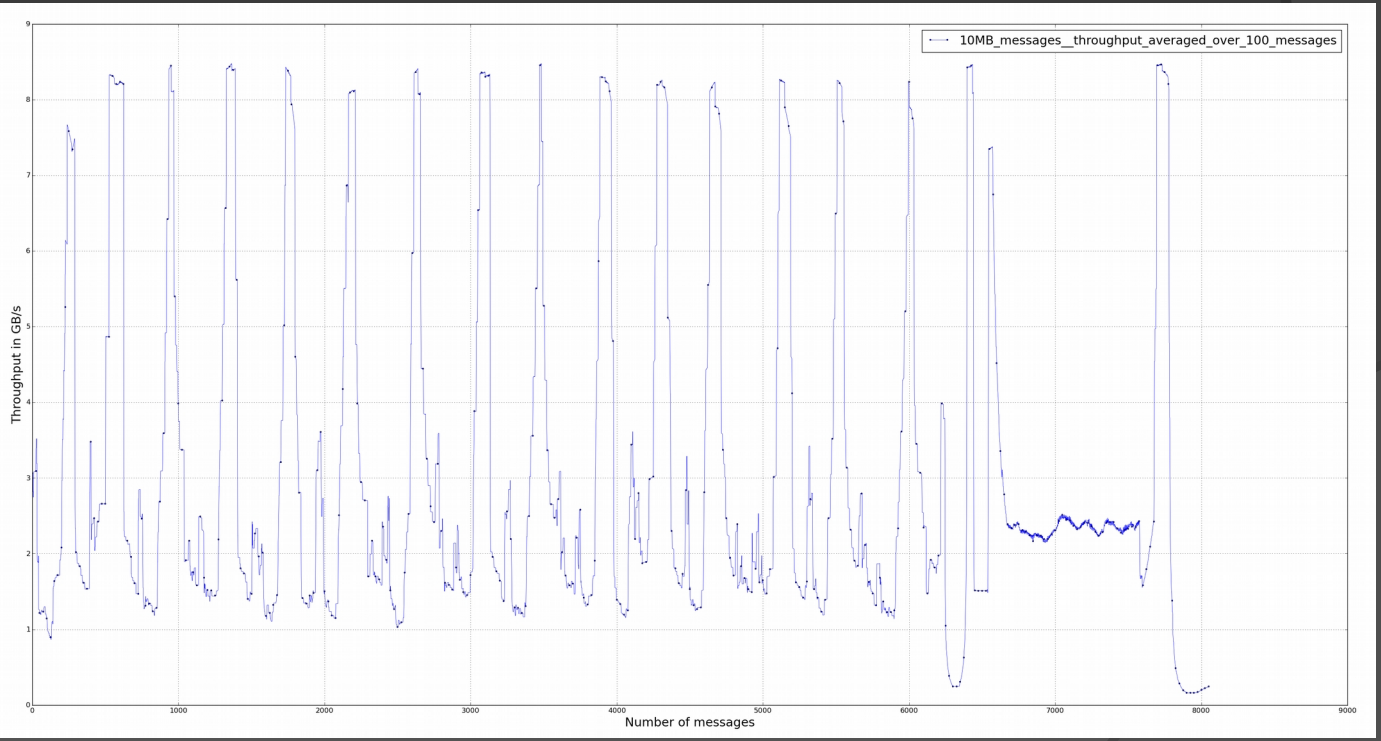
Δ MESSAGE SIZE



Averaged throughput over changing message size

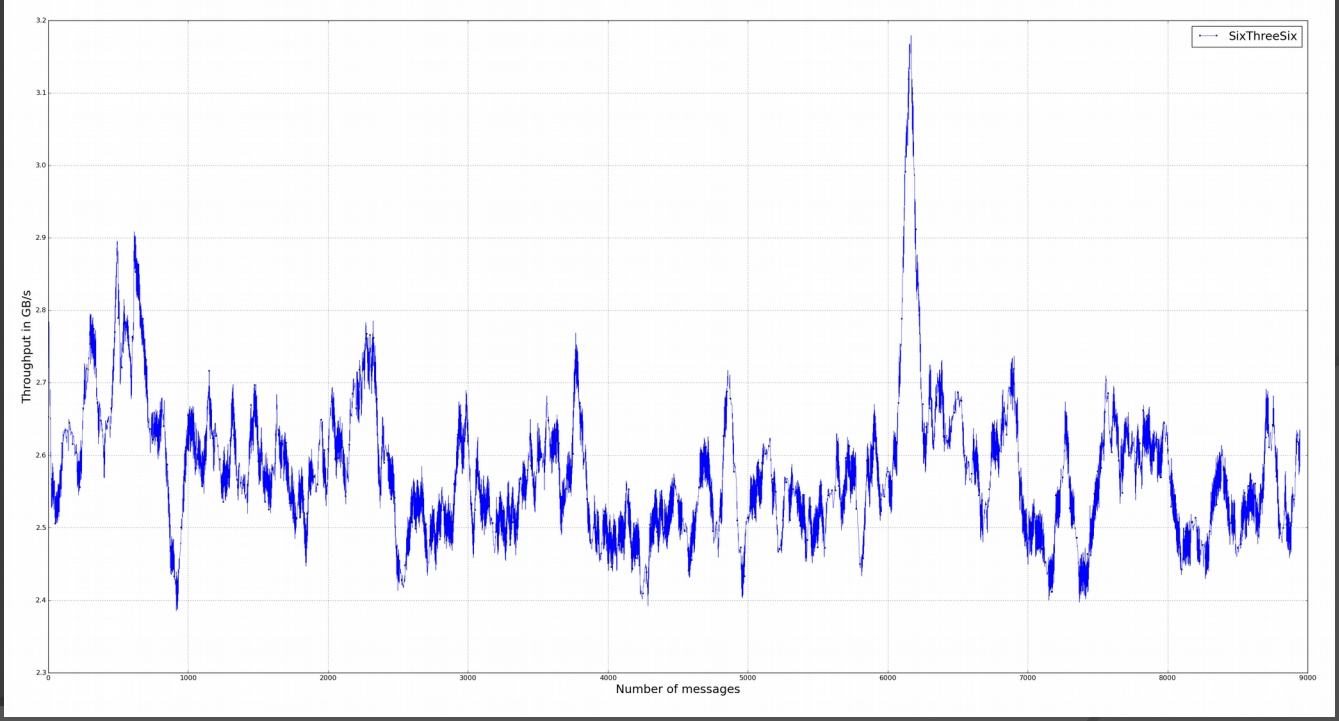
PARTITIONS

- 3 producers, 1 topic, asynchronous, 3 consumer threads
  - Averager 6.49Gbps (8000 messages)



