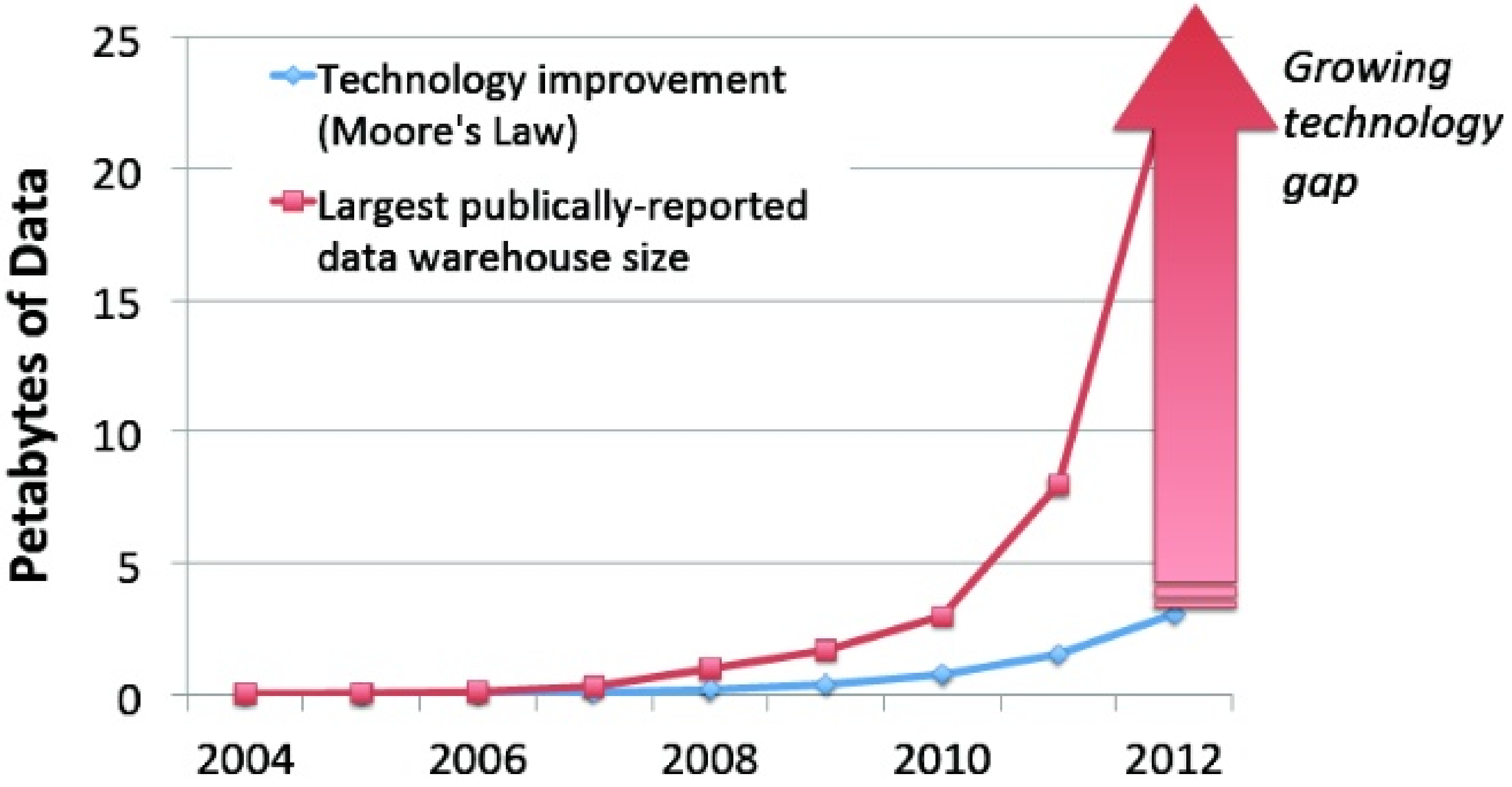
Performance Characterization of In-Memory Data Analytics on a Scale-up Server

Ahsan Javed Awan
KTH Royal Institute of Technology

SUMMIT
EUROPE 2016

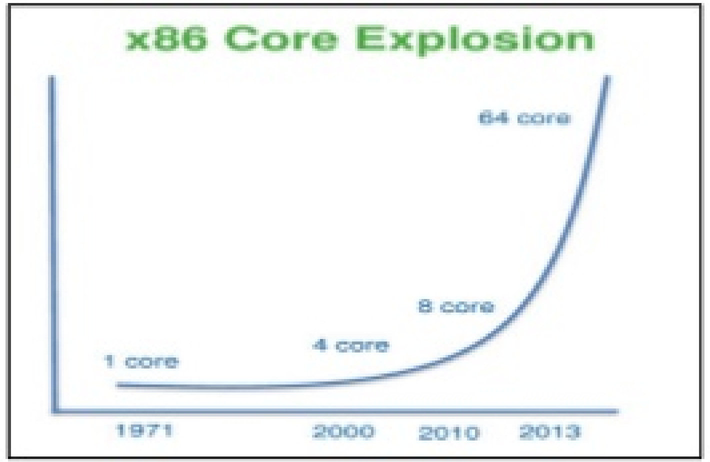
Why should we care about architecture support?



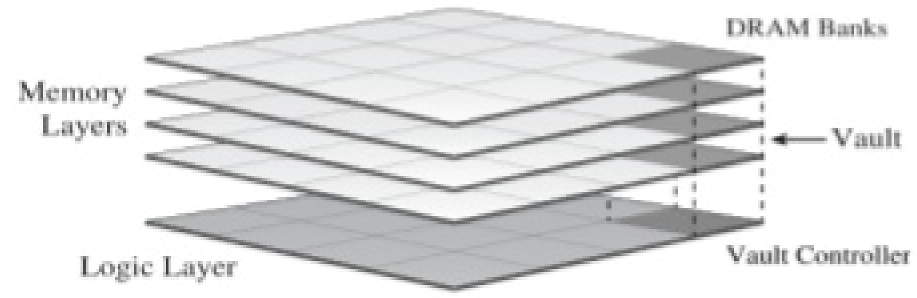
WinterCorp Survey, www.wintercorp.com

Cont..

- Exponential increase in core count.
- A mismatch between the characteristics of emerging big data workloads and the underlying hardware.
- Newer promising technologies (Hybrid Memory Cubes, NVRAM etc)



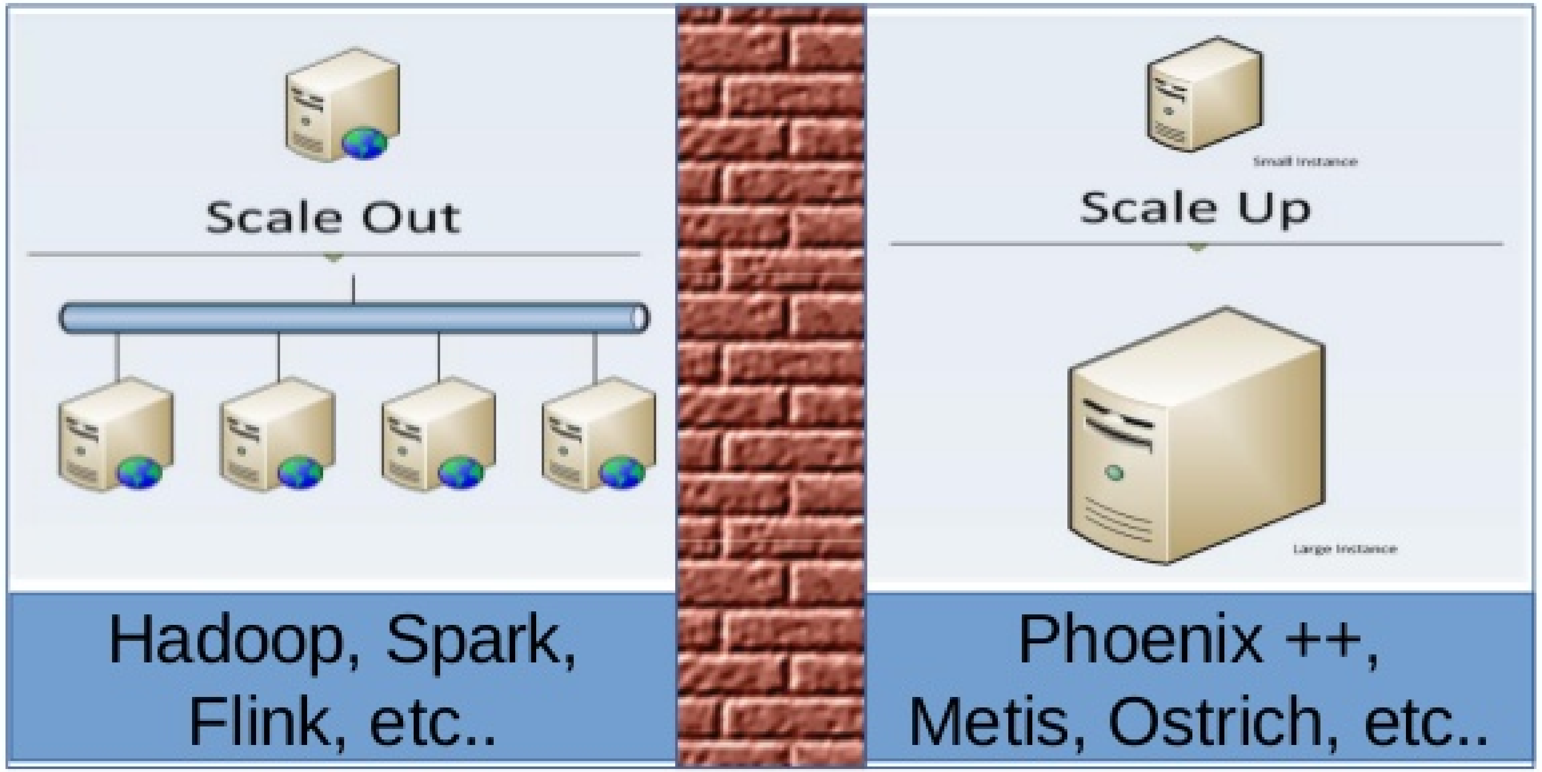
- Clearing the clouds, ASPLOS' 12
- Characterizing data analysis workloads, IISWC' 13
- Understanding the behavior of in-memory computing workloads, IISWC' 14



. 1. Structure of an HMC.

*Source: SGI

Cont...



Hadoop, Spark, Flink, etc..	Phoenix ++, Metis, Ostrich, etc..

Our Focus

Improve the node level performance through architecture support

*Source: http://navcode.info/2012/12/24/cloud-scaling-schemes/

Our Contribution

Poor Multi-core Scalability of data analytics with Spark

Problems Identified

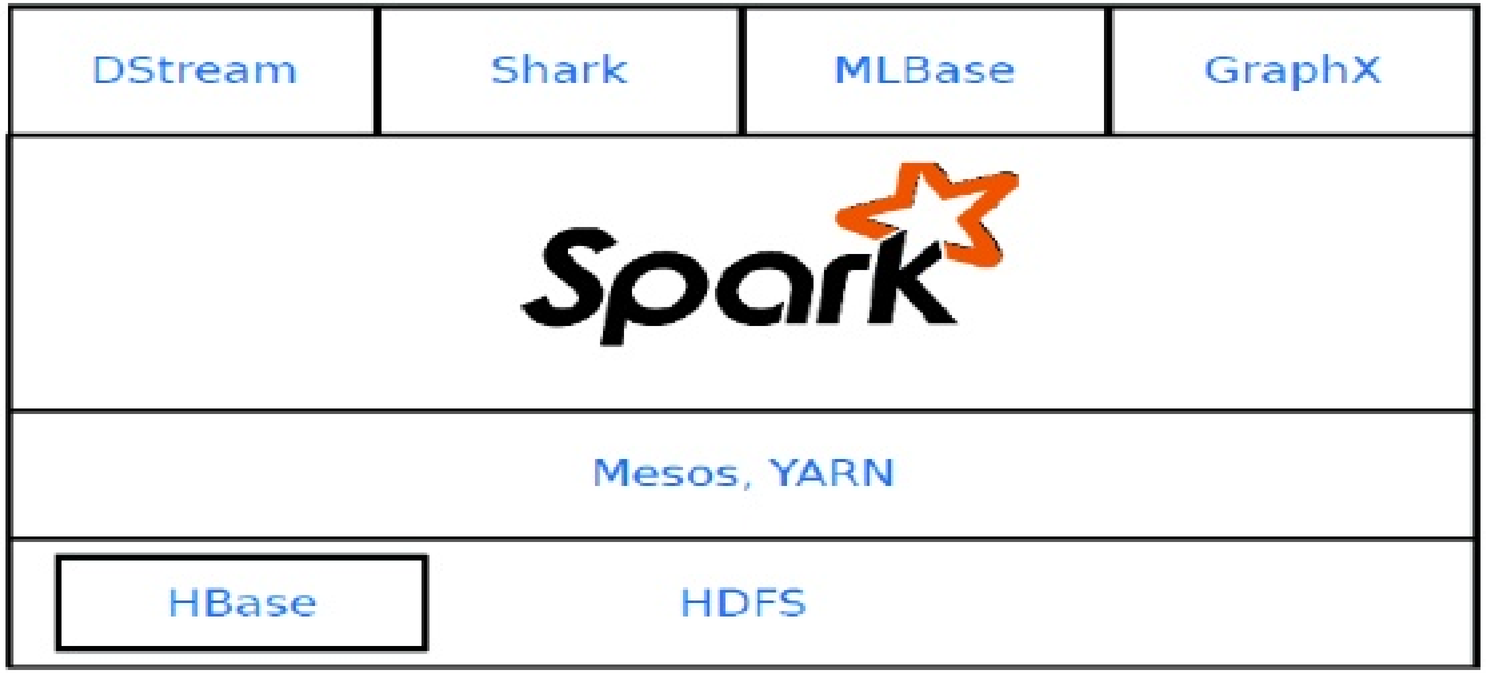
- Work Time Inflation
- Thread Level Load Imbalance
- Wait Time in I/O
- GC overhead

DRAM Bound

Solutions Proposed

- NUMA Awareness
- Hyper Threaded Cores
- No next-line prefetchers
- Lower DRAM speed
- Future node based on Hybrid ISP + 2D PIM
- PS over G1 GC
- Multiple Small executors

Which Scale-out Framework ?



