Understanding CPU Microarchitecture and Platform Characteristics to Maximize Big Data Performance

Presenter: Tony Wu

Understanding CPU Microarchitecture and Platform Characteristics to Maximize Big Data Performance

Tony Wu, Lingxiang Xiang

Intel Corporation

Oct 22, 2021

Legal Disclaimers

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. No computer system can be absolutely secure. Check with your system manufacturer or retailer or learn more at [intel.com].

For more complete information about performance and benchmark results, visit www.intel.com/benchmarks.

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors.

Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit www.intel.com/benchmarks.

Tests document performance of components on a particular test, in specific systems. Differences in hardware, software, or configuration will affect actual performance. Consult other sources of information to evaluate performance as you consider your purchase. For more complete information about performance and benchmark results, visit www.intel.com/benchmarks

Cost reduction scenarios described are intended as examples of how a given Intel- based product, in the specified circumstances and configurations, may affect future costs and provide cost savings. Circumstances will vary. Intel does not guarantee any costs or cost reduction.

Results have been estimated or simulated using internal Intel analysis or architecture simulation or modeling, and provided to you for informational purposes. Any differences in your system hardware, software or configuration may affect your actual performance.

Intel does not control or audit third-party benchmark data or the web sites referenced in this document. You should visit the referenced web site and confirm whether referenced data are accurate.

Intel processors of the same SKU may vary in frequency or power as a result of natural variability in the production process.

All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest Intel product specifications and roadmaps.

No computer system can be absolutely secure.

Intel, the Intel logo, Xeon, Intel vPro, Intel Xeon Phi, Look Inside., are trademarks of Intel Corporation in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others.

Microsoft, Windows, and the Windows logo are trademarks, or registered trademarks of Microsoft Corporation in the United States and/or other countries.

Optimization Notice

Optimization Notice: Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel.

Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

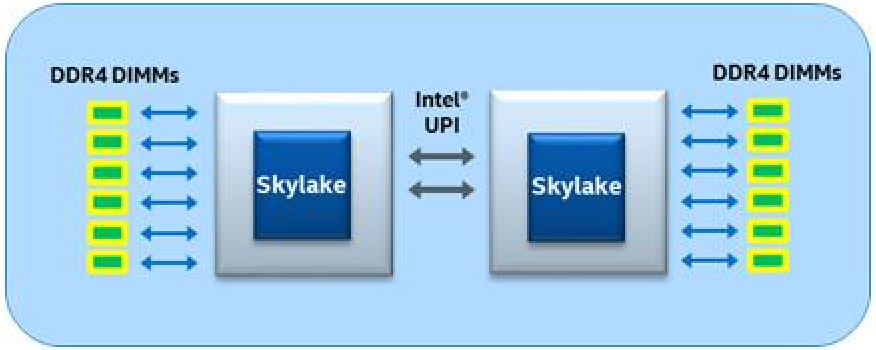
Notice revision #20110804

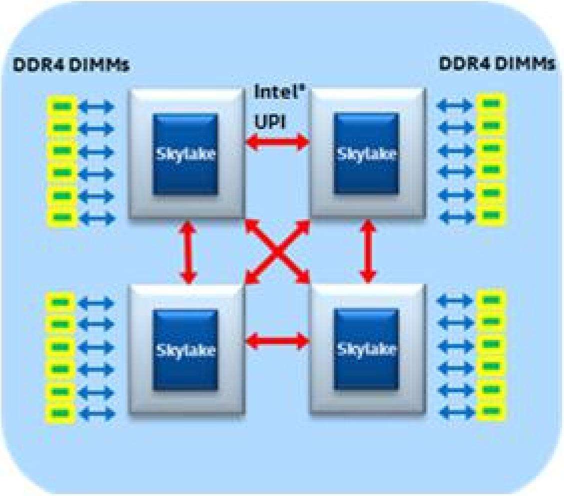
Agenda

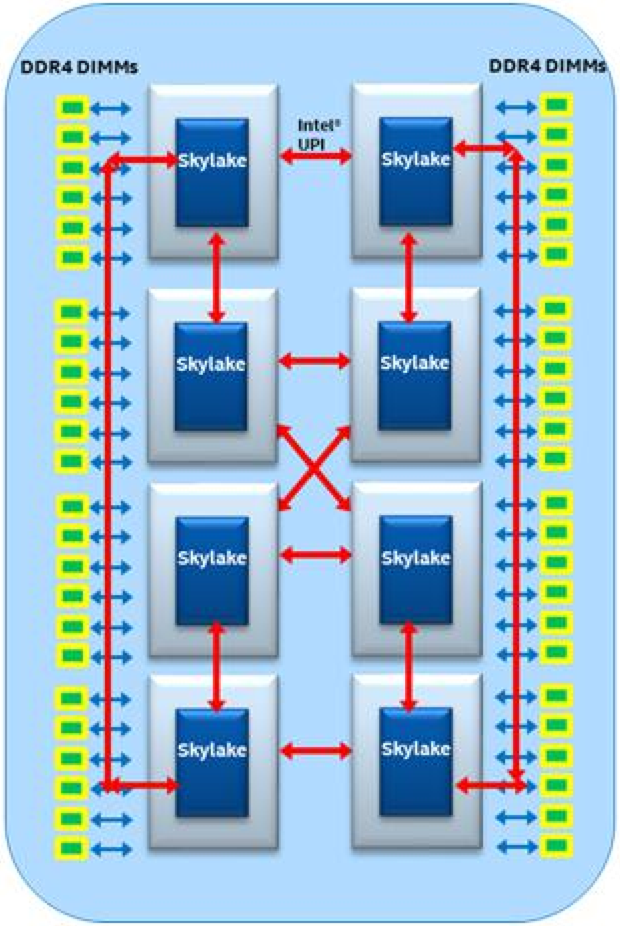
- Background
  - Intel Xeon Sever for Data Center
  - Apache Spark for Big Data
  - OLAP-DS Workload: based on a schema derived from TPC-DS
- Case Study
  - Case 1: Data Locality
  - Case 2: Cache Contention
  - Case 3: Resource Balancing
- Summary