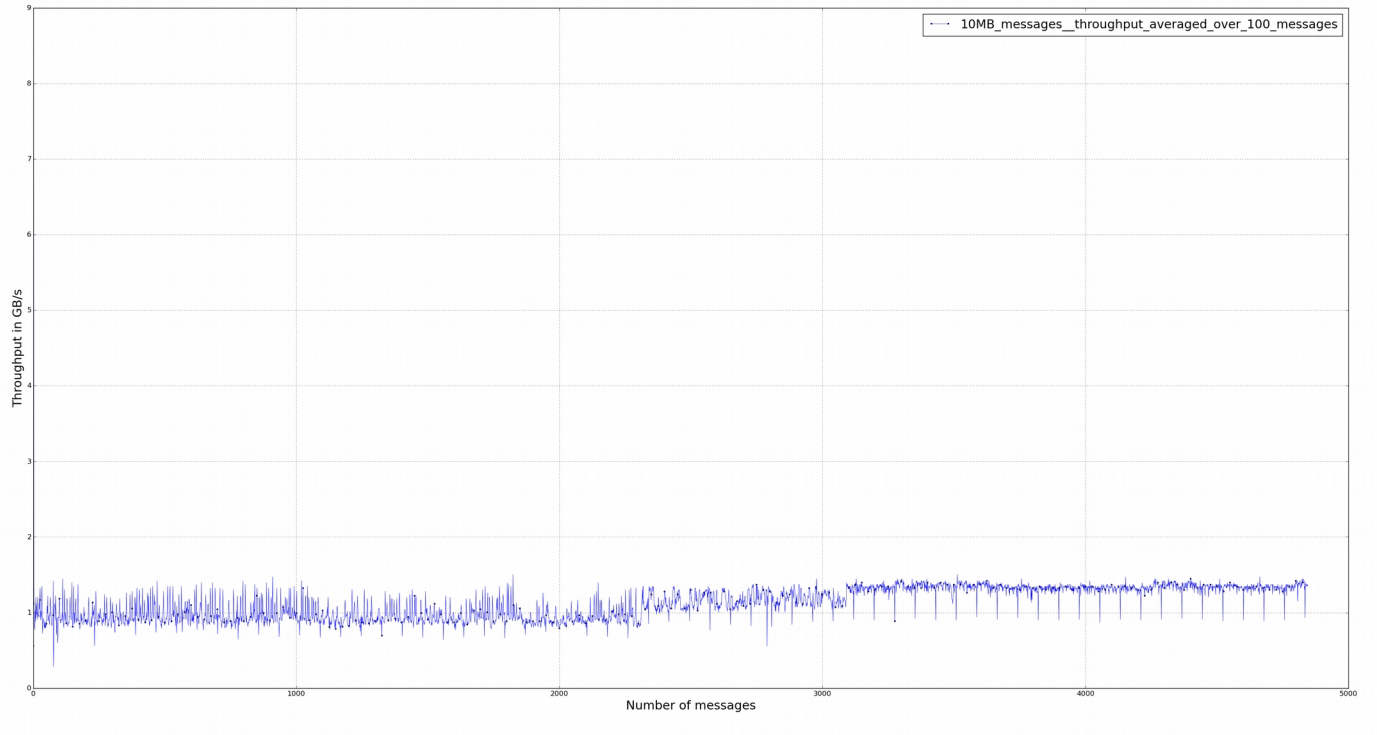
PARTITIONS

- 6 producers, 1 topic, asynchronous, 6 consumer threads
  - Averager 2.6Gbps (8000 messages)



PARTITIONS

- 6 producers, 1 topic, asynchronous, 6 consumer threads, and 6 brokers
  - Averager 1.2Gbps (8000 messages)



- Context
- TACC Wrangler Data Analysis System
- Benchmark Setup
- Tests and Results
- Conclusions

Conclusions

And where to from here

TARGETTING 10G

- Apparent optimum for a single node producer on this hardware:
  - ~10MB messages
  - 3 Producers matching 3 consumers/consumer trheads
- More brokers, producers, consumers are detremental
- 6.49Gbps < 10Gbps

ALTERNATIVE AVENUES

- Parallelism -multiple topics if this is tollerable
  - (Potential ordering and chunking overheads)
- In a shared file system environment perhaps the file pointers rather than files should be moved
  - (not suitable in many applications)
- Nothing to be gained in better hardware

HPC PRODUCTION CLUSTER ENVIRONMENT

- Pros
  - Shared files system
  - tmpfs
  - Scale
- Cons:
  - User space installs only
  - SLURM
    - Idev
    - Job times-out loosing configurations, leaving a mess
  - Queuing for time
  - Loading cost and impremanance of data
  - Stability of Kafka / Other users interferring - ?

HPC PRODUCTION CLUSTER ENVIRONMENT

- Lessons learned:
  - Develop in your destination environment
  - Flash Storage makes life easy
    - Caveat -it is wiped when your reservation runs outs.
  - Lustr…
- No battle scars – credit to XSEDE wrangler management team and Kafak builders

REFERENCES

- Benchmarking Apache Kafka: 2 Million Writes Per Second (On Three Cheap Machines)Benchmark

https://engineering.linkedin.com/kafka/benchmarking-apache-kafka-2-million-writes-second-three-cheap-machines

ACKNOWLEDGEMENTS

- NASA Jet Propulsion Laboratory
  - Research & Technology Development: “Archiving, Processing and Dissemination for the Big Data Era”

- XSEDE
  - This work used the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation grant number ACI-1053575.
  - "XSEDE: Accelerating Scientific Discovery"
    - John Towns, Timothy Cockerill, Maytal Dahan, Ian Foster, Kelly Gaither, Andrew Grimshaw, Victor Hazlewood, Scott Lathrop, Dave Lifka, Gregory D. Peterson, Ralph Roskies, J. Ray Scott, Nancy Wilkins-Diehr, , Computing in Science & Engineering, vol.16, no. 5, pp. 62-74, Sept.-Oct. 2014, doi:10.1109/MCSE.2014.80