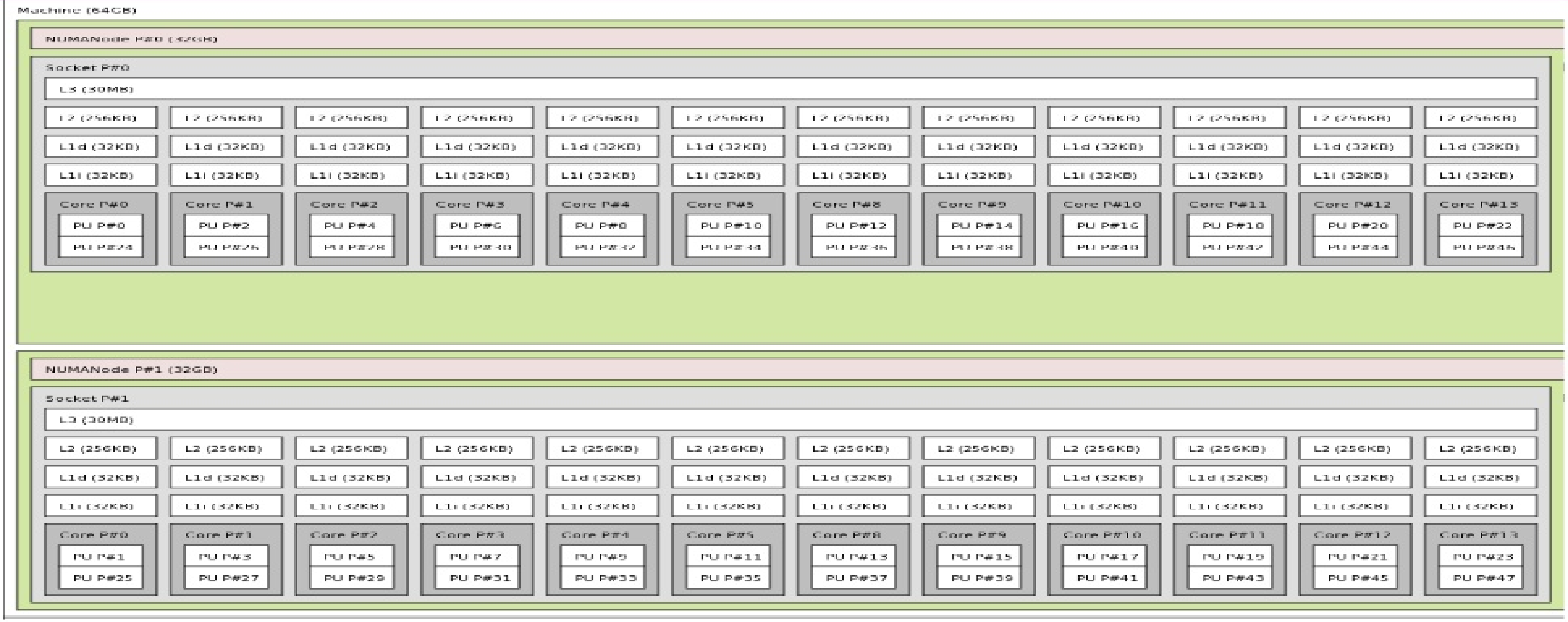
- Tuning of Spark internal Parameters
- Tuning of JVM Parameters (Heap size etc..)
- Micro-architecture Level Analysis using Hardware Performance Counters.

[Picture Courtesy: Amir H. Payberah]

Which Benchmarks ?

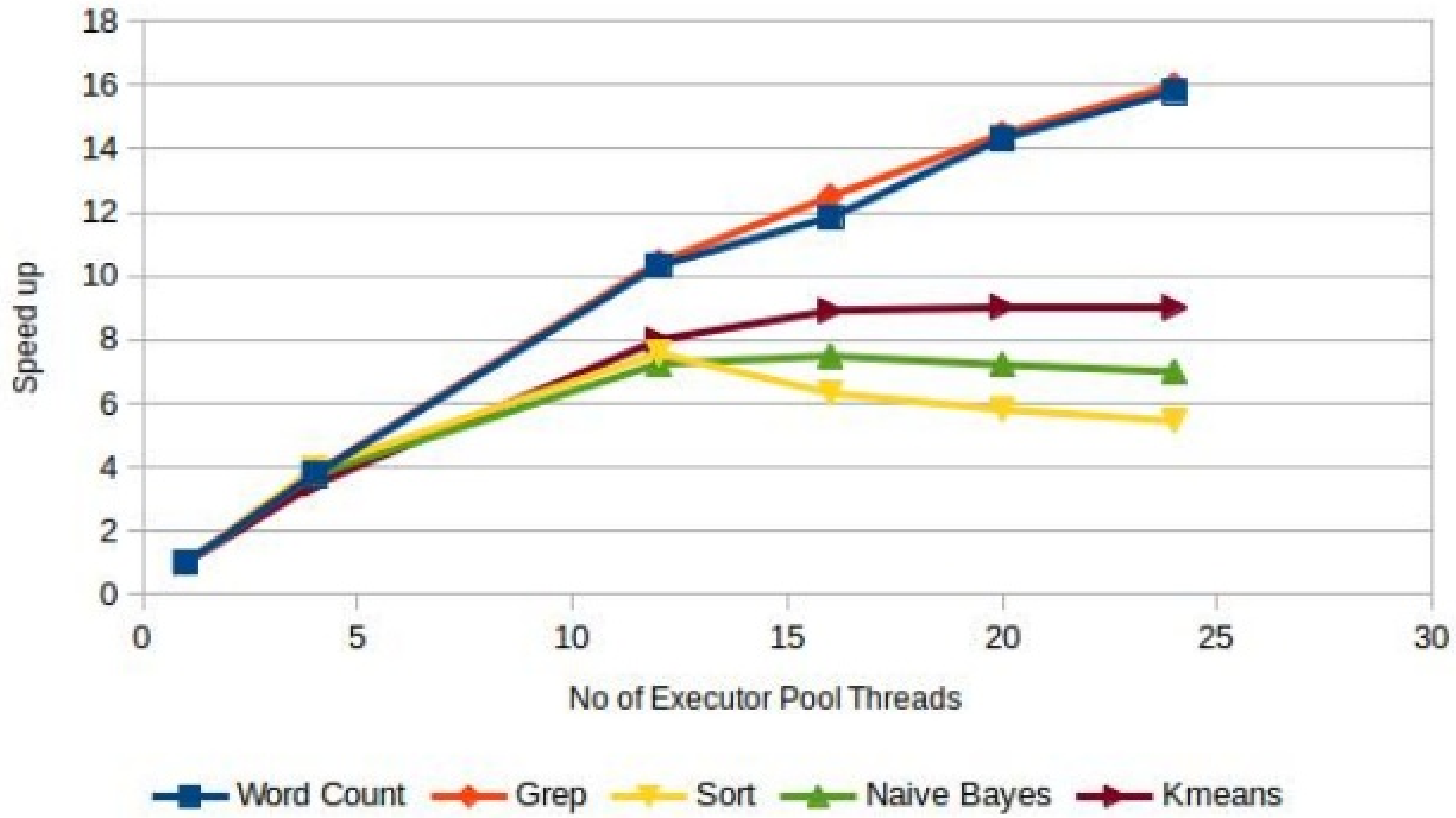
Spark Library	Workload	Description	Input data-sets
Spark Core	Word Count (Wc)	counts the number of occurrence of each word in a text file	Wikipedia Entries (Structured)
	Grep (Gp)	searches for the keyword The in a text file and filters out the lines with matching strings to the output file	
	Sort (So)	ranks records by their key	Numerical Records
	NaiveBayes (Nb)	runs sentiment classification	Amazon Movie Reviews
Spark Mllib	K-Means (Km)	uses K-Means clustering algorithm from Spark Mllib. The benchmark is run for 4 iterations with 8 desired clusters	Numerical Records (Structured)
	Gaussian (Gu)	uses Gaussian clustering algorithm from Spark Mllib. The benchmark is run for 10 iterations with 2 desired clusters	
	Sparse NaiveBayes (SNb)	uses NaiveBayes classification alogrithm from Spark Mllib	
	Support Vector Machines (Svm)	uses SVM classification alogrithm from Spark Mllib	
	Logistic Regression(Logr)	uses Logistic Regression alogrithm from Spark Mllib	
Graph X	Page Rank (Pr)	measures the importance of each vertex in a graph. The benchmark is run for 20 iterations	Live Journal Graph
	Connected Components (Cc)	labels each connected component of the graph with the ID of its lowest-numbered vertex	
	Triangles (Tr)	determines the number of triangles passing through each vertex	
Spark Streaming	Windowed Word Count (WWc)	generates every 10 seconds, word counts over the last 30 sec of,data received on a TCP socket every 2 sec.	Wikipedia Entries
	Streaming Kmeans (Skm)	uses streaming version of K-Means clustering algorithm from Spark Mllib. The benchmark is run for 4 iterations with 8 desired clusters	Numerical Records
	Streaming Logistic Regression (Slogr)	uses streaming version of Logistic Regression algorithm from Spark Mllib. The benchmark is run for 4 iterations with 8 desired clusters	
	Streaming Linear Regression (Slr)	uses streaming version of Logistic Regression algorithm from Spark Mllib. The benchmark is run for 4 iterations with 8 desired clusters	
Spark SQL	Aggregation (SqlAg)	implements aggregation query from BigdataBench using DataFrame API	Tables
	Join (SqlJo)	implements join query from BigdataBench using DataFrame API	

Which Machine ?



Intel's Ivy Bridge Server

Do Spark workloads have good multi-core scalability?

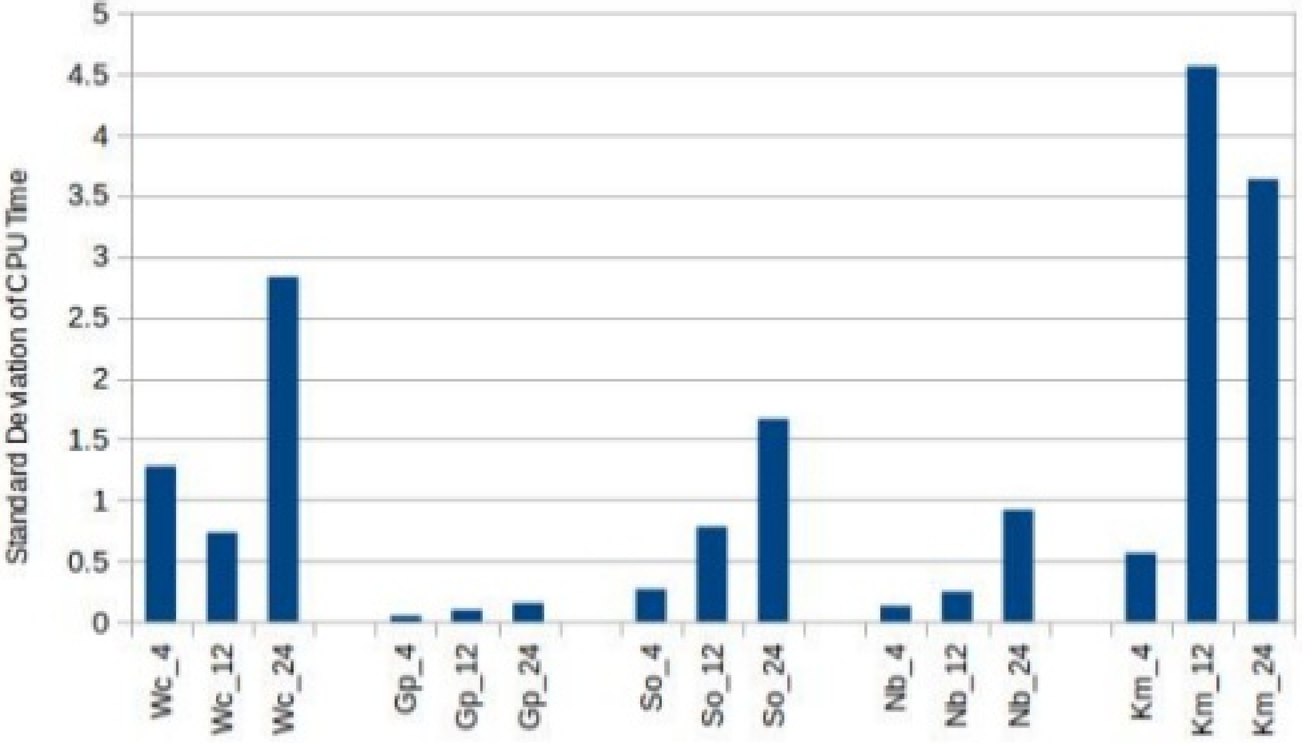


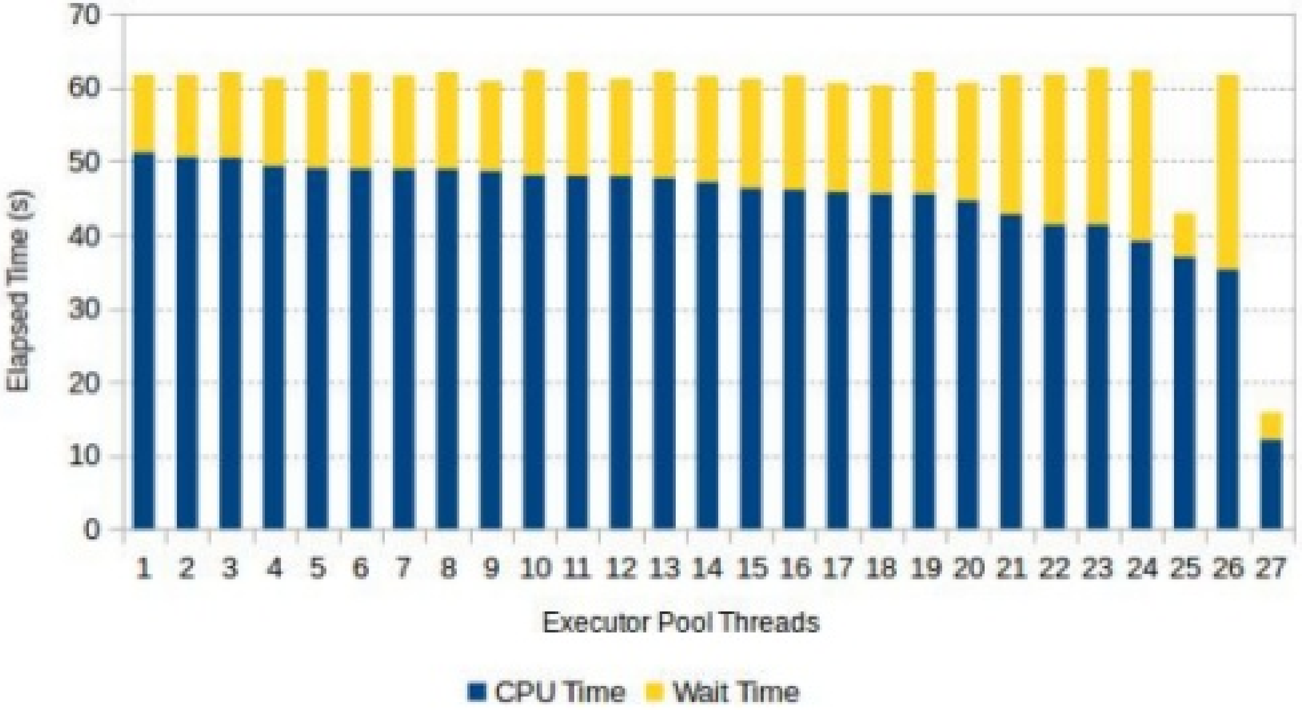
Spark scales poorly in Scale-up configuration