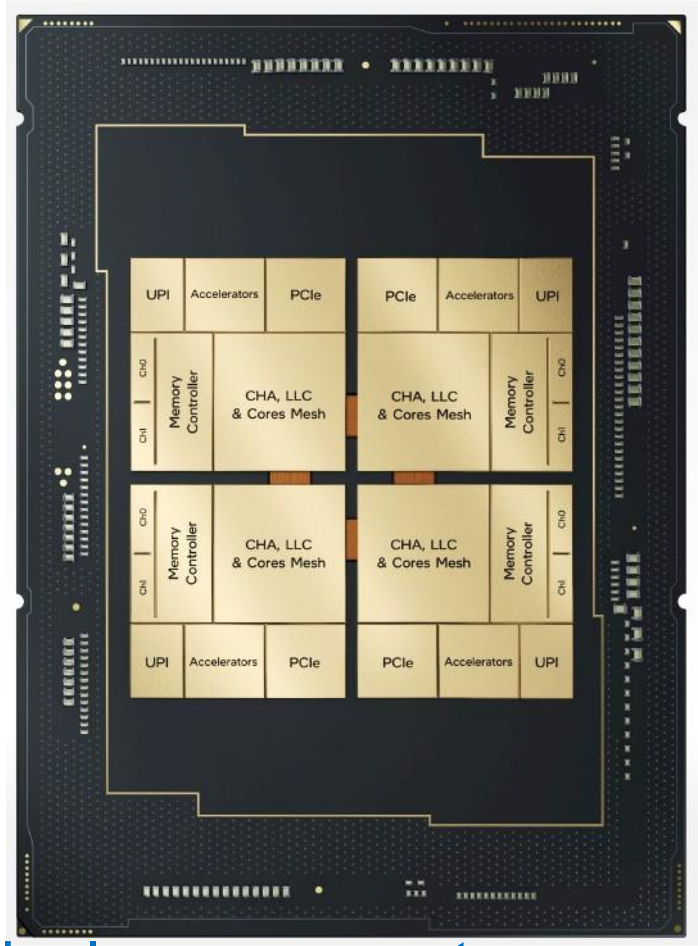
Intel Xeon Server

Intel Xeon Scalable Architecture







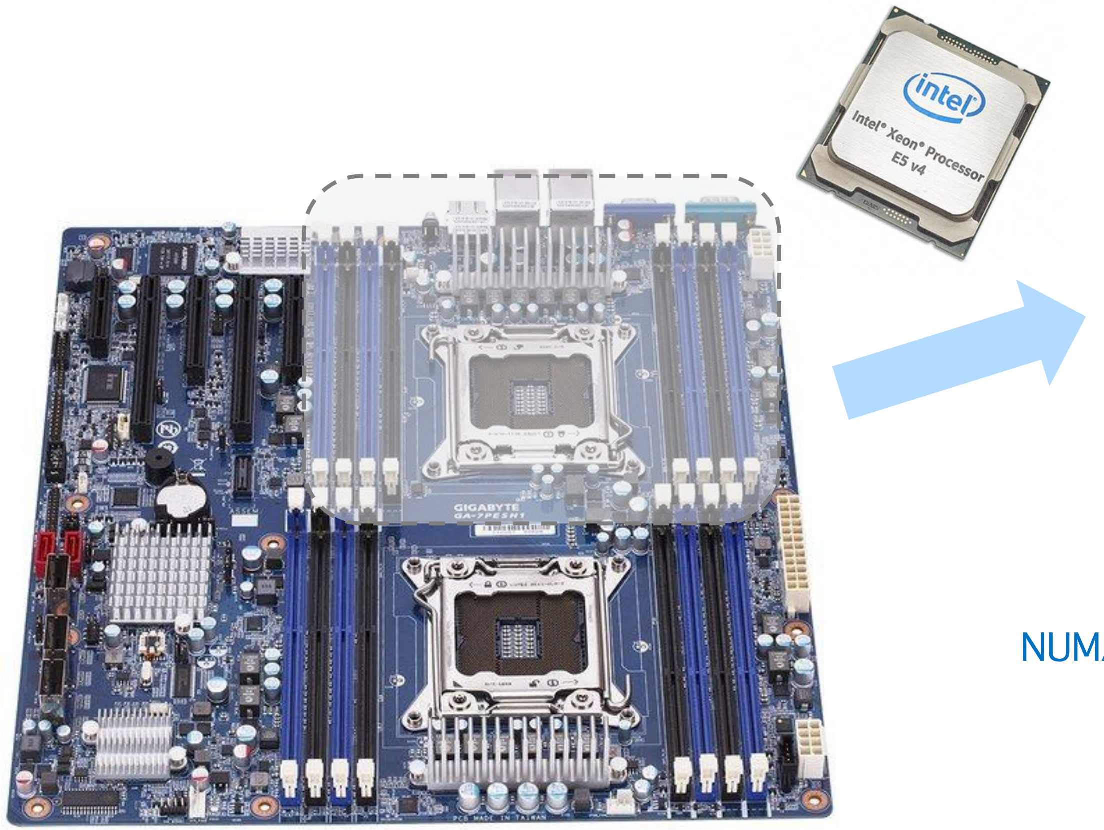


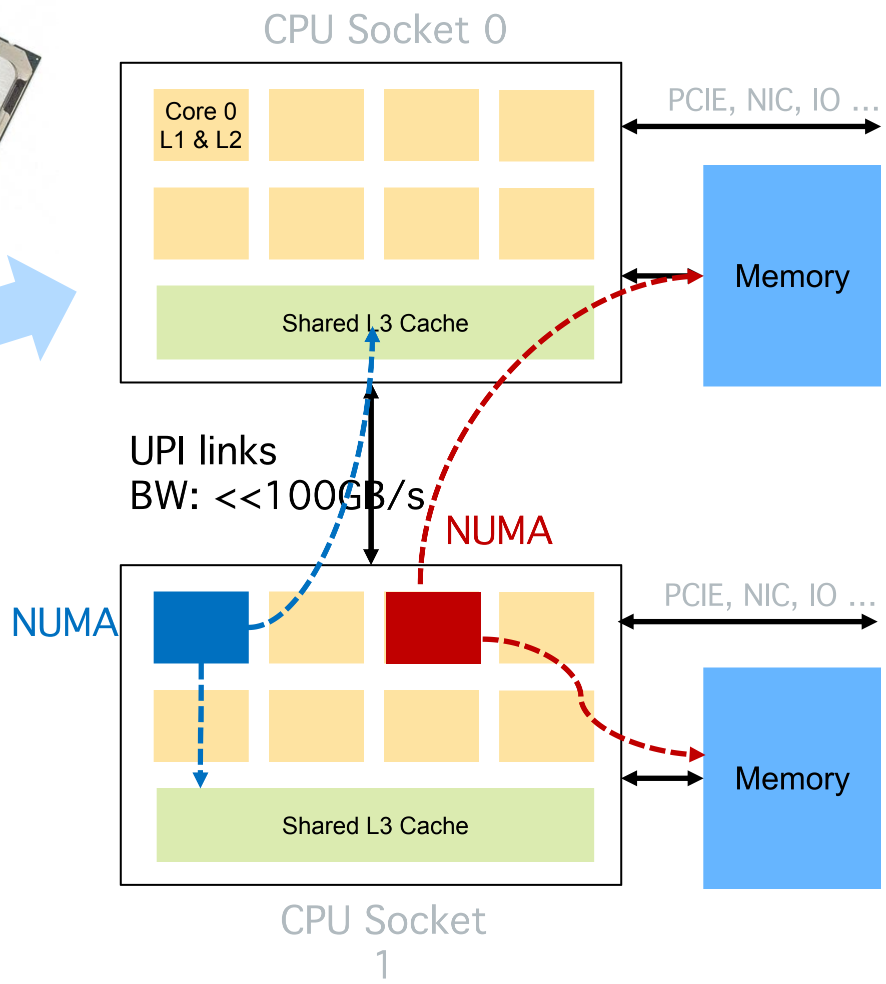
NUMA (Non-Uniform Memory Access): to scale from single processor to multiple processors

https://software.intel.com/content/www/us/en/develop/download/intel-64-and-ia-32-architectures-optimization-reference-manual.html

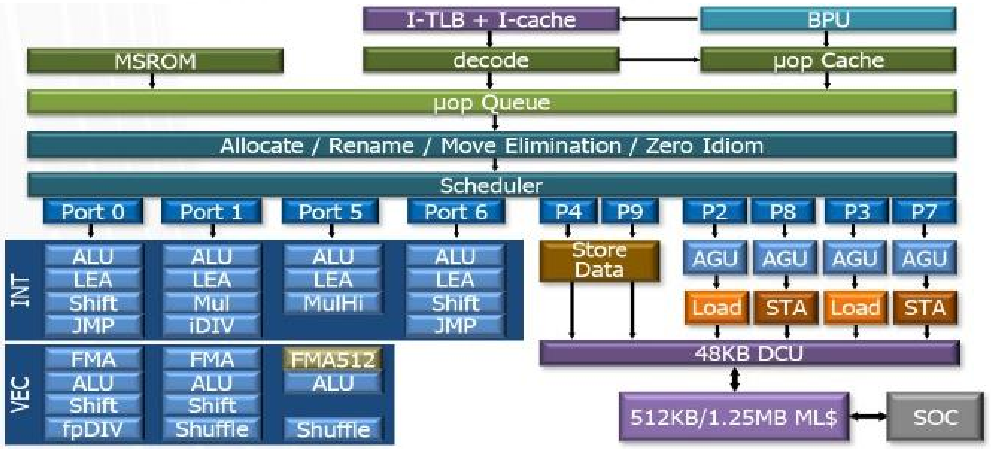
https://download.intel.com/newsroom/2021/client-computing/intel-architecture-day-2021-presentation.pdf