SPARK SUMMIT EUROPE 2016

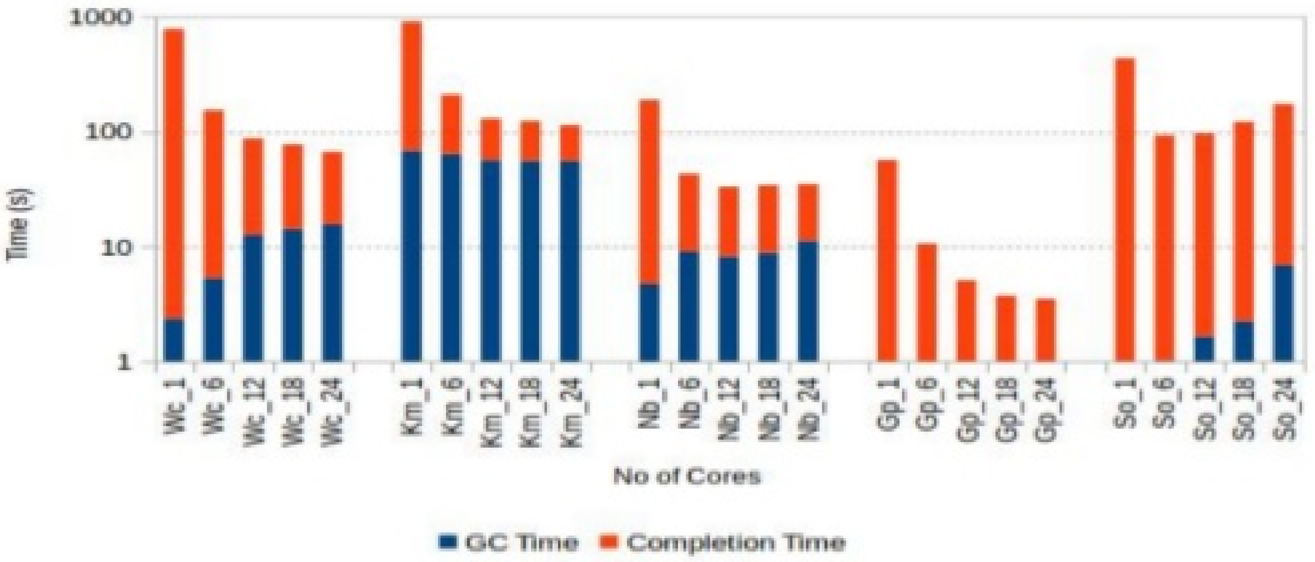
UPC BSC KTH VETENSKAP OCH KONST

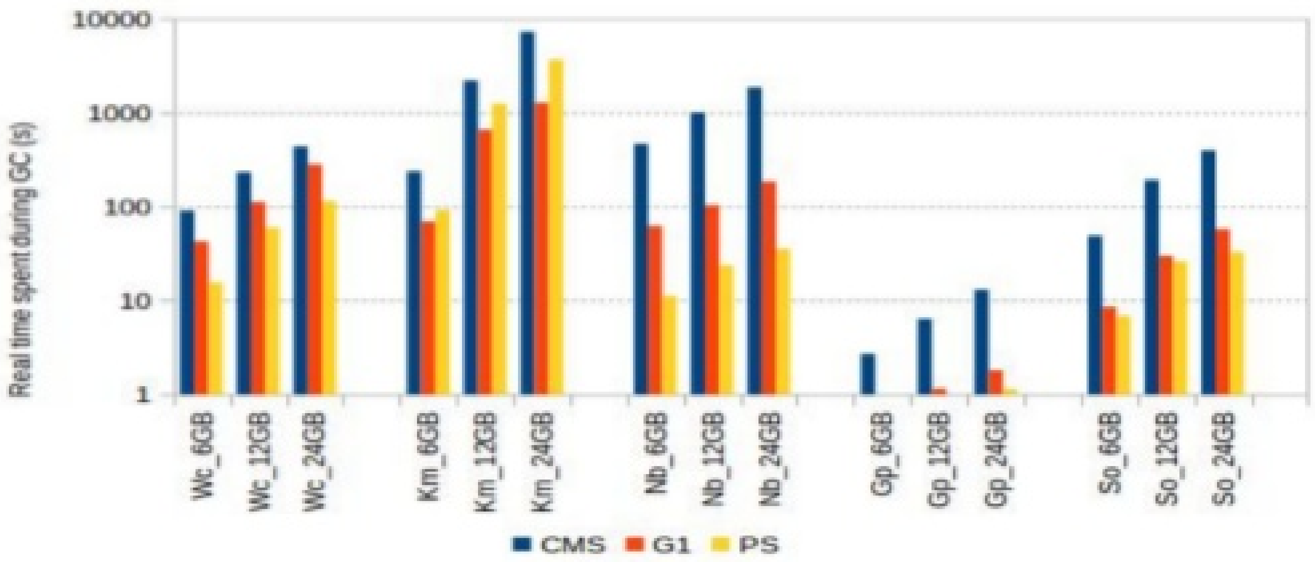
Is there a thread level load imbalance ?





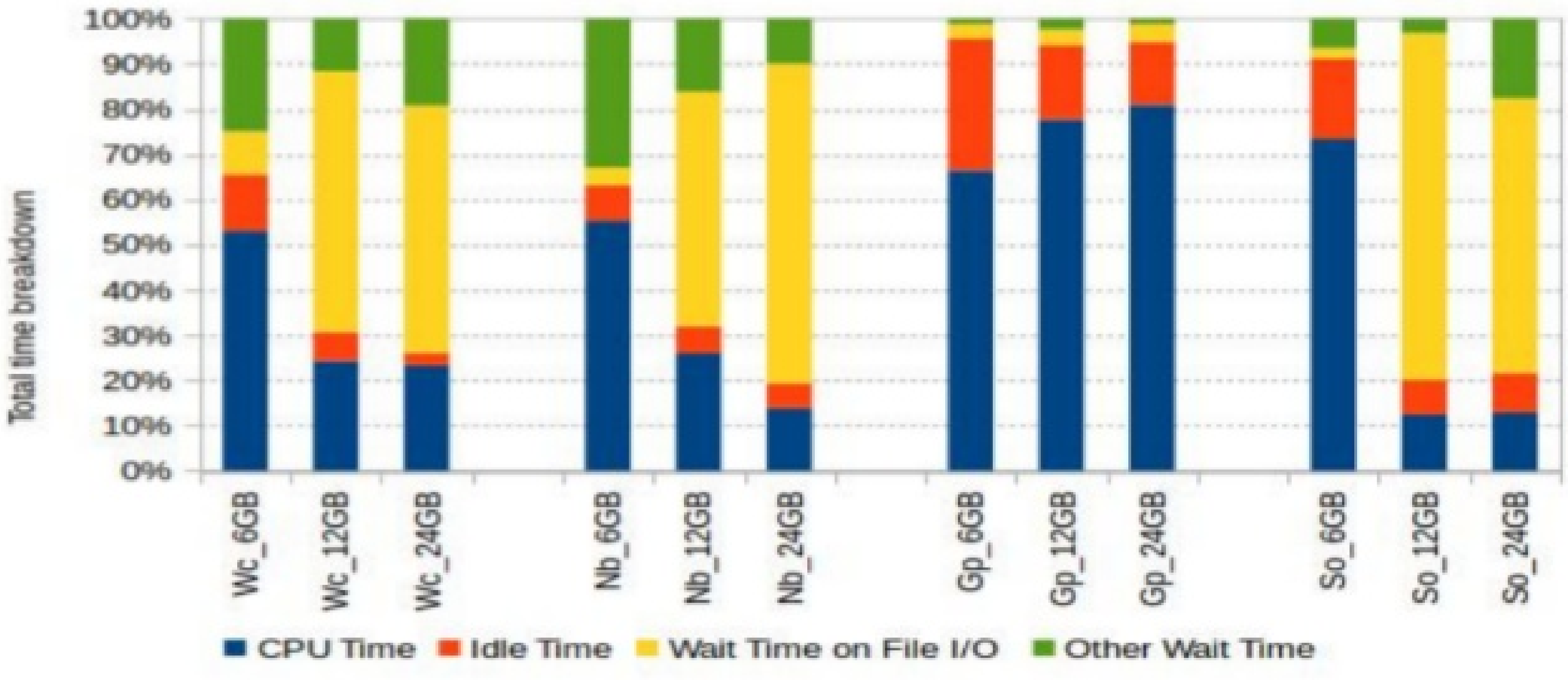
Is GC detrimental to scalability of Spark applications?





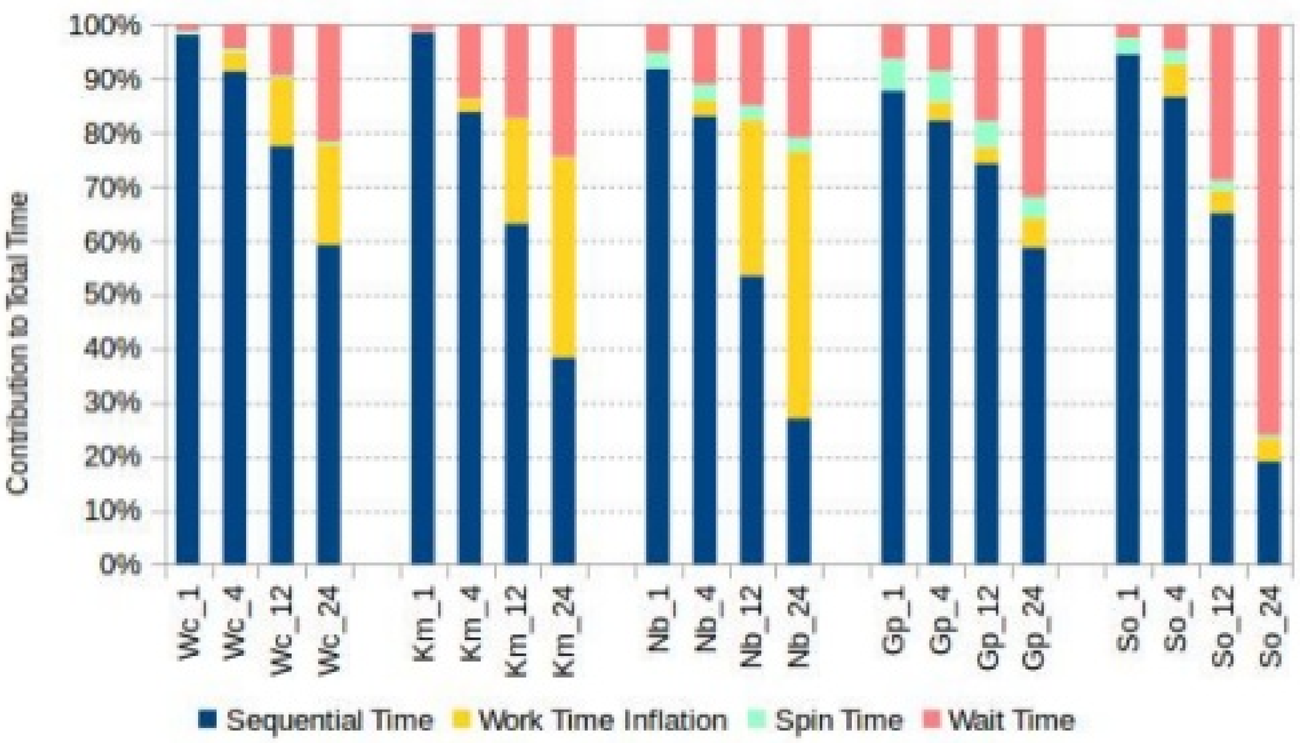
GC time does not scale linearly at larger datasets

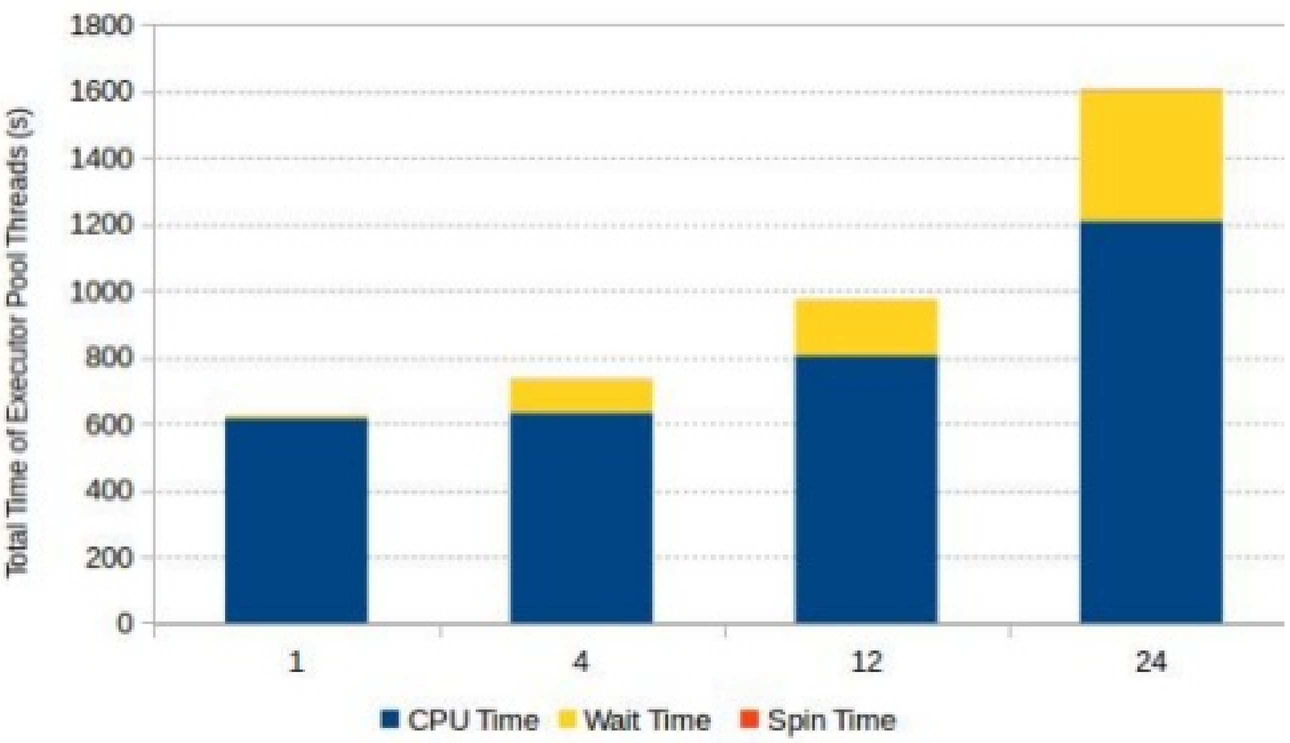
Is File I/O detrimental to performance ?