Intel Xeon System: NUMA Architecture





Intel Xeon Processor



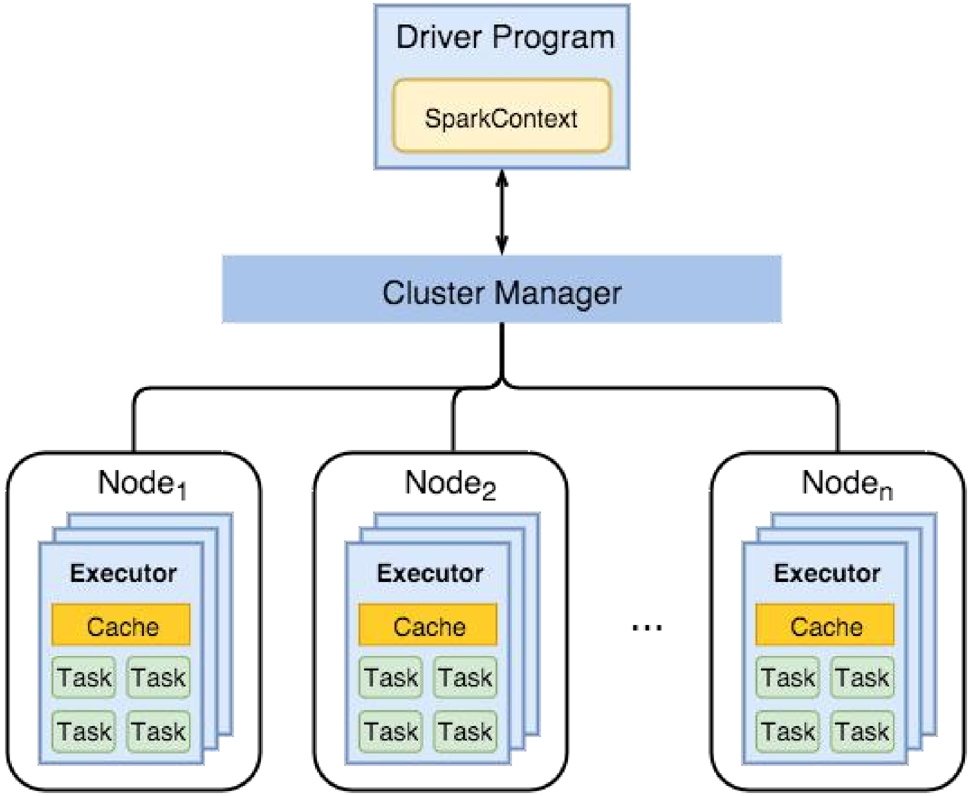
L3 shared among all cores. Some other resources (e.g. L1, L2, etc.) private to individual cores, which are shared/partitioned between two hardware threads.

https://software.intel.com/content/www/us/en/develop/download/intel-64-and-ia-32-architectures-optimization-reference-manual.html

Apache Spark for Big Data

Apache Spark Overview

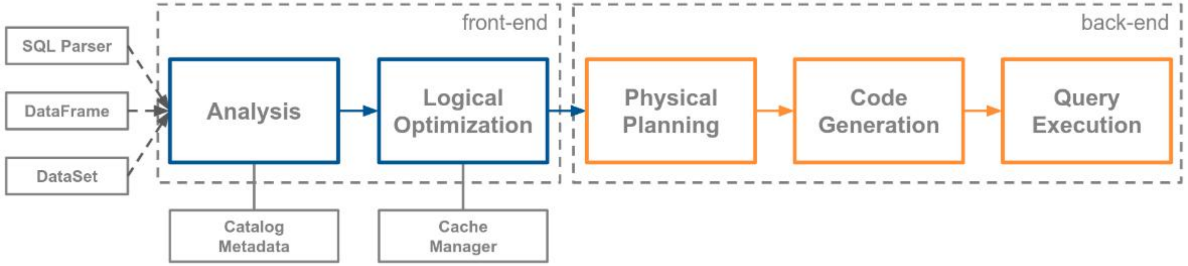
- Executor: The process responsible for executing a task.
- Manager: The machine on which the Driver program runs
- Worker: The machine on which the Executor program runs
- One task is executed on one partition of data on one executor (machine)
  - Job: A piece of code which reads some input from HDFS or local, performs some computation on the data and writes some output data.
  - Stages: Jobs are divided into stages, Map and Reduce
  - Each stage has some tasks, one task per partition.



- Each executor is a Java process, running with # of vcore, memory size, etc.
- Apache Spark only supports homogenous setting for executor
  - same configuration for executors

Inside an Executor

Lifecycle of a query



Example

```
SELECT count(*) FROM questions
   WHERE year == 2019
   GROUP BY user_id
INNER JOIN users ON questions.user_id = users.user_id
```

Physical Plan

```
*Project
+– * BroadcastHashJoin Inner BuildRight
    |– * HashAggregate
    |  +– Exchange
    |     +– * HashAggregate
    |        +– * Project
    |           +– * Filter
    |              +– * ColumnarToRow
    |                 +– Scan parquet
    +– BoradcastExchange
       +– * Project
          +– * Filter
             +– * ColumnarToRow
                +– Scan parquet
```

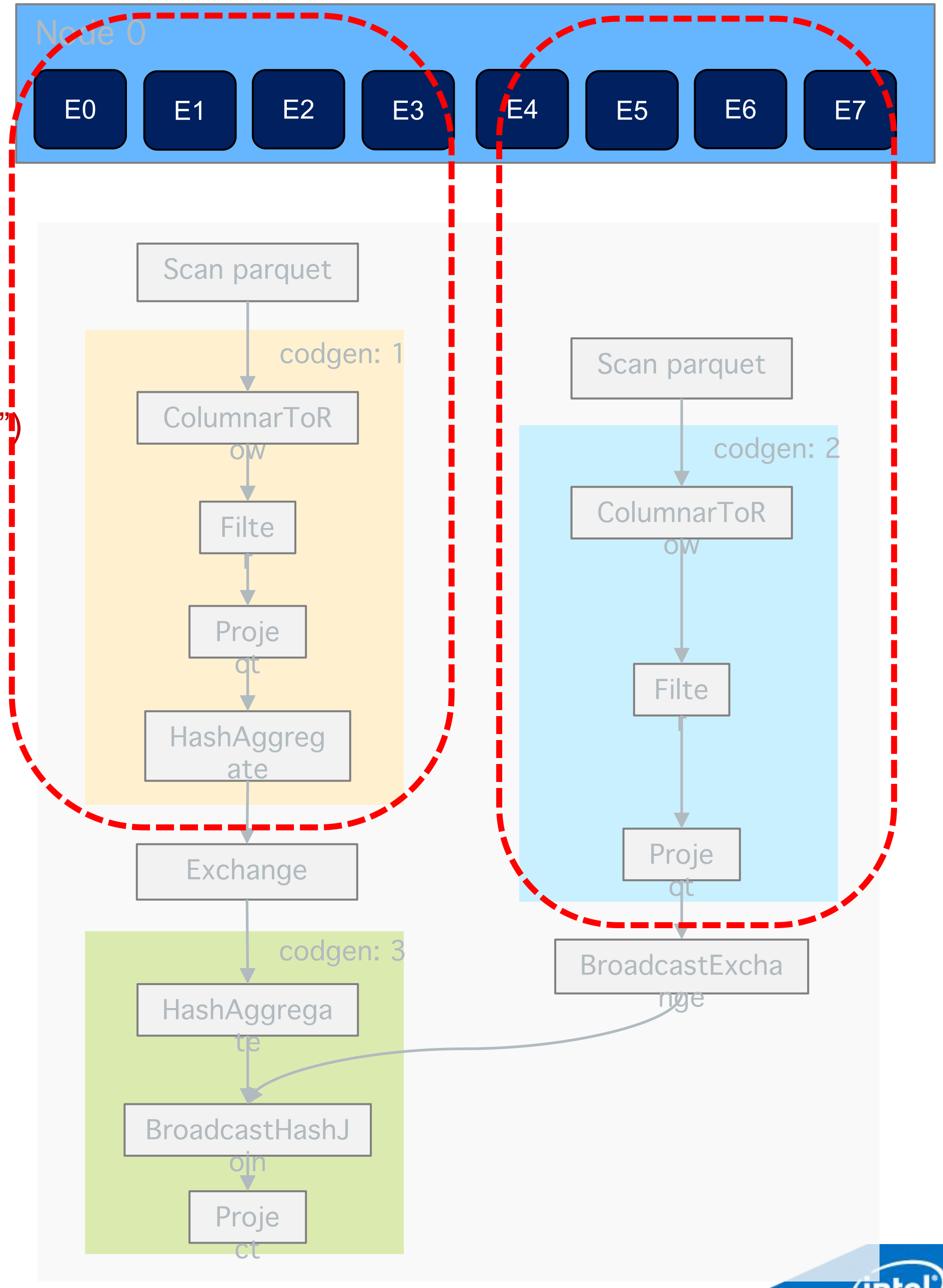
Execute A Physical Plan

```
*Project                                          codgen: 3
+– * BroadcastHashJoin Inner BuildRight
   |– * HashAggregate
   |  +– Exchange
   |     +– * HashAggregate                       codgen: 1
   |        +– * Project
   |           +– * Filter
   |              +– * ColumnarToRow
   |                 +– Scan parquet
   +– BroadcastExchange
      +– * Project                                codgen: 2
         +– * Filter
            +– * ColumnarToRow
               +– Scan parquet
```