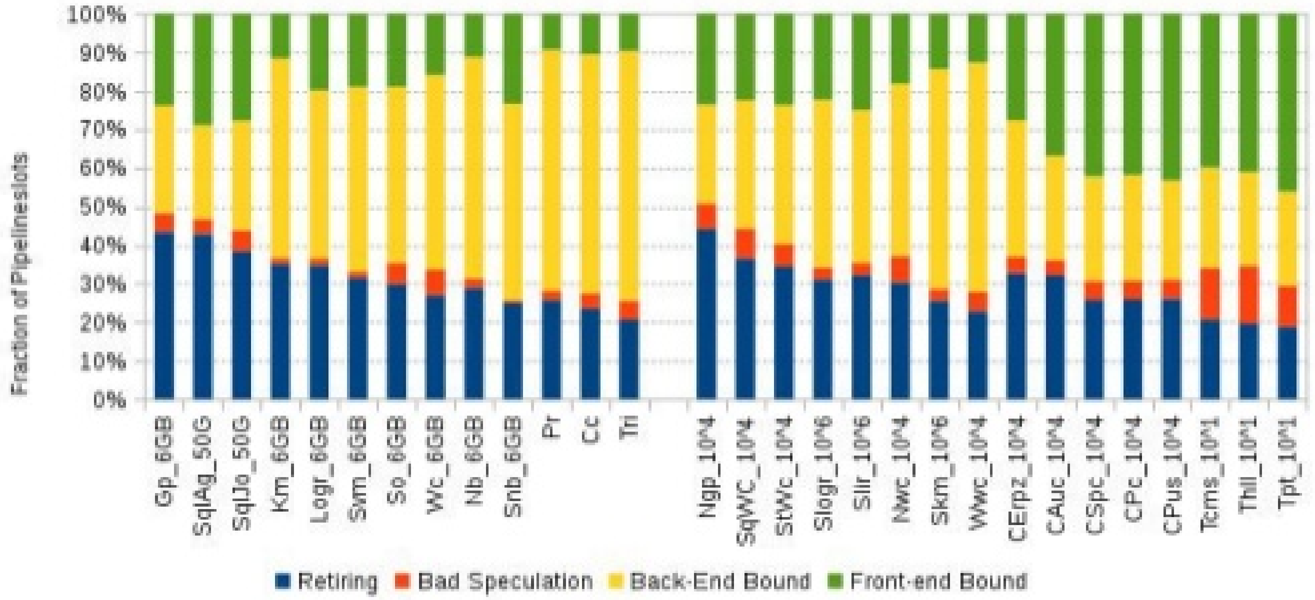
Fraction of file I/O increases by 25x in Sort respectively when input data is increased by 4x

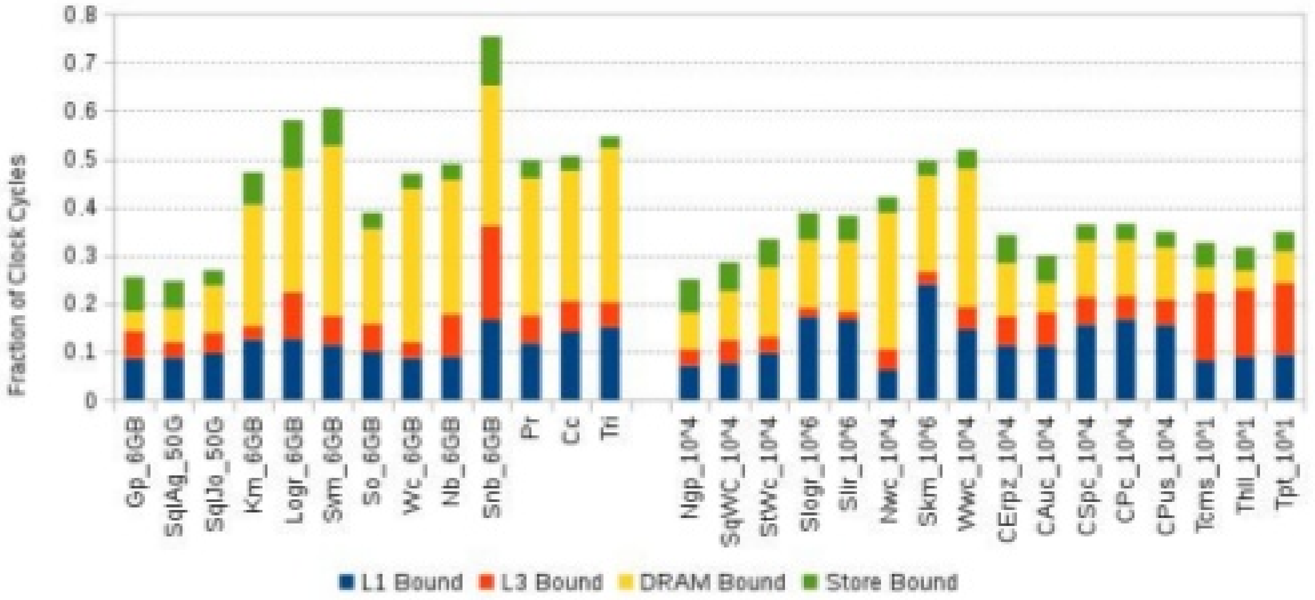
Is there work time inflation ?





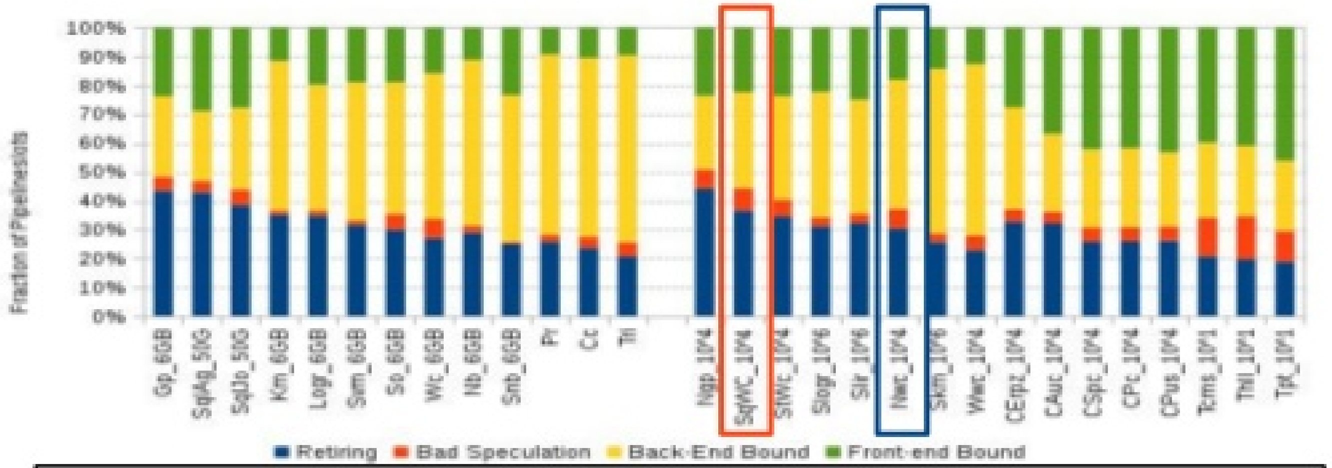
Are workloads DRAM Bound?



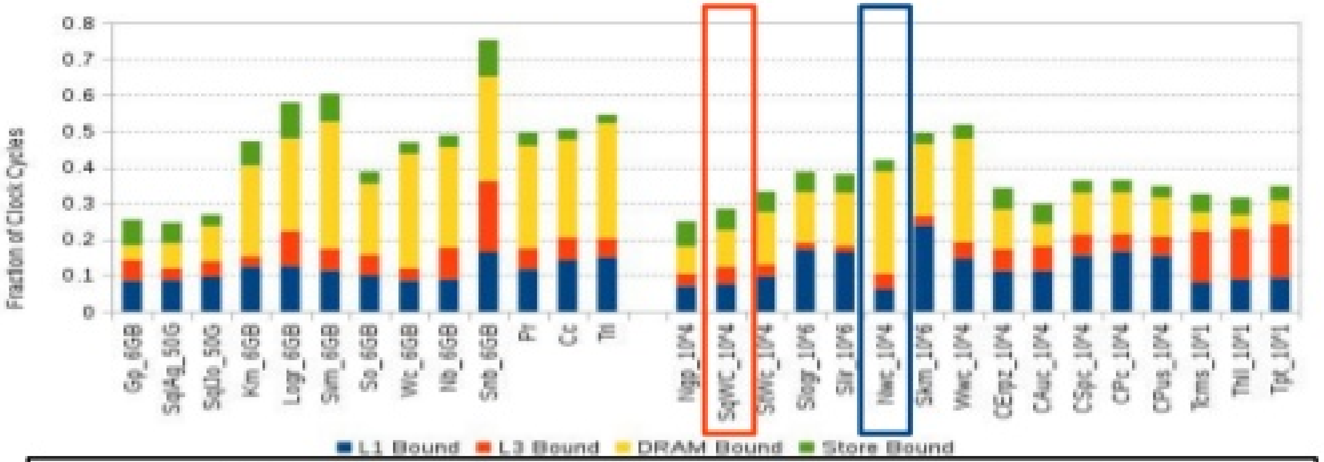


Poor instruction retirement due to frequent DRAM accesses

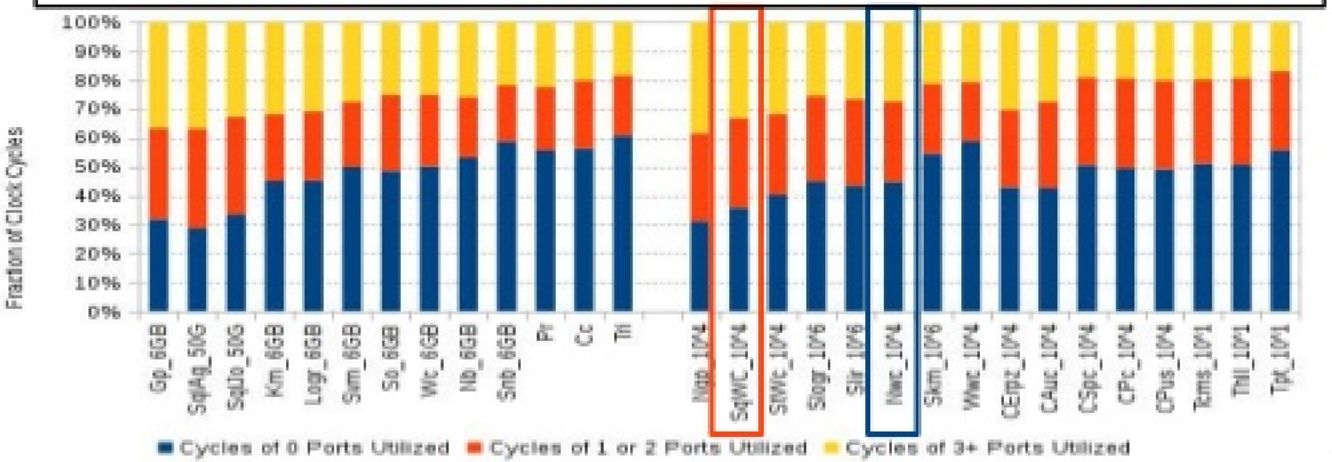
Do Dataframes perform better than RDDs at micro-architectural level?



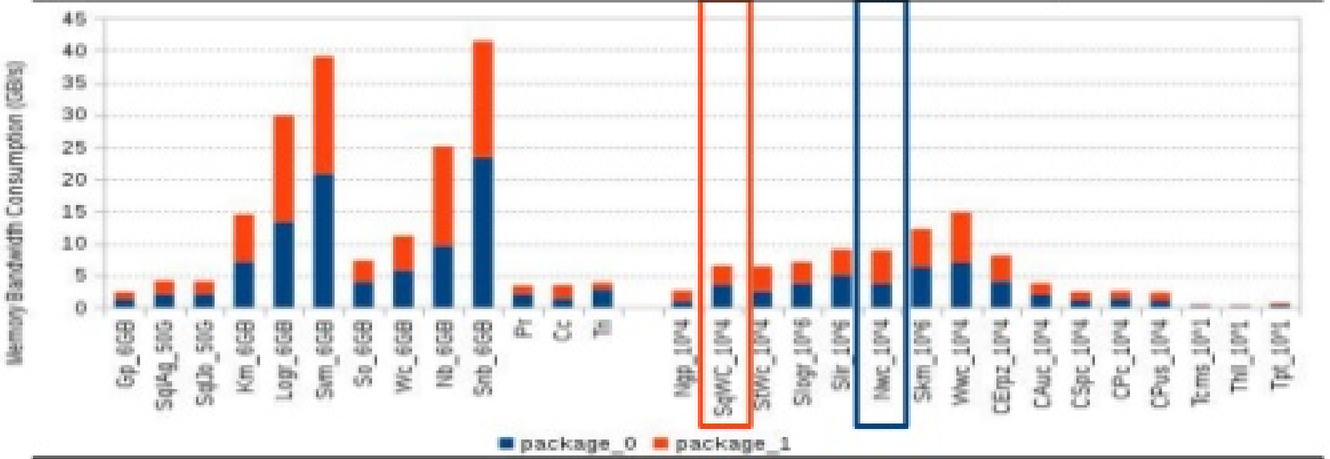
DataFrame exhibit 25% less back-end bound stalls



64% less DRAM bound stalled cycles



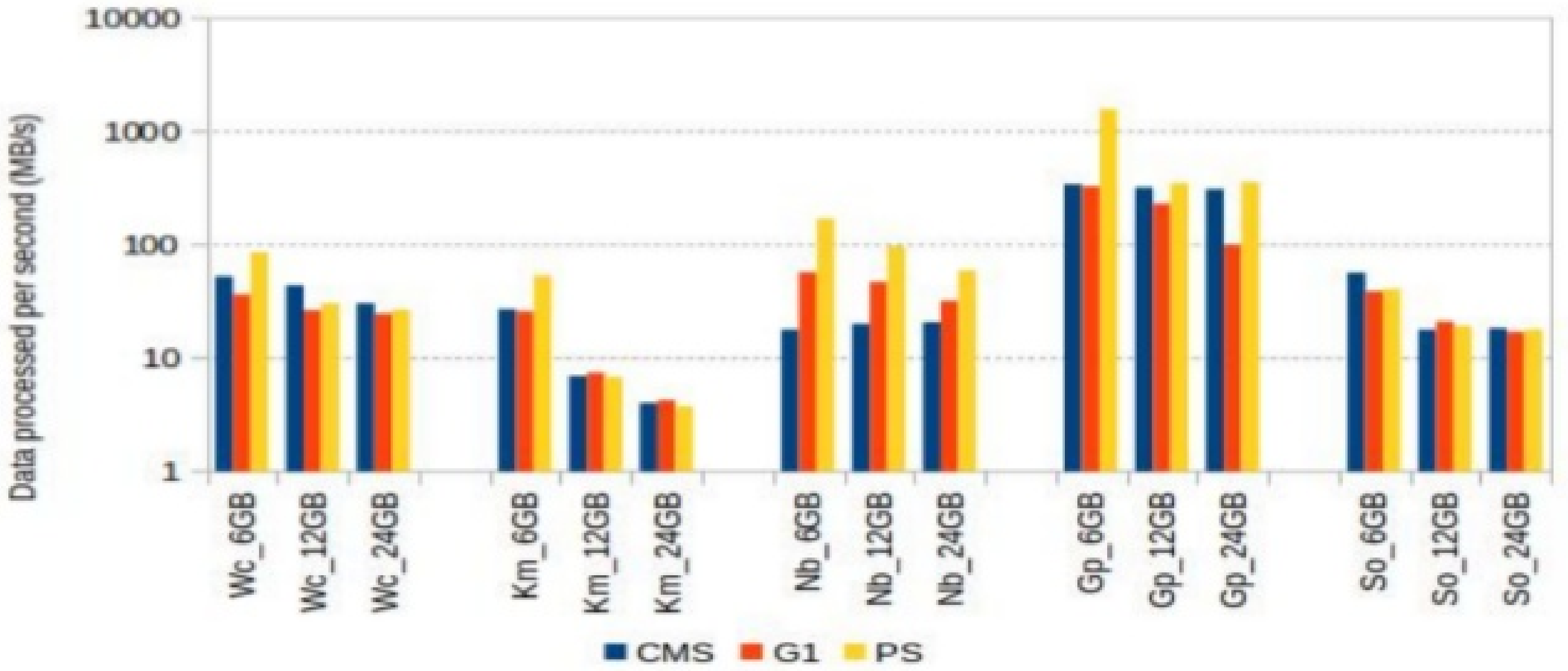
10% less starvation of execution resources