exchange data among executors

→ partial aggregation ("GROUP BY")
→ select columns
→ filter w/ condition
→ columnar -> row execution
→ read input file ("user")

A codgen stage: an execution flow can be assigned to an executor




```
SELECT count(*) FROM questions
  WHERE year == 2019
  GROUP BY user_id
INNER JOIN users ON questions.user_id = users.user_id
```

OLAP-DS: based on a schema derived from TPC-DS

OLAP-DS Benchmark overview

- TPC-DS is a decision support benchmark that models several generally applicable aspects of a decision support system
- This study uses an OLAP workload (OLAP-DS) based on a schema derived from TPC-DS
  - Including 99 queries
  - examining large volumes of data, answering real-world business questions via ad-hoc reporting, online analytical processing and data-mining, and database maintenance functions
  - Performance Test is defined as Power Test and Throughput Tests

OLAP-DS Throughput Test

- The Throughput Test measures the ability of the system to process the most queries in the least amount of time with multiple users.
  - A query stream is defined as the sequential execution of a permutation of queries submitted by a single emulated user.
  - A session is defined as a uniquely identified process context capable of supporting the execution of user-initiated database activity.
  - A Throughput Test consists of S_q query sessions each running a single query stream.
  - The value of S_q is any even number larger than or equal to 4.

- Case Study
  - Case 1: Data Locality
  - Case 2: Cache Contention
  - Case 3: Resource Balancing

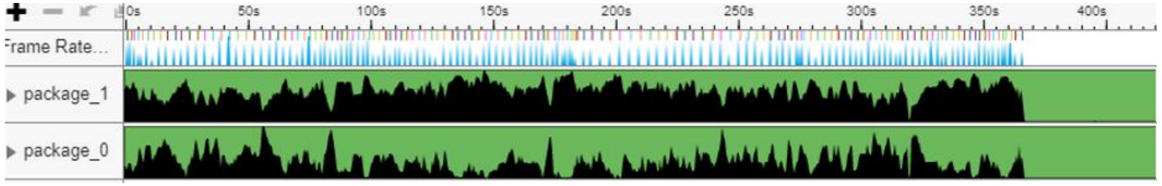
Case 1: Data Locality

- Excessive remote access may result in intensive inter-connect traffic and therefore poor performance

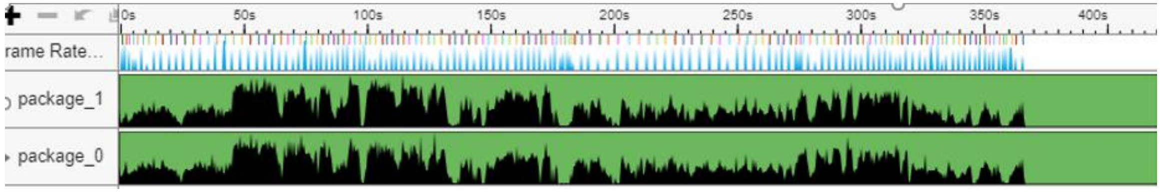
An example: Q88 on 2S Xeon, 28core/socket, totally 112 threads

<--num-executors 7 --executor-cores 16>

Poor Data locality / execution time 370sec



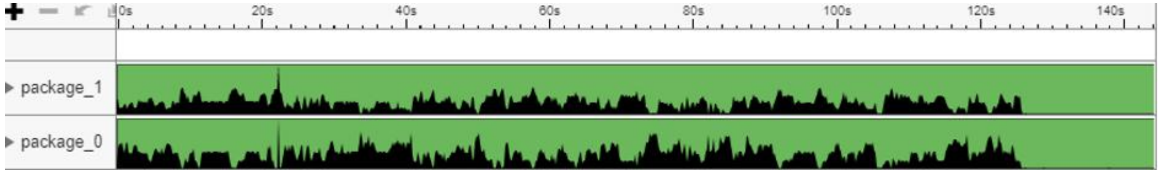
CPU cycle



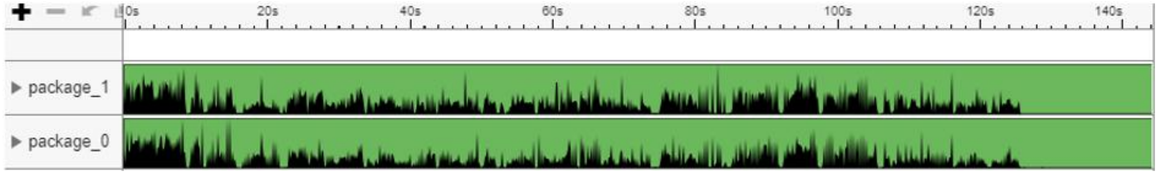
requests coming from a remote socket

<--num-executors 28 --executor-cores 4>

Improved Data locality / execution time 125sec



CPU cycle

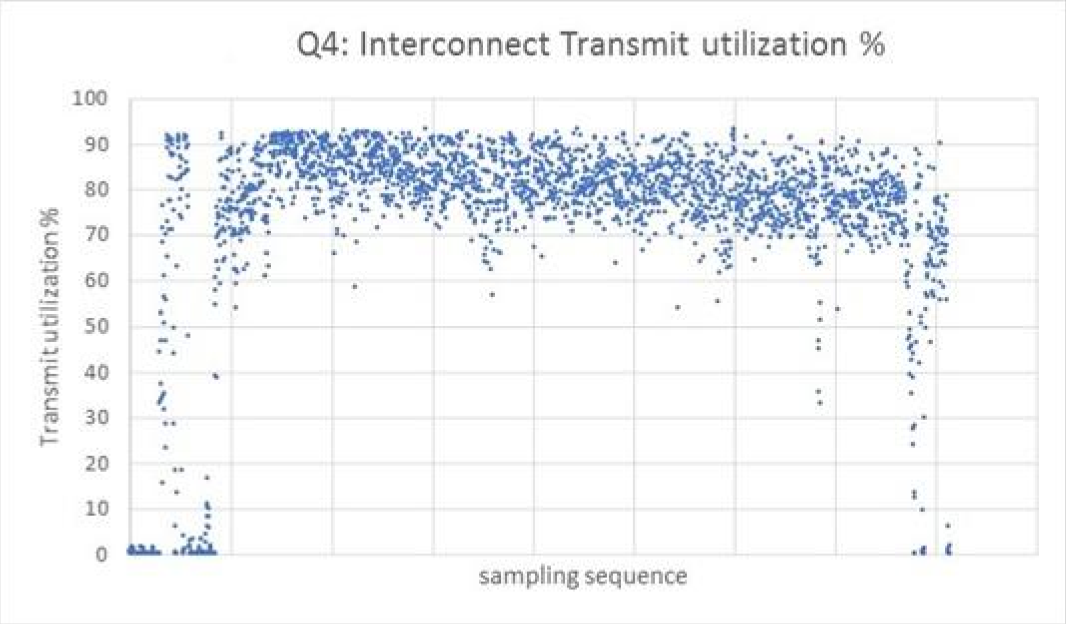


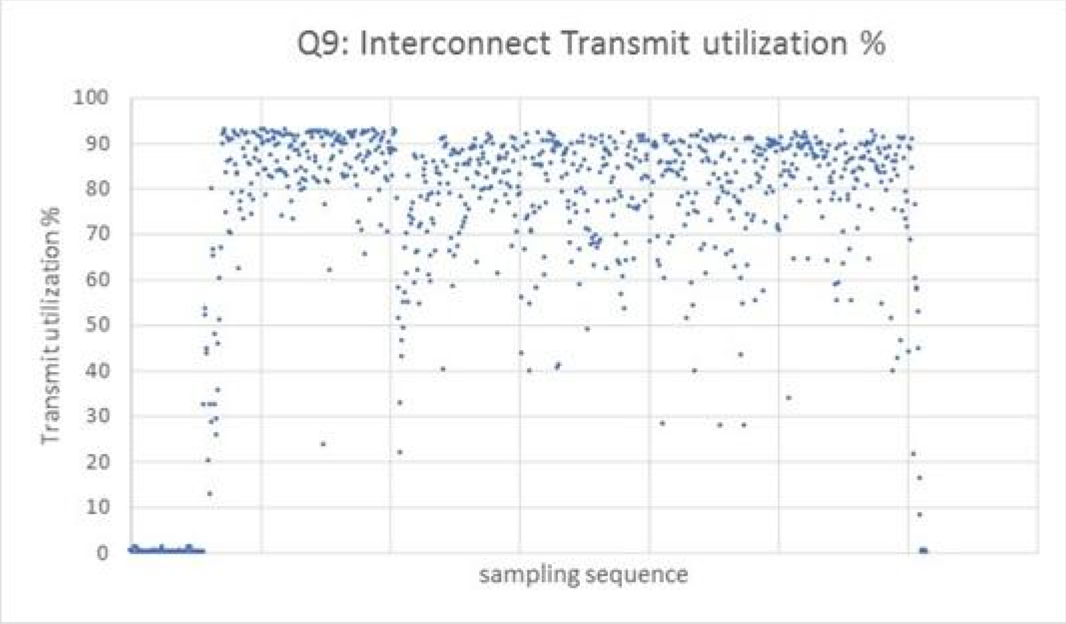
requests coming from a remote socket

Avoid artificial inter-socket communication that one executor always has cores on different sockets

Interconnect Traffic

- To develop or enable large applications with NUMA-aware is a challenge and demands significant investment on engineering resource
- Most enterprise and cloud applications demonstrate moderate to high (~50% or higher) remote memory access





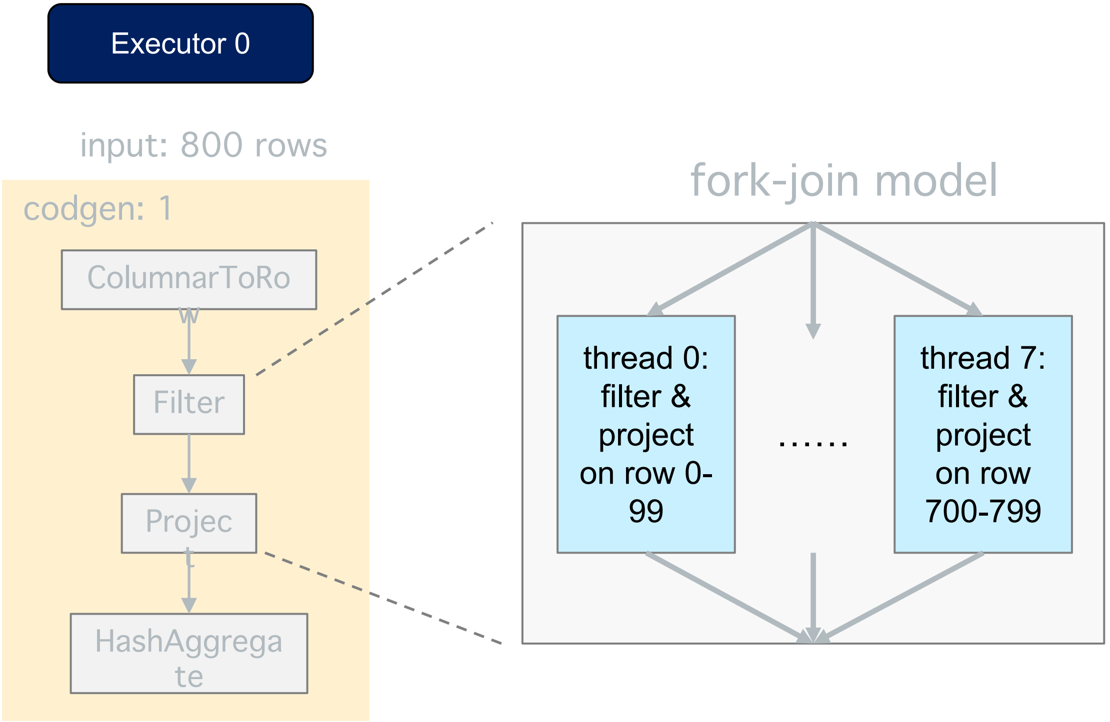
Even with good configuration, some software applications are inherently NUMA unfriendly

Why Spark Executor NUMA unfriendly

An executor is a multi-threaded process

Threading model: coarse-grain threading

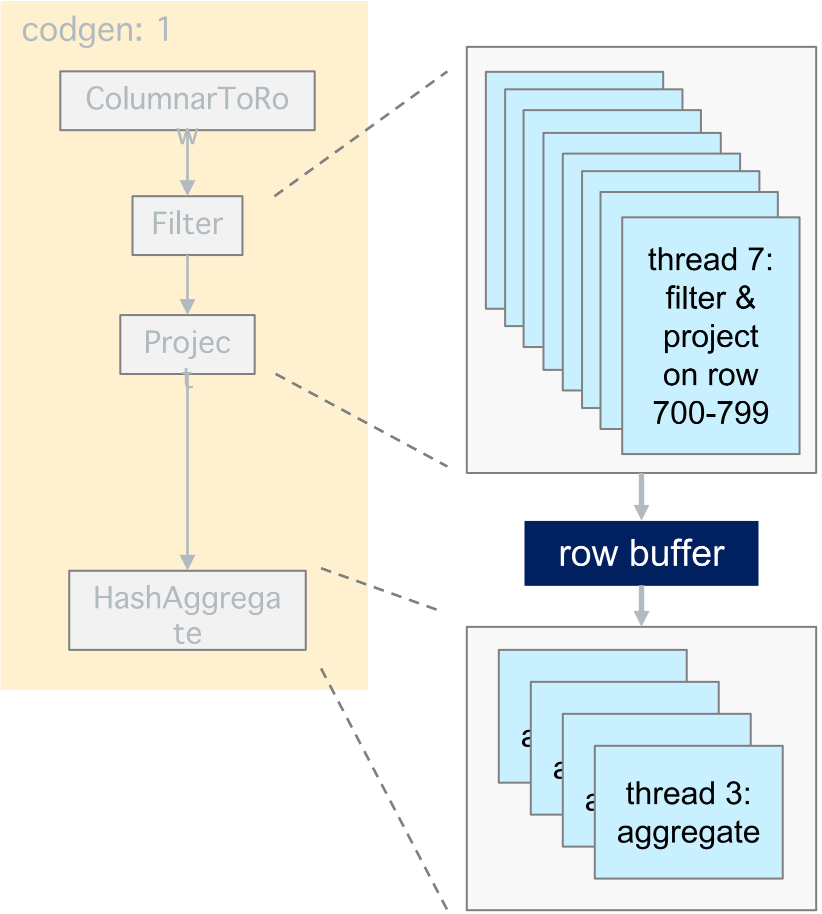
- spawn a new thread for a sub-task
  - Subquery, Filter, BroadcastExchange (local)
- # of active threads is dynamic at run time
  - difficult to pre-determine when and where a particular task thread will launch