25% less BW consumption

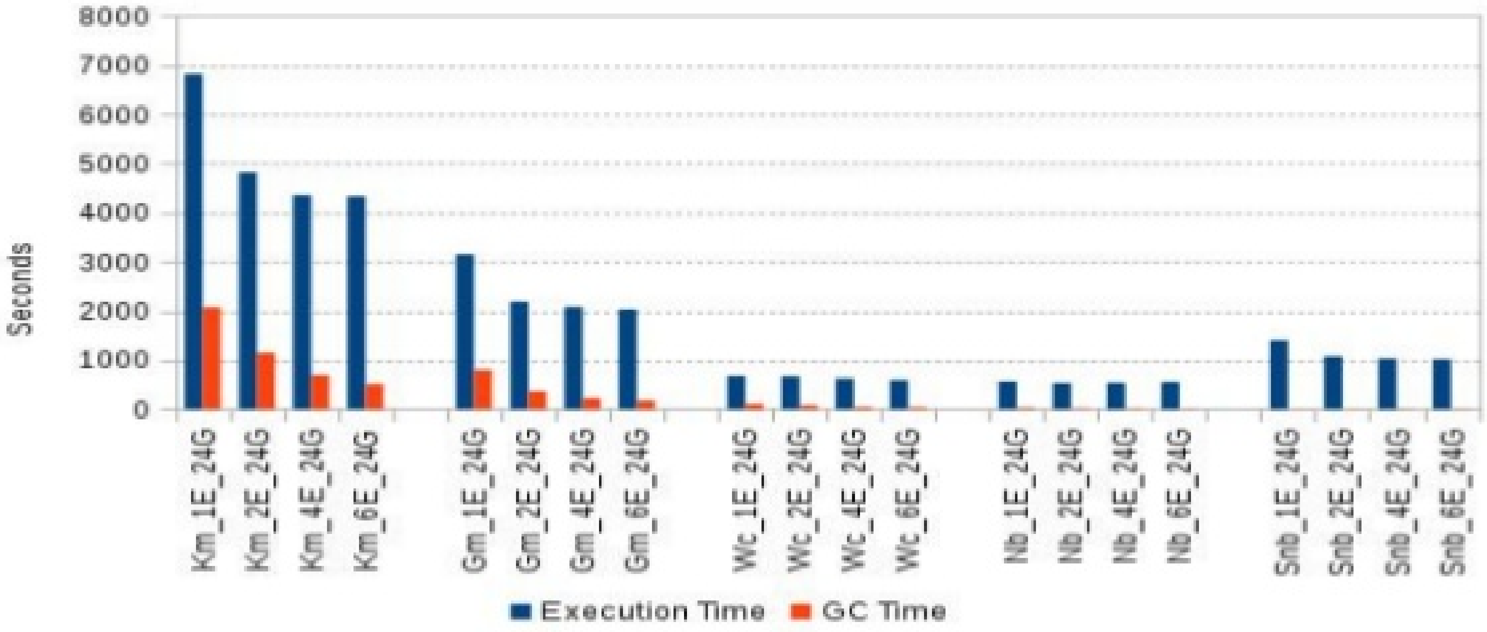
Dataframes have better micro-architectural performance than RDDs

The choice of Garbage Collector impact the data processing capability of the system



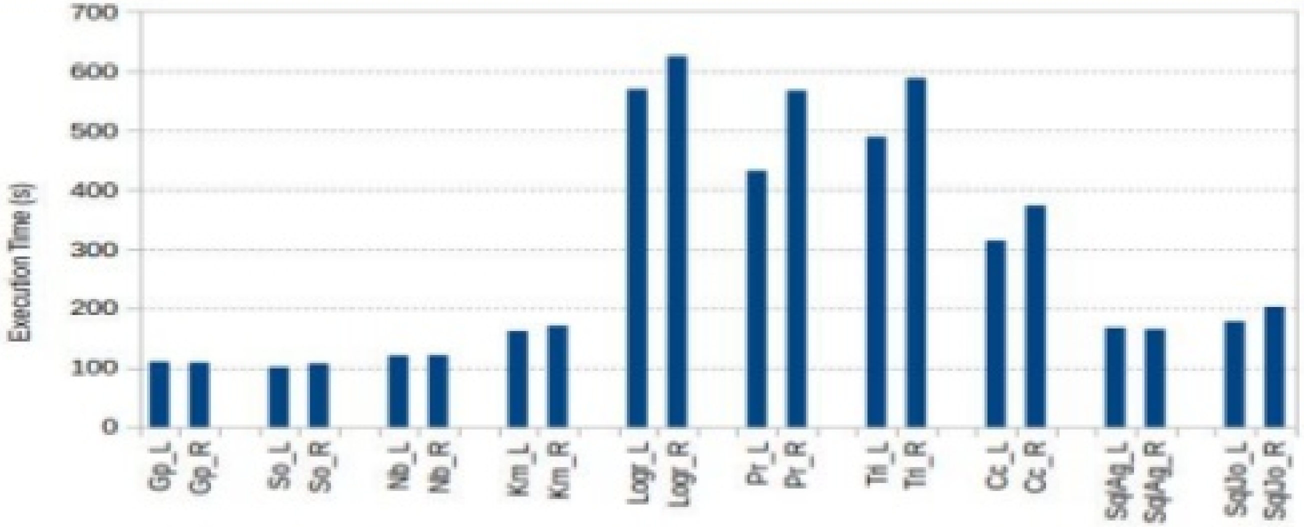
Improvement in DPS ranges from 1.4x to 3.7x on average in Parallel Scavenge as compared to G1

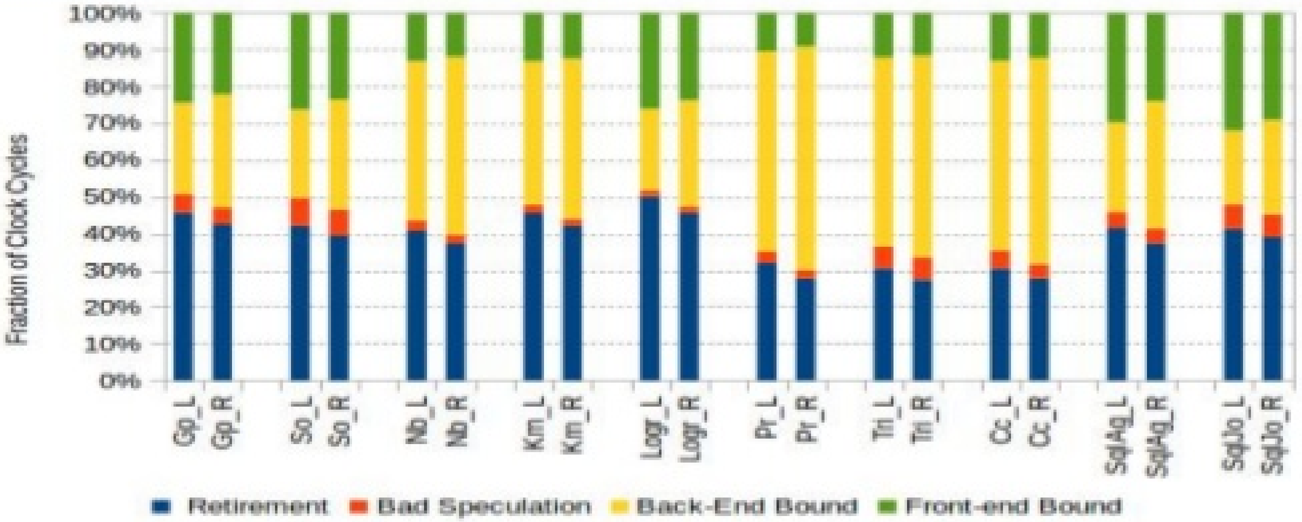
Multiple Small executors instead of single large executor

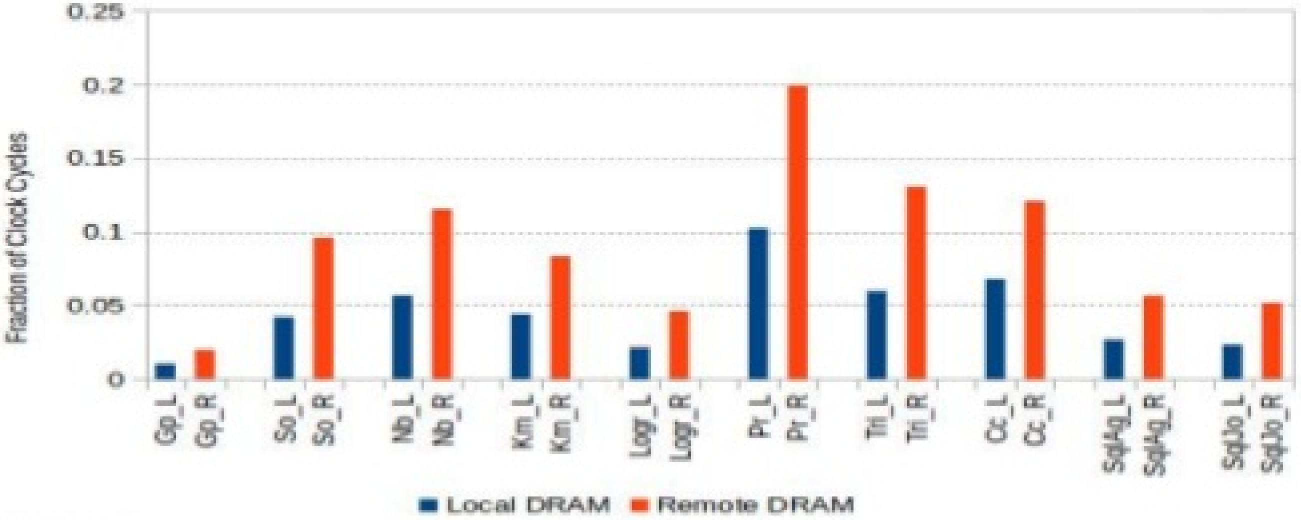


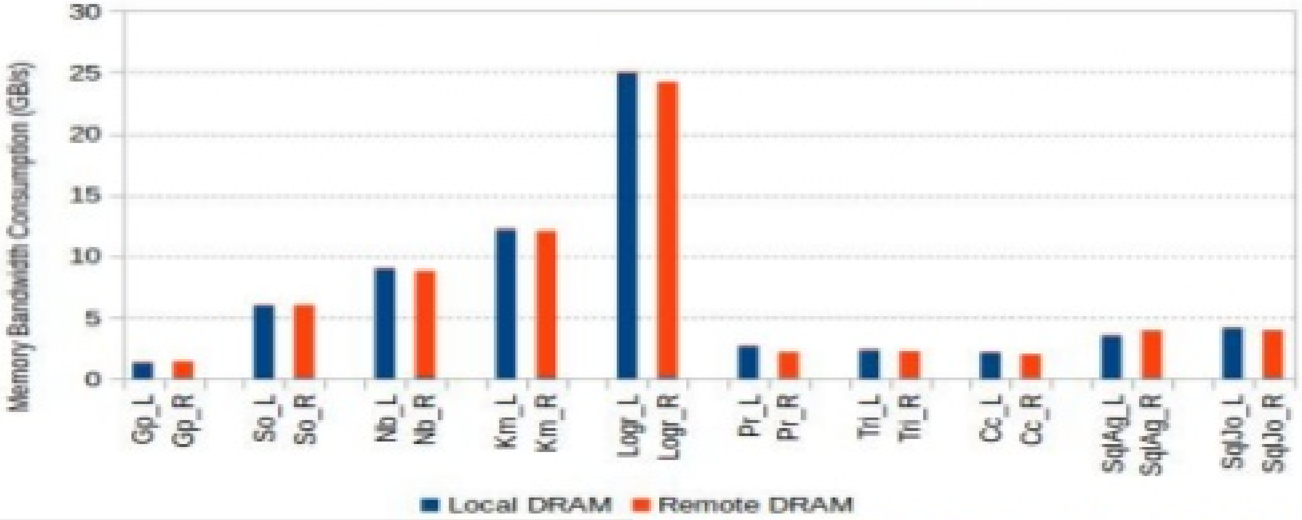
Multiple small executors can provide up-to 36% performance gain

NUMA Awareness





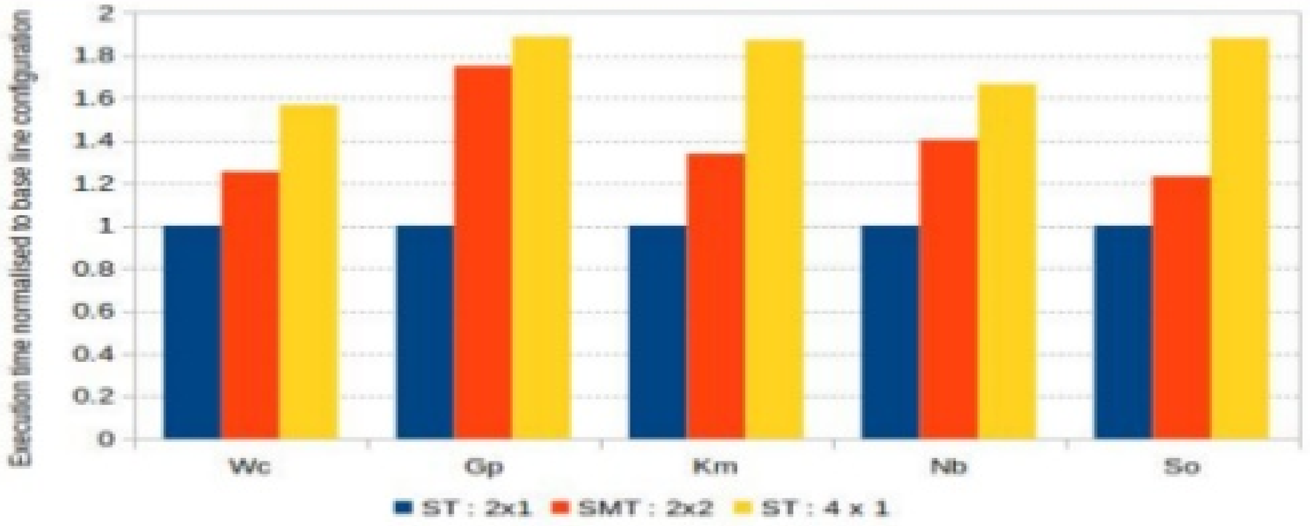


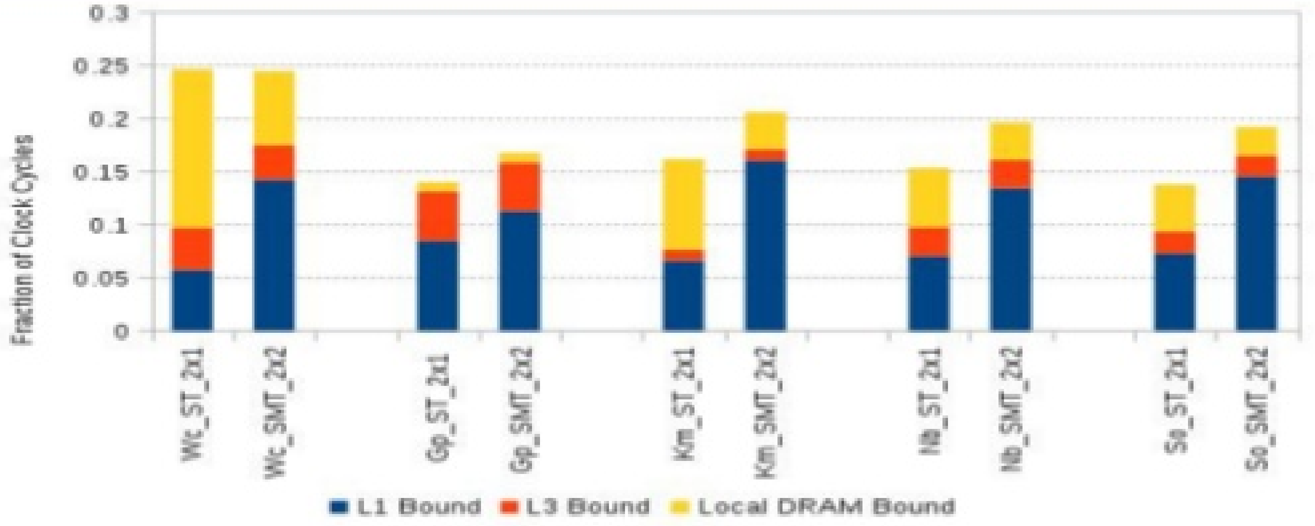


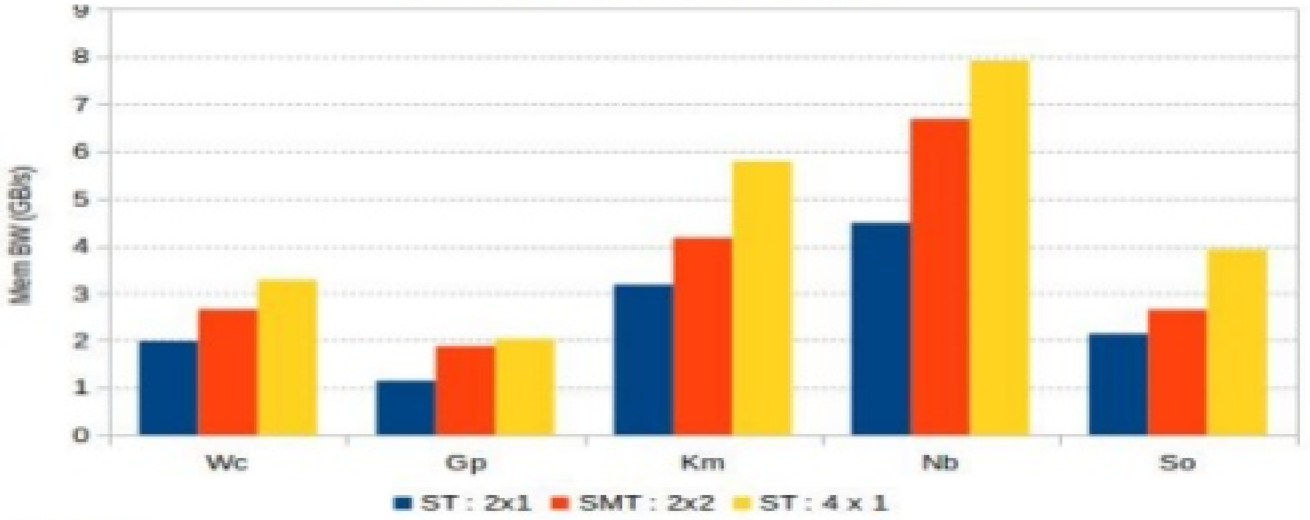
SPARK SUMMIT EUROPE 2016

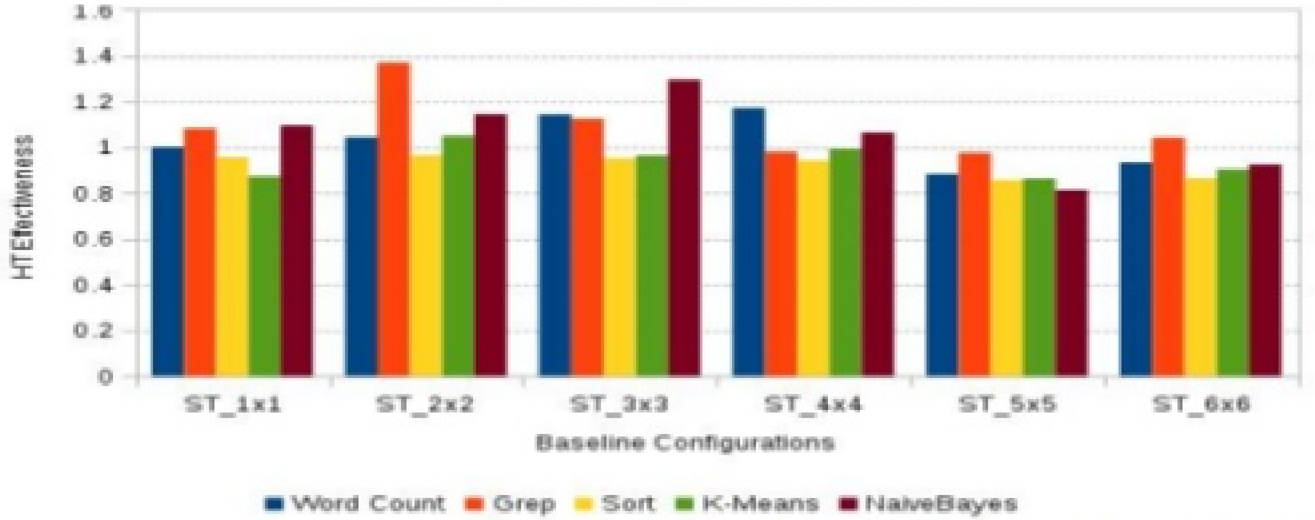
NUMA Awareness results in 10% speed up on average

Hyper Threading is effective





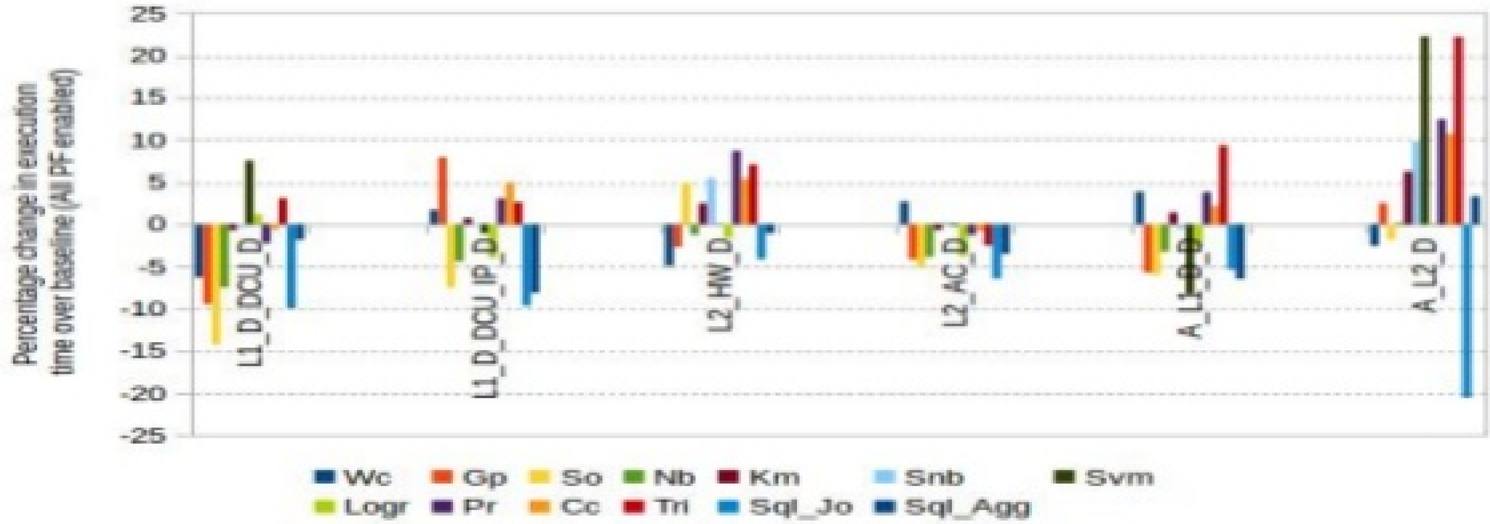


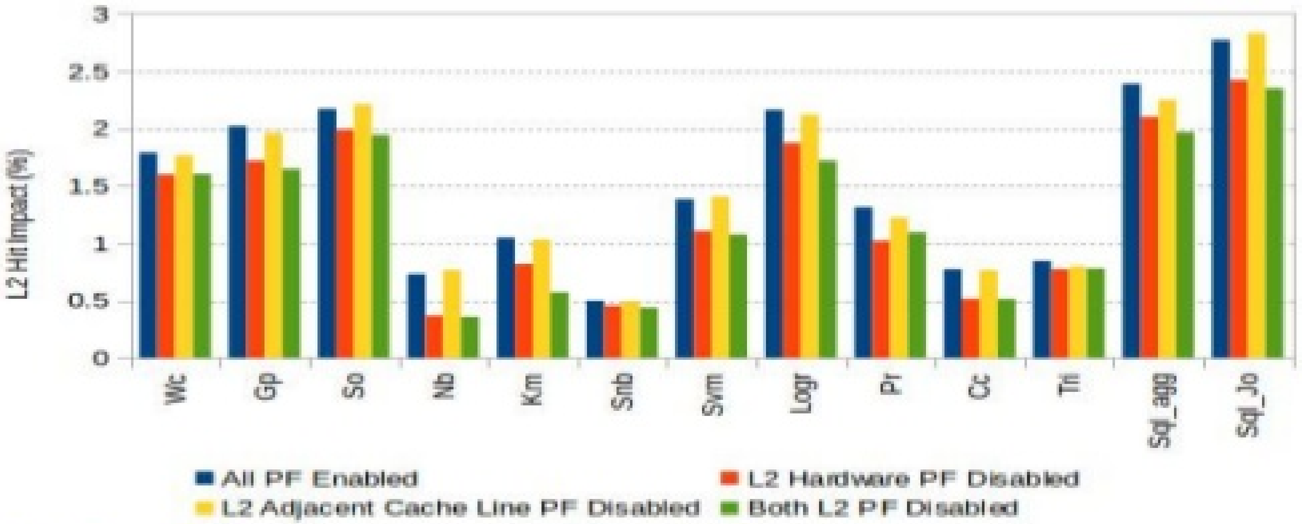


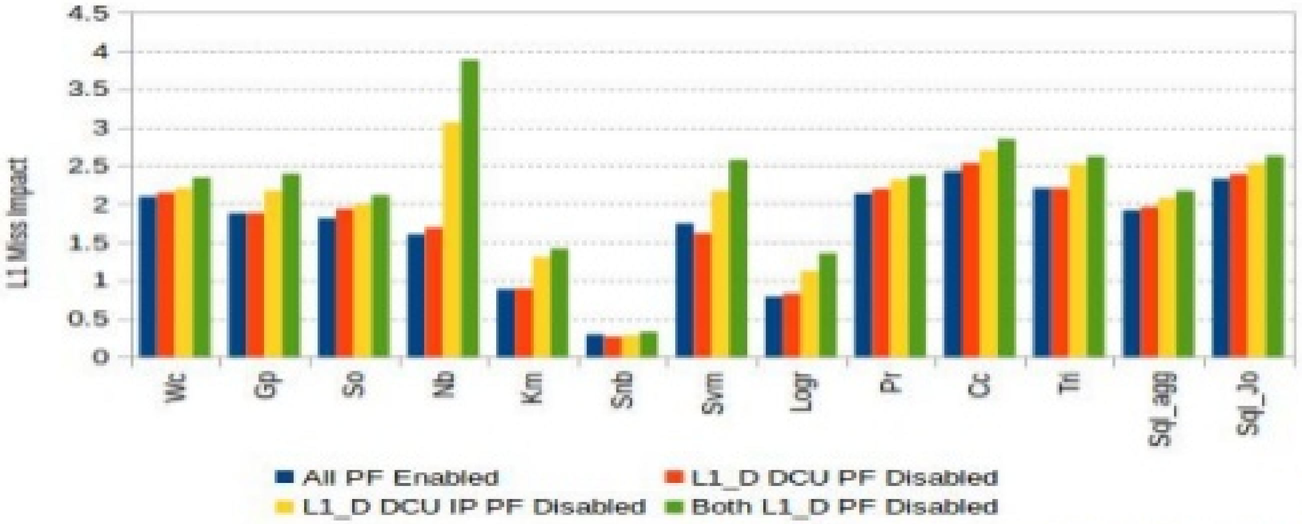
Hyper threading reduces the DRAM bound stalls by 50%

Disable next-line prefetchers

Prefetcher	Bit No. in MSR (0x1A4)	Description
L2 hardware prefetcher	0	Fetches additional lines of code or data into the L2 cache
L2 adjacent cache line prefetcher	1	Fetches the cache line that comprises a cache line pair(128 bytes)
DCU prefetcher	2	Fetches the next cache line into L1-D cache
DCU IP prefetcher	3	Uses sequential load history (based on Instruction Pointer of previous loads) to determine whether to prefetch additional lines