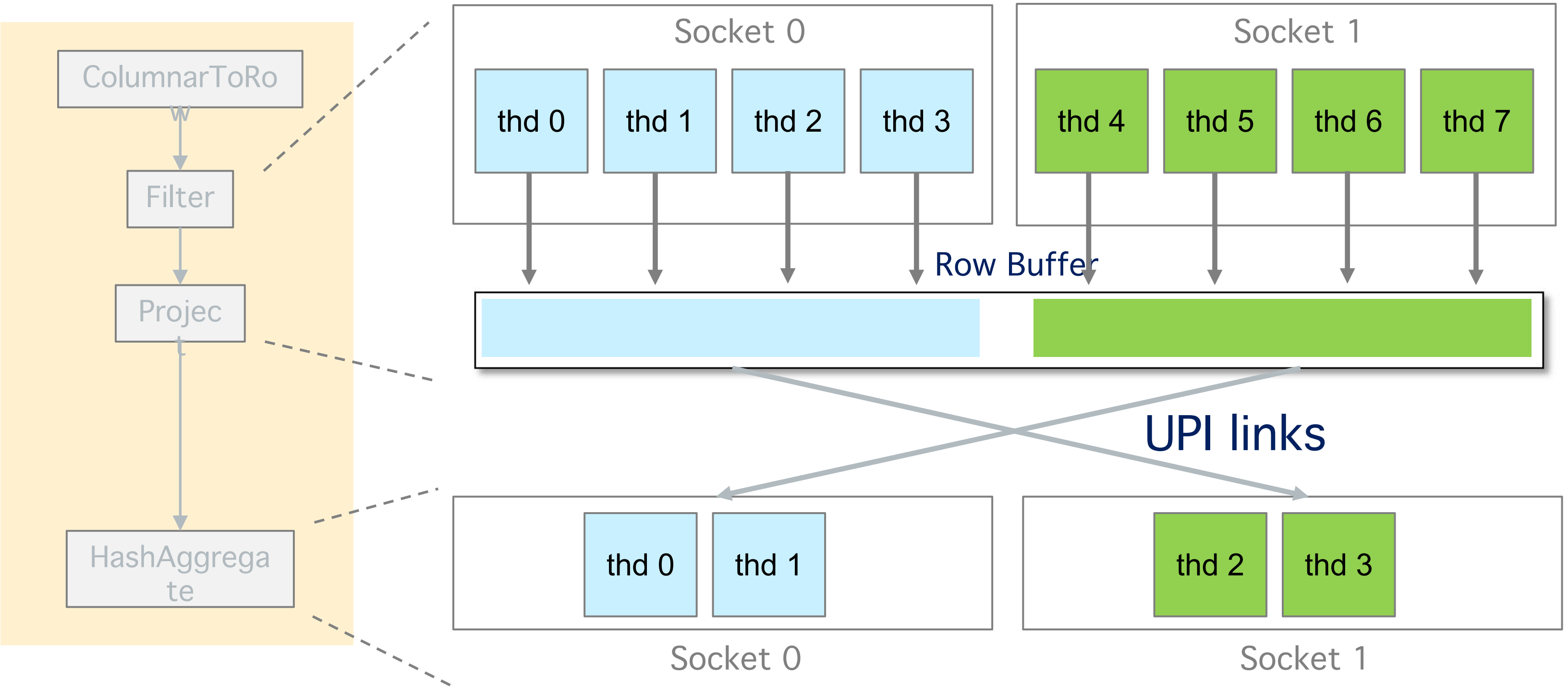
Why Executor NUMA unfriendly (2)

Inter-thread communication

- producer-consumer pattern
  - one sub-task's outputs → next sub-task's input
    - unmanaged row buffer (an array of key-value items)
- a thread may read a part of the buffer written by other threads in previous steps
  - esp. for aggregation ops (HashAggregate)
- and if threads are placed in different sockets, interconnect (UPI) traffic occurs



Why Executor NUMA unfriendly (3)



Rule of thumb (sensitivity to thread placement)

- a fatter DAG tree → more active threads
- more local exchange ops (HashAggregate, BroadcastExchange…) → more communication

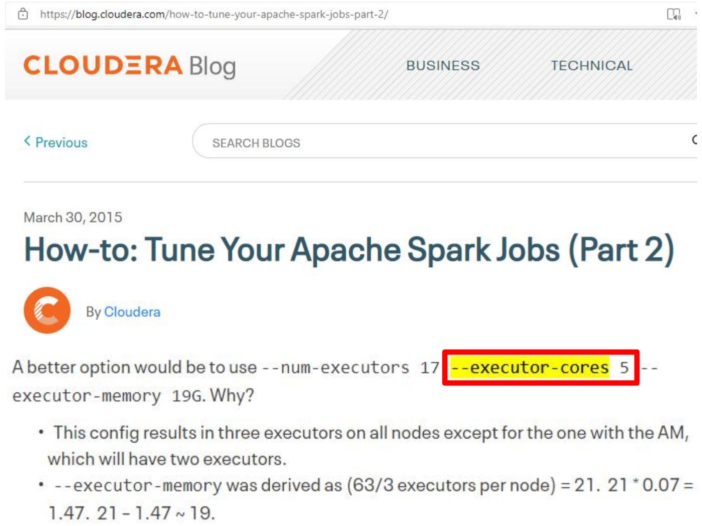
What can we do with Poor Data Locality?

- Software tuning and optimization
  - NUMA-binding
  - Configure executors (# of core) properly
  - NUMA-aware memory allocation
- Hardware enhancement and solution
  - Increase interconnect bandwidth
  - We have developed some novel technologies to improve interconnect efficiency

Case 2: Cache Contention

- The CPU architectural characteristic (e.g. shared L1/L2 cache) has been ignored or overlooked by the community
- The sub-optimal configuration leads to 7% performance regression in a state-of-art Xeon server

An example of cache contention:
https://blog.cloudera.com/how-to-tune-your-apache-spark-jobs-part-2/


https://blog.cloudera.com/how-to-tune-your-apache-spark-jobs-part-2/

CLOUDERA Blog BUSINESS TECHNICAL

< Previous SEARCH BLOGS

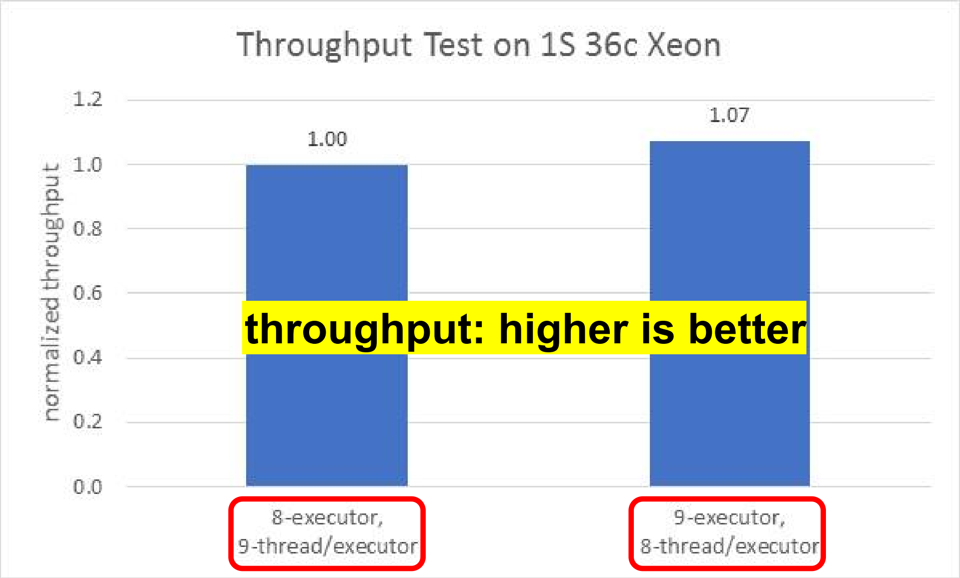
March 30, 2015

How-to: Tune Your Apache Spark Jobs (Part 2)

By Cloudera

A better option would be to use `--num-executors 17 --executor-cores 5 --executor-memory 19G`. Why?