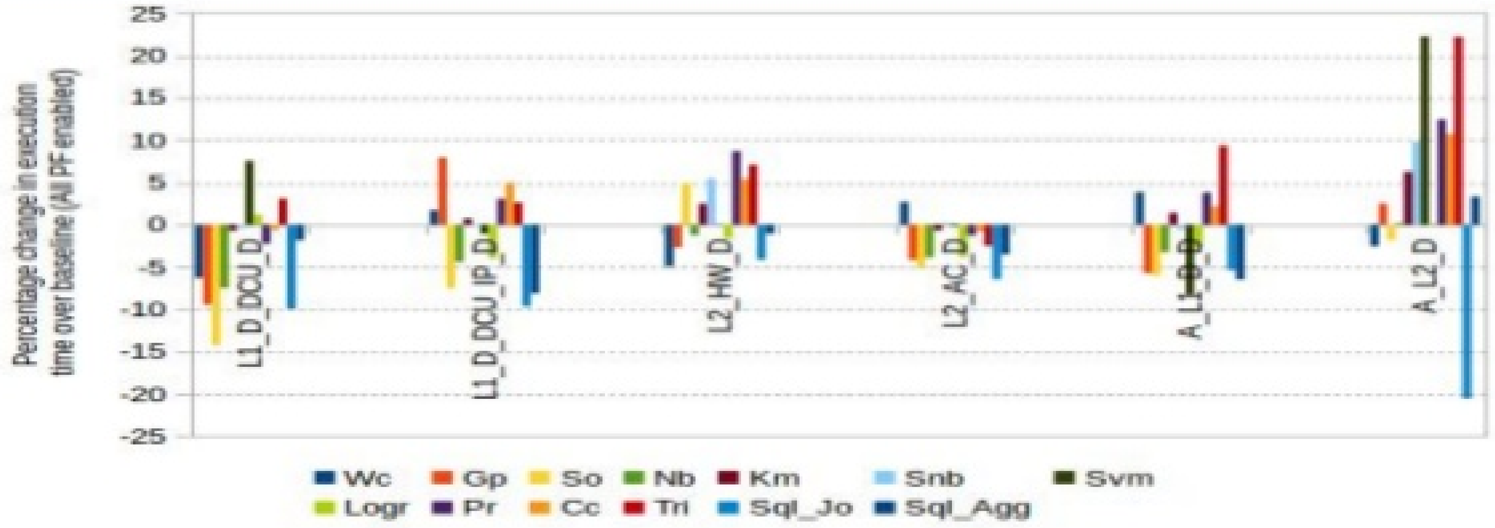


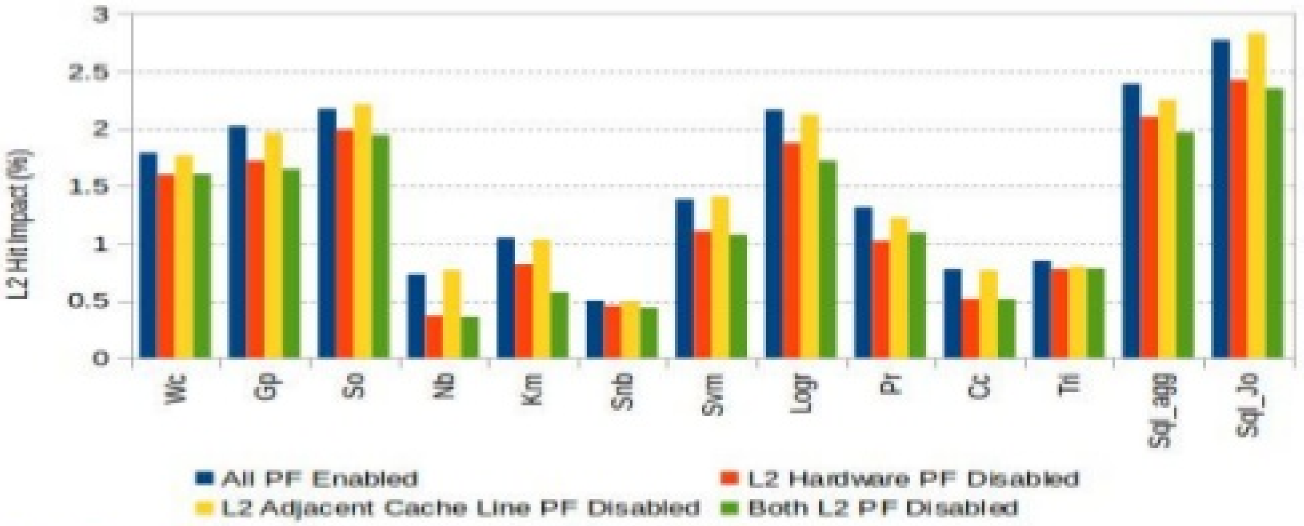
Disabling next-line prefetchers can improve the performance by 15%

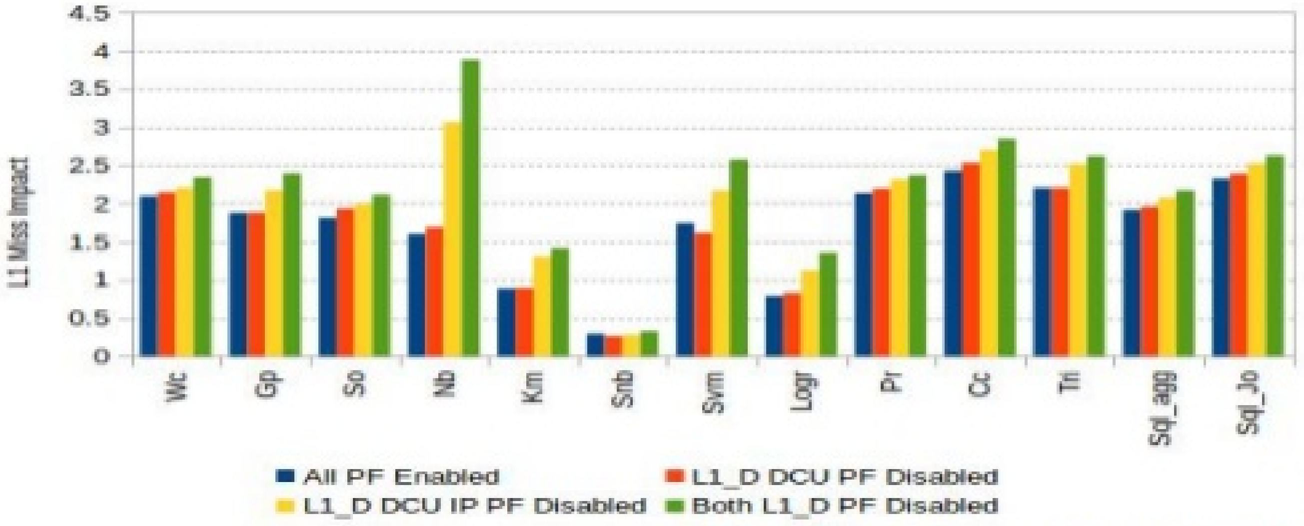
Use Near Data Computing Architecture

Disable next-line prefetchers

Prefetcher	Bit No. in MSR (0x1A4)	Description
L2 hardware prefetcher	0	Fetches additional lines of code or data into the L2 cache
L2 adjacent cache line prefetcher	1	Fetches the cache line that comprises a cache line pair(128 bytes)
DCU prefetcher	2	Fetches the next cache line into L1-D cache
DCU IP prefetcher	3	Uses sequential load history (based on Instruction Pointer of previous loads) to determine whether to prefetch additional lines







Disabling next-line prefetchers can improve the performance by 15%

2D PIM vs 3D Stacked PIM

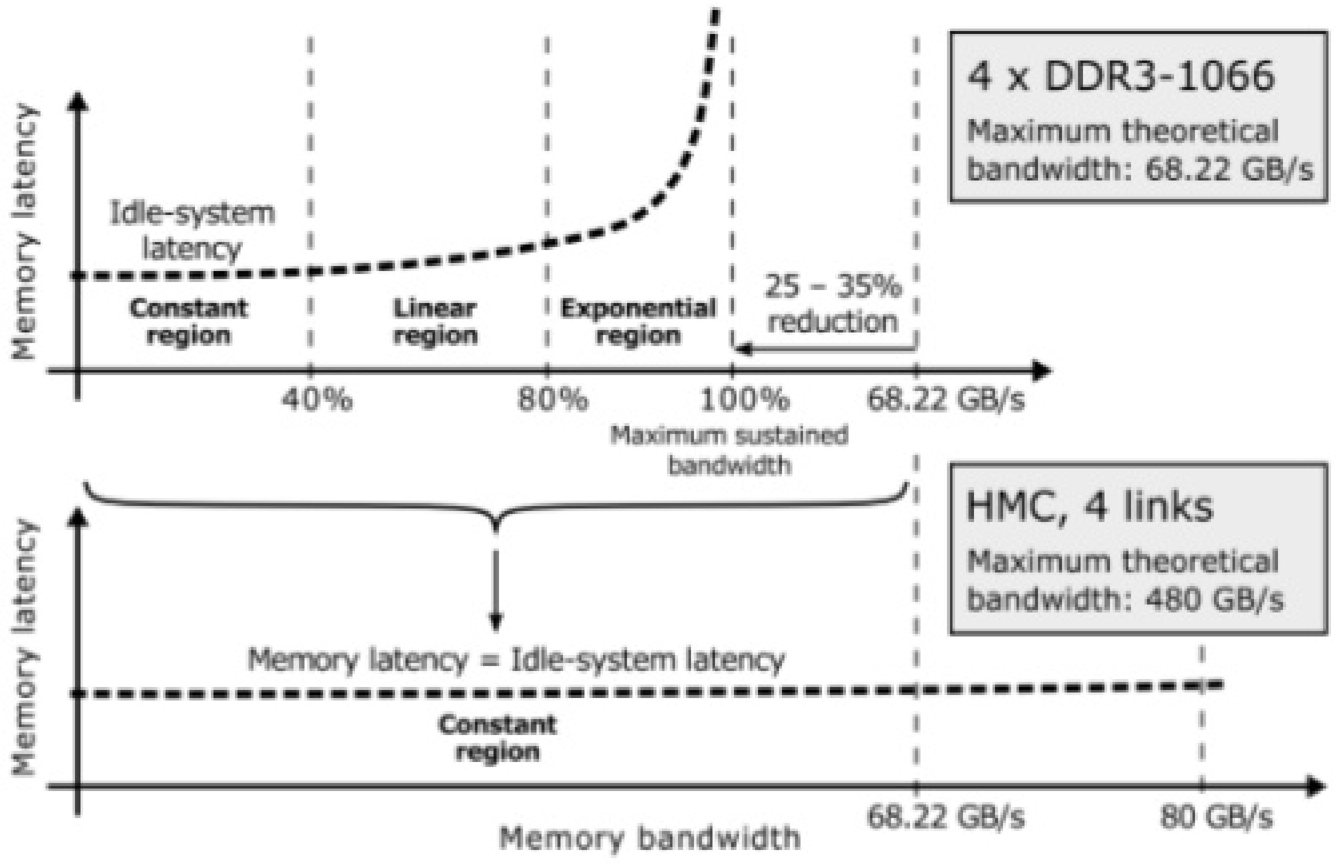
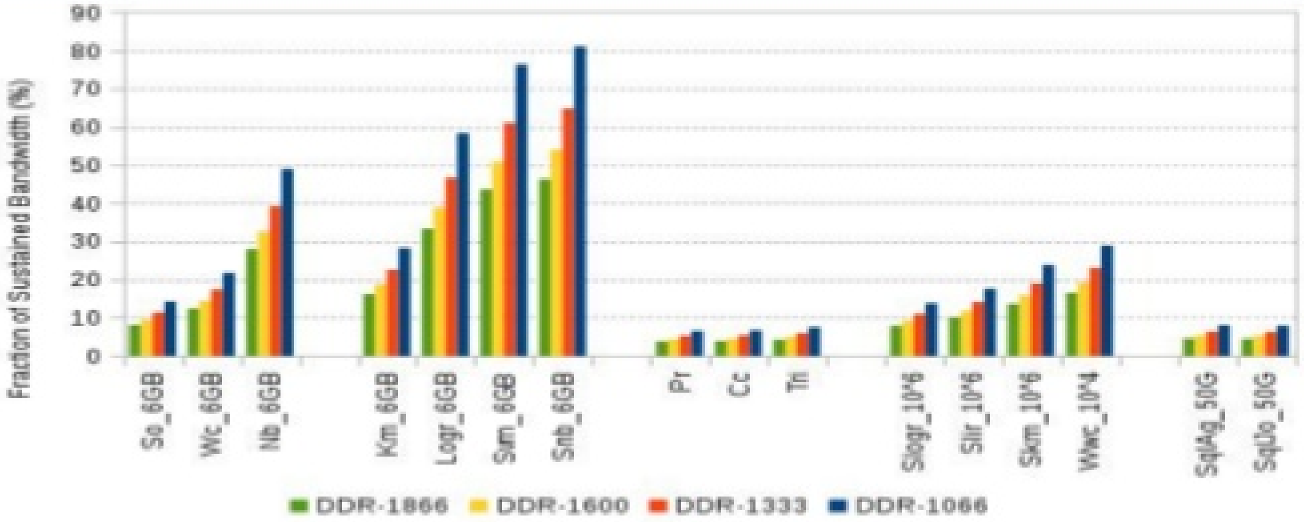
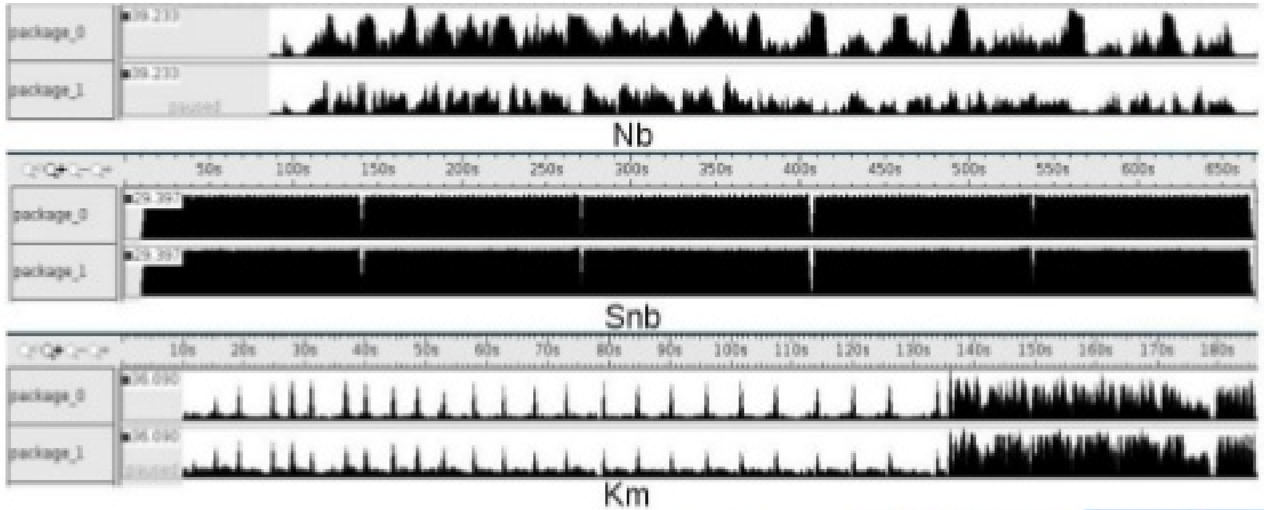


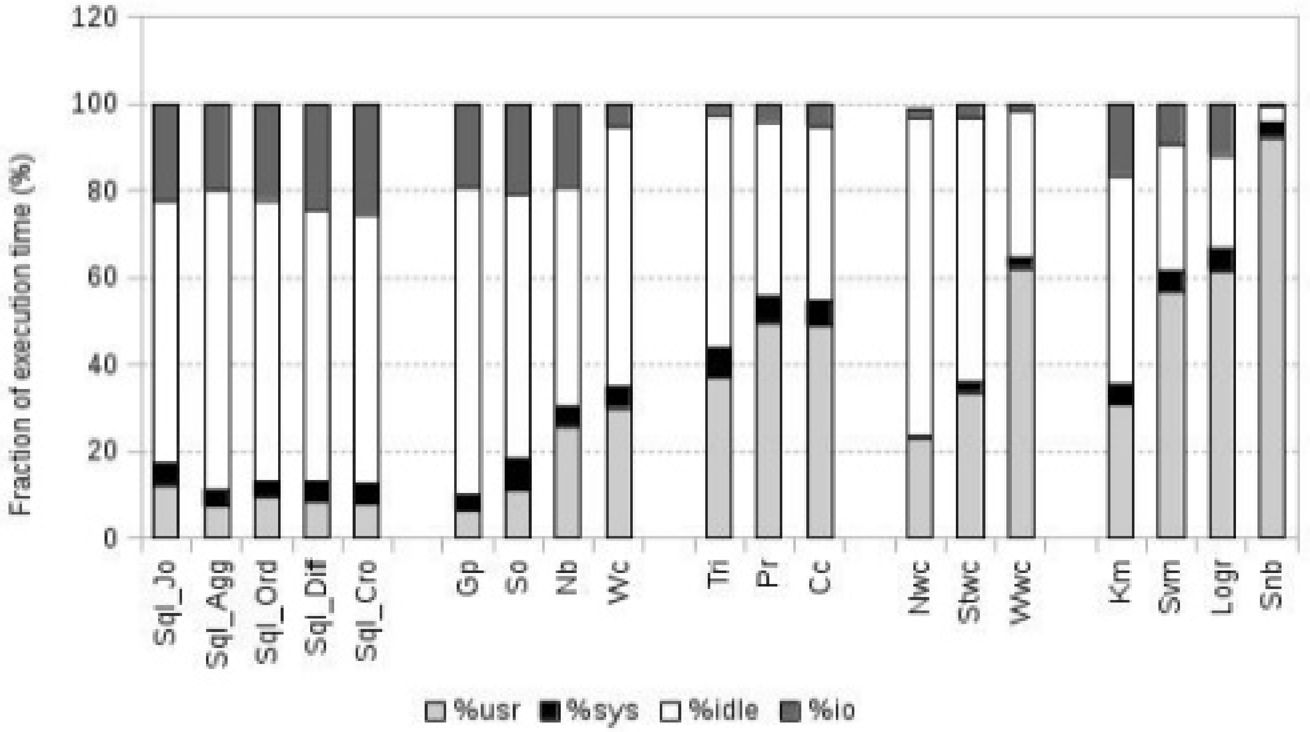
Figure 4: Bandwidth-latency curves of DDR3 and HMC systems

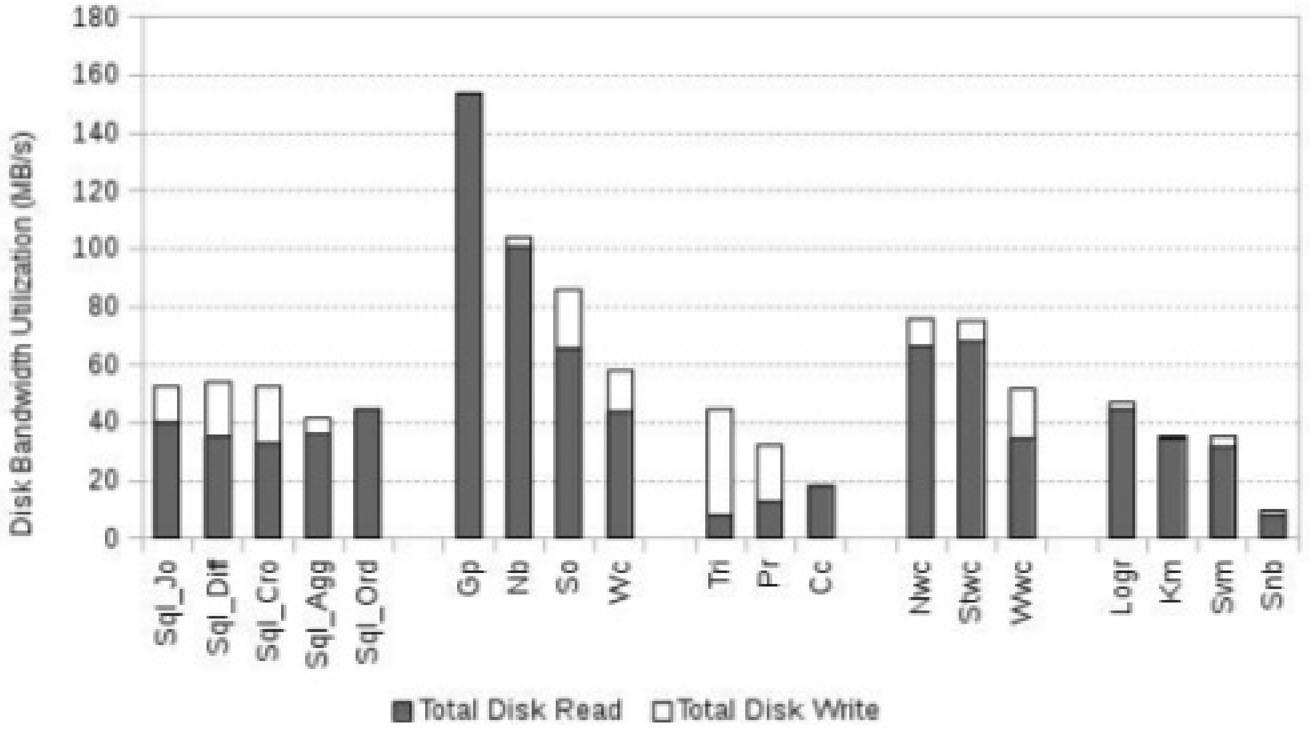




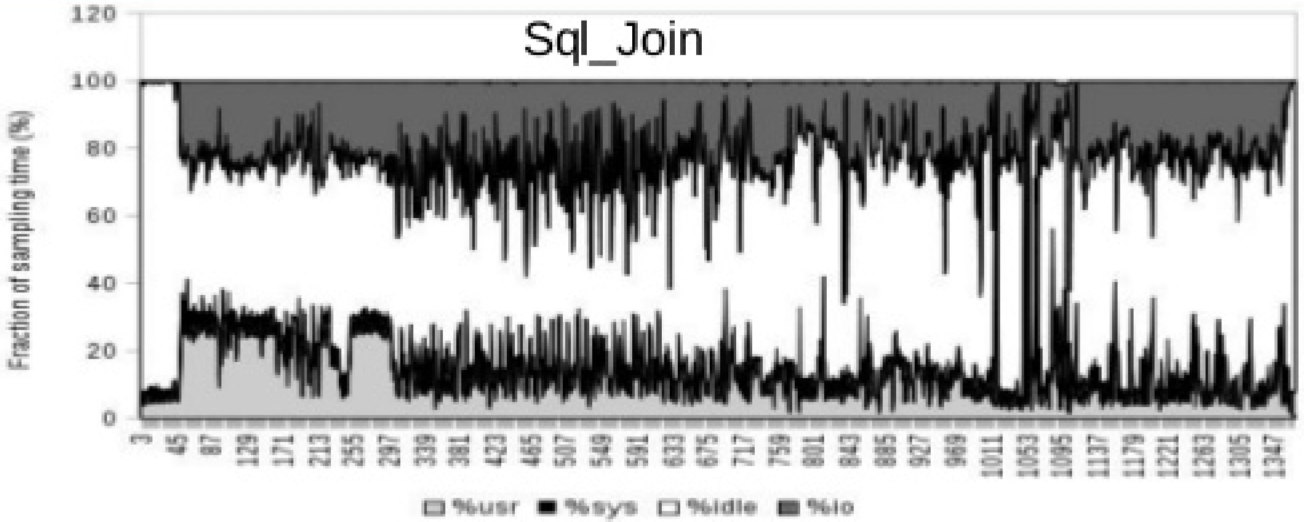
High Bandwidth Memories are not required for Spark

Sub-setting the workloads for ISP

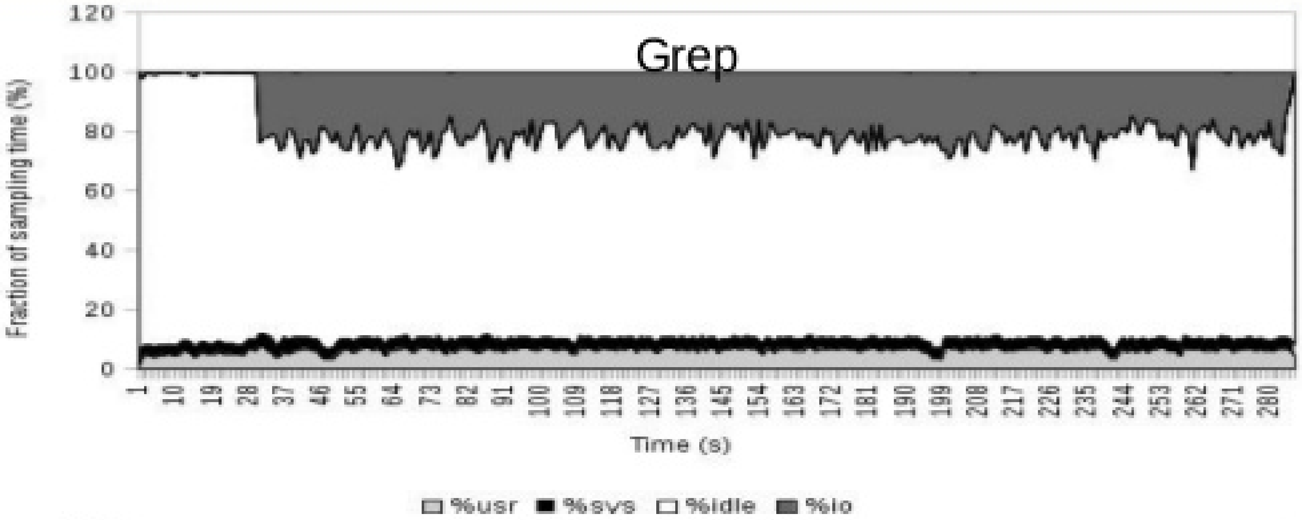




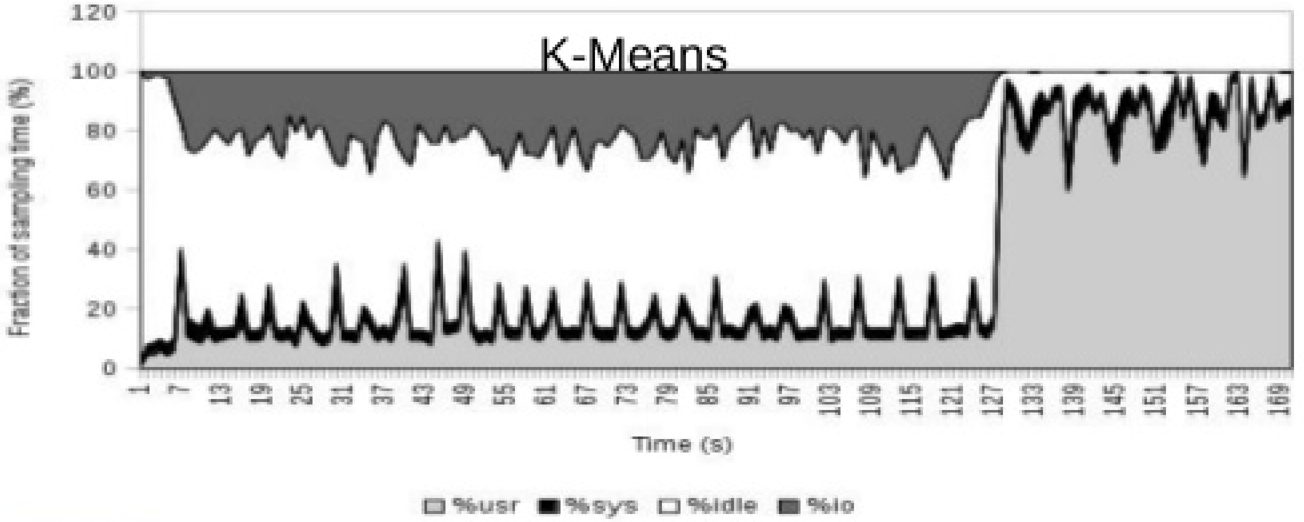
Cont..



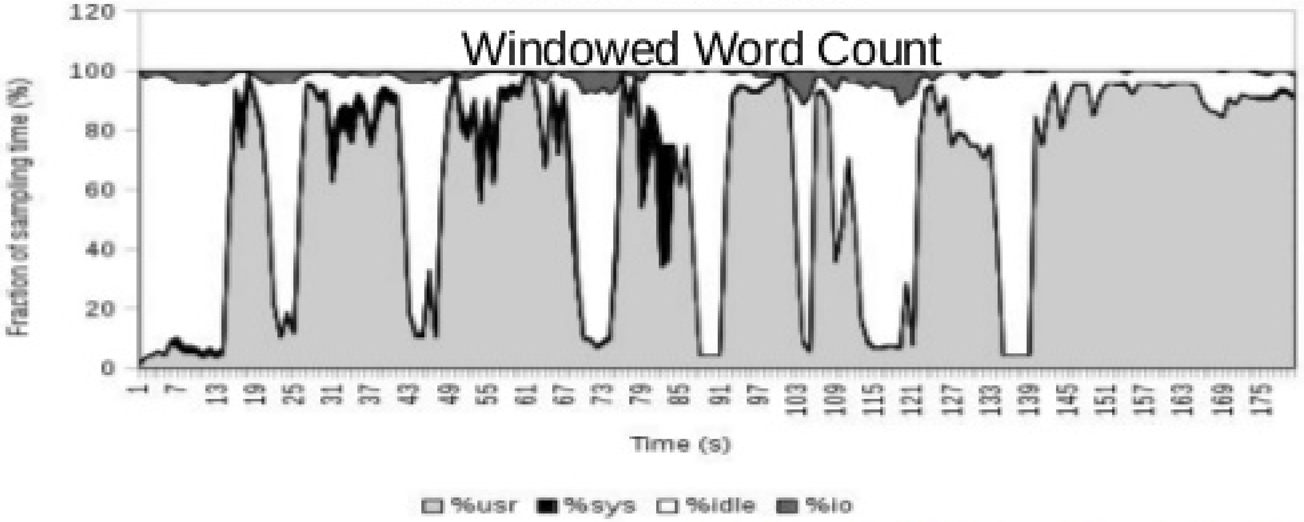
Sql_Join
Fraction of sampling time (%)
%usr
%sys
%idle
%io



Grep
Fraction of sampling time (%)
Time (s)
%usr
%sys
%idle
%io



K-Means
Fraction of sampling time (%)
Time (s)
%usr
%sys
%idle
%io



Windowed Word Count
Fraction of sampling time (%)
Time (s)
%usr
%sys
%idle
%io

Cont..

- In-storage processing (ISP) is more suitable for Spark SQL queries.
- Implications of 2D processing in memory (PIM) better match the characteristics of Graph-X and Spark Streaming workloads.
- A hybrid ISP plus 2D PIM architecture is required for Spark MLlib workloads.

Summary of our work

Poor Multi-core Scalability of data analytics with Spark

Work Time Inflation

DRAM Bound

Thread Level Load Imbalance

Wait Time in I/O

GC overhead

NUMA Awareness

Hyper Threaded Cores

No next-line prefetchers

Lower DRAM speed

Future node based on Hybrid ISP + 2D PIM

PS over G1 GC

Multiple Small executors

Problems Identified

Solutions Proposed

Further Reading

- Performance characterization of in-memory data analytics on a modern cloud server, in 5th IEEE Conference on Big Data and Cloud Computing, 2015 (Best Paper Award).
- How Data Volume Affects Spark Based Data Analytics on a Scale-up Server in 6th Workshop on Big Data Benchmarks, Performance Optimization and Emerging Hardware (BpoE), held in conjunction with VLDB 2015, Hawaii, USA .
- Micro-architectural Characterization of Apache Spark on Batch and Stream Processing Workloads, in 6th IEEE Conference on Big Data and Cloud Computing, 2016.
- Node Architecture Implications for In-Memory Data Analytics in Scale-in Clusters in 3rd IEEE/ACM Conference in Big Data Computing, Applications and Technologies, 2016.
- Implications of In-Memory Data Analytics with Apache Spark on Near Data Computing Architectures (under submission).

THANK YOU.

Email: ajawan@kth.se
Profile: www.kth.se/profile/ajawan/

Acknowledgements:
Mats Brorsson(KTH)
Vladimir Vlassov(KTH)
Eduard Ayguade(UPC/BSC)

SUMMIT
EUROPE 2016