- This config results in three executors on all nodes except for the one with the AM, which will have two executors.
- `--executor-memory` was derived as (63/3 executors per node) = 21. 21 * 0.07 = 1.47. 21 - 1.47 ~ 19.

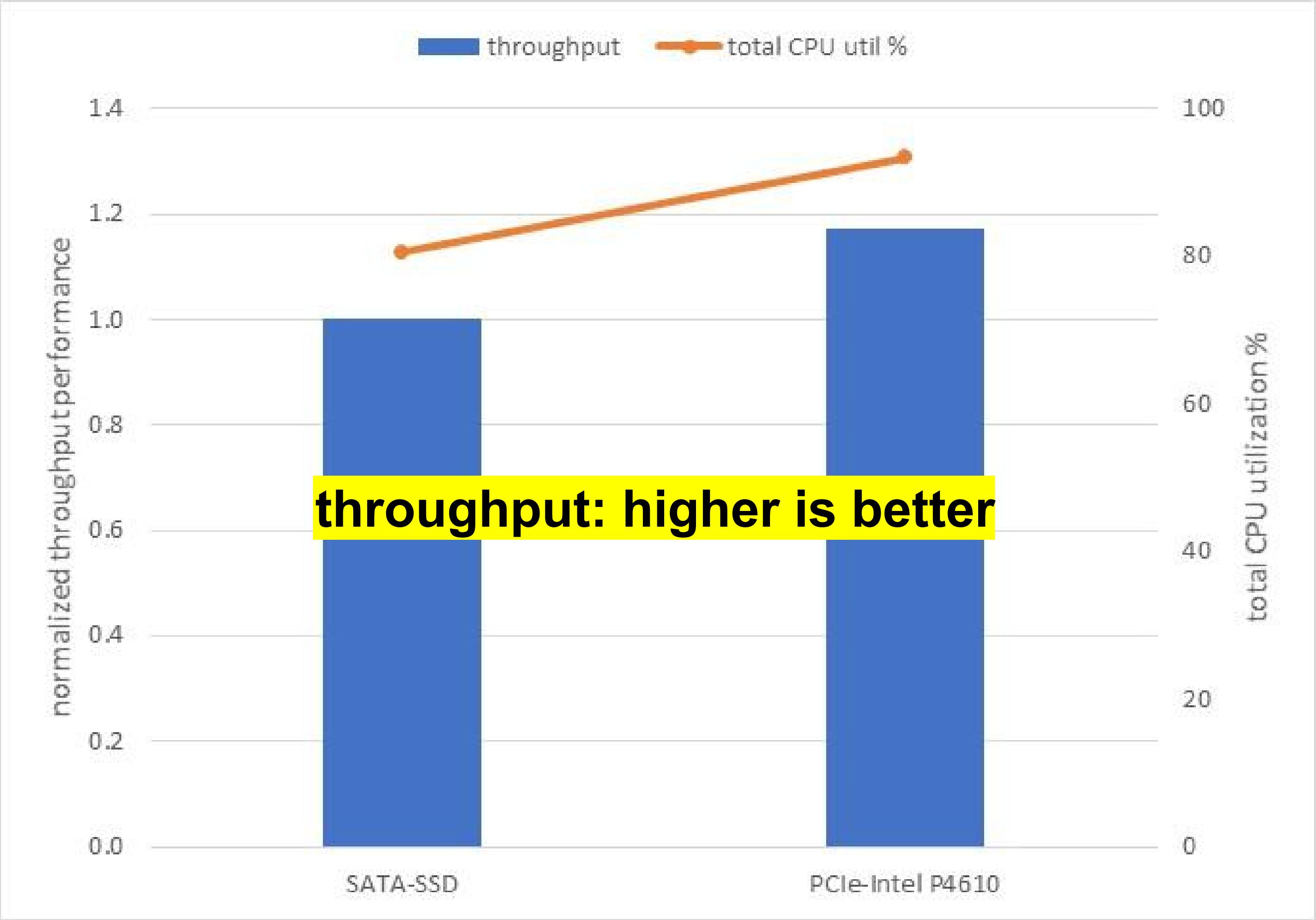


Avoid artificial L1/L2 contention: configure even # of vcore when HT is on

Case 3: System Resource Balancing

- Big Data system performance demands a balance of CPU, memory, storage, and networking resources
  - Compute: # of core, CPU frequency
  - Memory: bandwidth, capacity
  - Storage: Read/Write bandwidth
  - Network: bandwidth
- Fallacies and Pitfalls
  - Average usage
  - Theoretical peak capacity/bandwidth

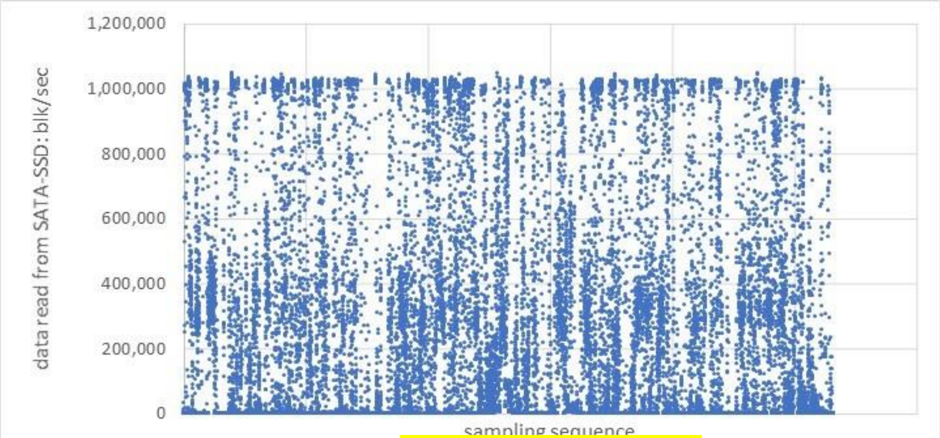
Fallacies and Pitfalls (1): Average Usage

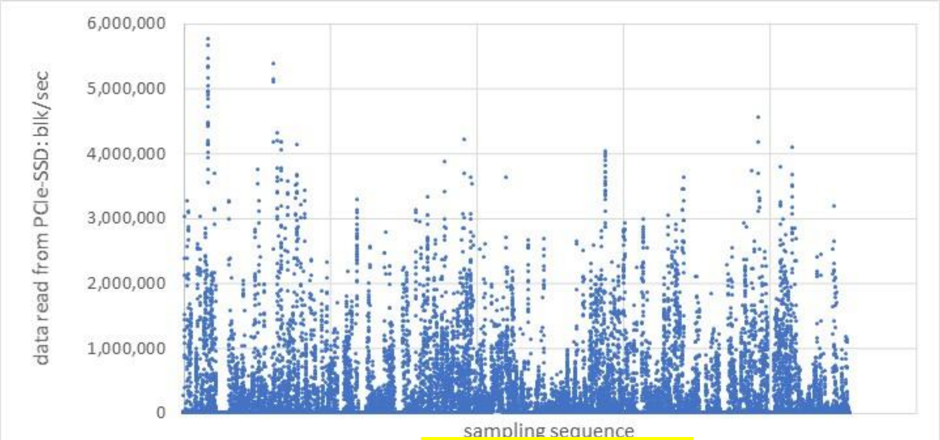


	SATA-SSD	P4618
bread/s	236,074	365,481
bwrtn/s	112,833	124,161

- SATA-SSD: SanDisk SDSSDA-1T00, Read/write speeds of up to 535MB/s / 450MB/s
- PCIe-SSD: Intel P4618 6.4TB, Read/write speeds of up to 6650MB/s / 5350MB/s
- Average usage is well below the SATA-SSD's spec.
- Why a faster SSD (i.e. PCIe-SSD) leads to 17% performance improvement?

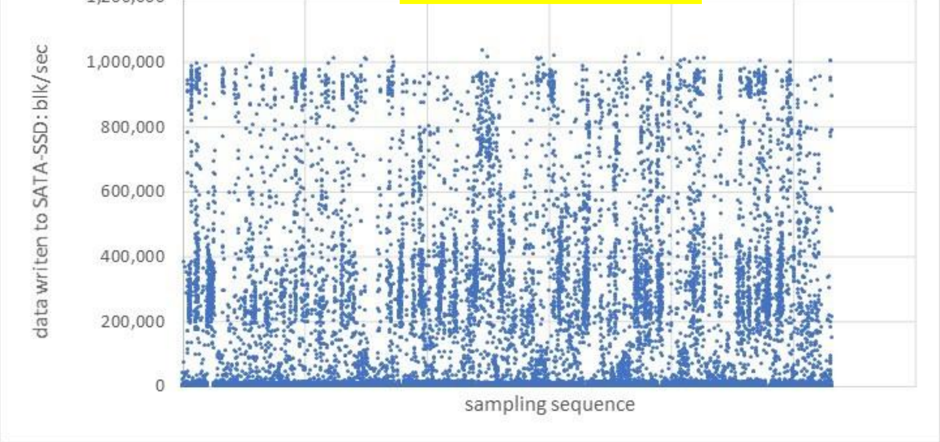
I/O Bandwidth Bounded with SATA-SSD

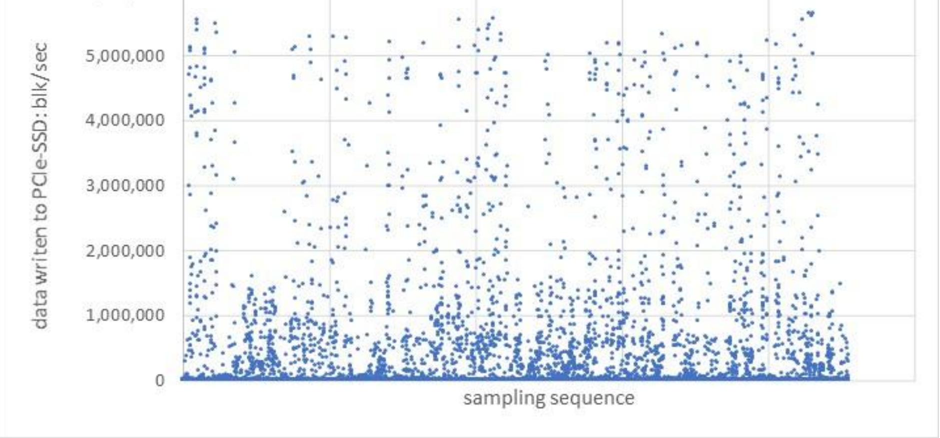




SATA-SSD

PCIe-SSD





- SATA-SSD: frequently reach Read / Write bandwidth limits

- PCIe-SSD: enough capability to meet Read / Write demand

Transient behavior is important for non-steady state workloads

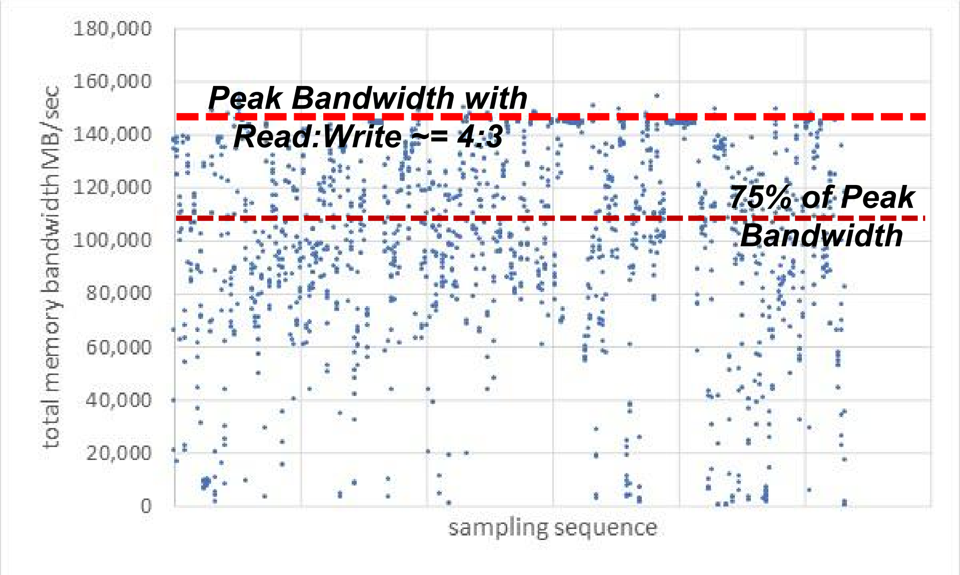
Fallacies and Pitfalls (2): Theoretical Peak

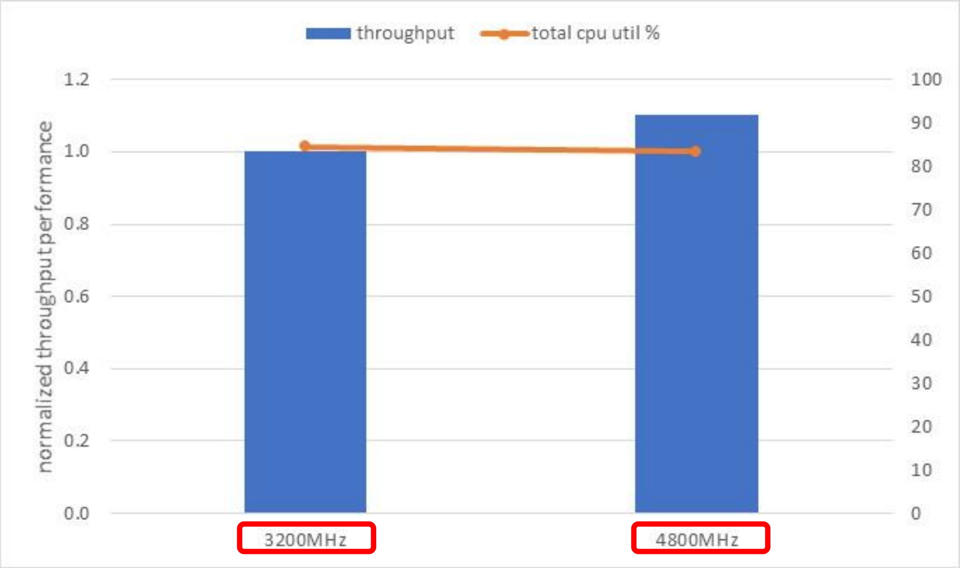
- Theoretical peak bandwidth is greater than actual bandwidth

  <memory-speed (3.2GT/s)* data-bus-width (8 bytes per channel) x # of channel (8) x # of socket (1)>
  = 205 GB/sec

- The actual peak bandwidth depends on data access pattern, and is usually much less than the theoretical peak

  <actual peak bandwidth with Read:Write ~= 4:3> ~= 145 MB/sec, measured via a microbenchmark





A good grasp of queuing theory (e.g. 75% rule) helps your performance analysis

Summary

- Develop your knowledge on CPU microarchitecture
    - NUMA is almost everywhere
- Understand your application and workload
    - Balance system resources
- Use your experience and intuition
    - Knowledge of Probability and Queuing theory speeds up your analysis

© 2021 Intel Corporation

Intel, the Intel logo, and Xeon are trademarks of Intel Corporation in the U.S. and/or other countries.
*Other names and brands may be claimed as the property of others.

Java is a registered trademark of Oracle and/or its affiliates.

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors.

Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit http://www.intel.com/performance.

Intel does not control or audit third-party benchmark data or the web sites referenced in this document. You should visit the referenced web site and confirm whether referenced data are accurate.

（排名不分先后）

"我们在 DIVE 全球基础软件创新大会上等你"

DIVE

全球基础软件创新大会

议 / 题 / 提 / 交

大 / 会 / 官 / 网

深入基础软件，打造新型数字底座

2021.11.26-27 / 北京·悠唐皇冠假日酒店

THANKS