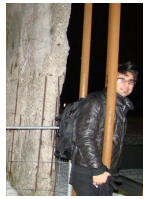
Near Data Computing Architectures: Opportunities and Challenges for Apache Spark

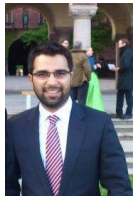
Ahsan J. Awan (KTH)

#EUres10

About me ?






Born in Pakistan — 1988

B.E. MTS NUST, Pakistan — 2010

EMECS, TUKL, Germany — 2011

EMECS, UoS, UK — 2012

Lecturer, NUST, Pakistan — 2013

EMJD-DC, KTH/SICS, Sweden — 2014

EMJD-DC, UPC/BSC, Spain — 2015

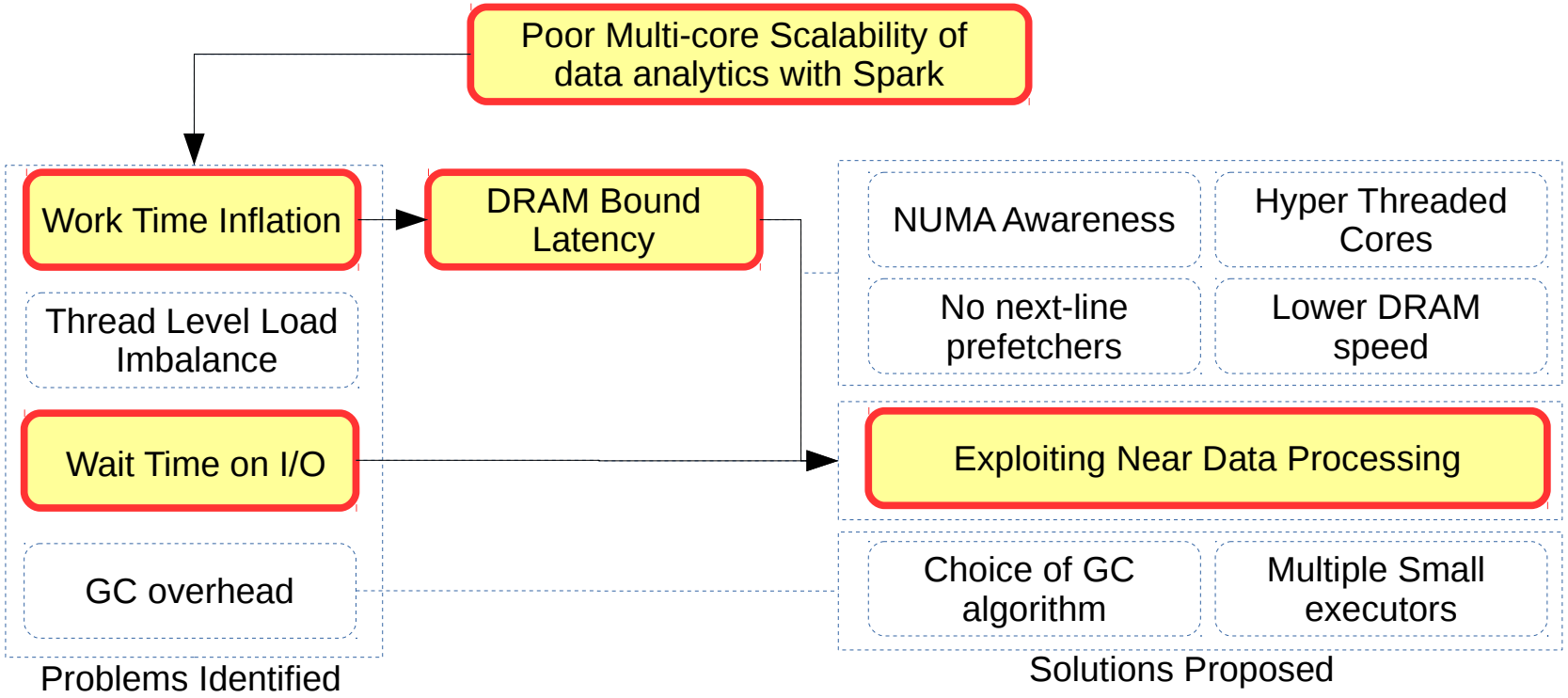
PhD Intern, Recore Netherlands — 2016

PhD Intern, IBM Research, Japan — 2017





A Quick Recap from last year ?



https://spark-summit.org/eu-2016/events/performance-characterization-of-apache-spark-on-scale-up-servers/

Further Reading ?

- Project Night-King: Improving the single performance for Apache Spark using Near Data Processing Architectures.
- Identifying the potential of Near Data Computing Architectures for Apache Spark in Memory Systems Symposium, 2017.

- Node Architecture Implications for In-Memory Data Analytics in Scale-in Clusters in IEEE/ACM Conference in Big Data Computing, Applications and Technologies, 2016.
- Micro-architectural Characterization of Apache Spark on Batch and Stream Processing Workloads, in IEEE Conference on Big Data and Cloud Computing, 2016.
- 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 .
- Performance characterization of in-memory data analytics on a modern cloud server, in IEEE Conference on Big Data and Cloud Computing, 2015 (Best Paper Award).

Exploiting NDP/Moving compute closer to data ?

1. Processing in Memory
2. In-Storage Processing

Improve the performance by reducing costly data movements back and forth between the CPUs and Memories

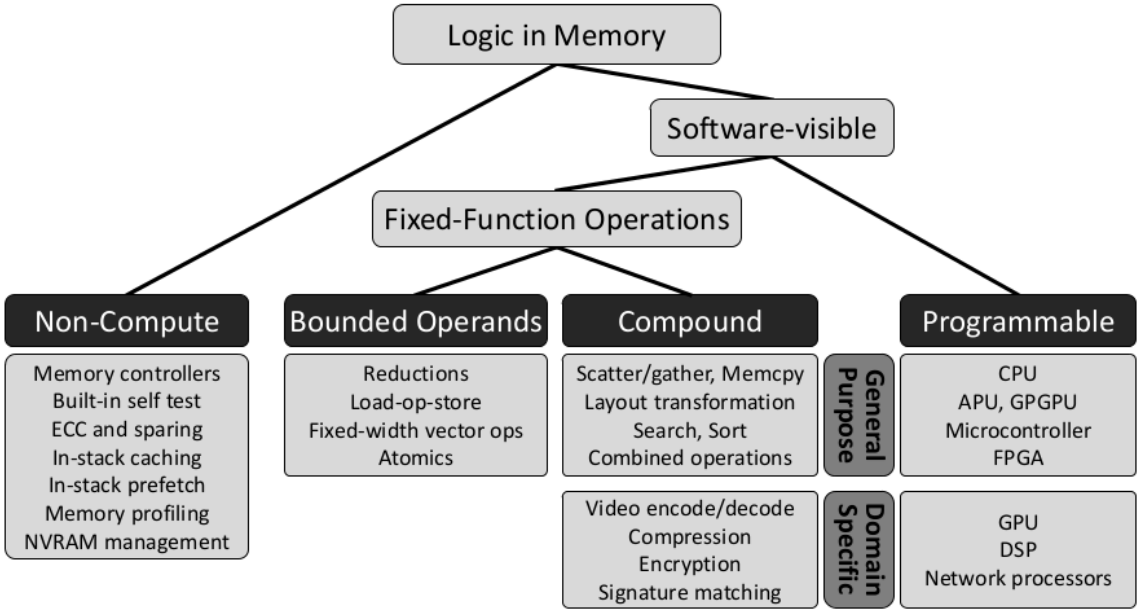
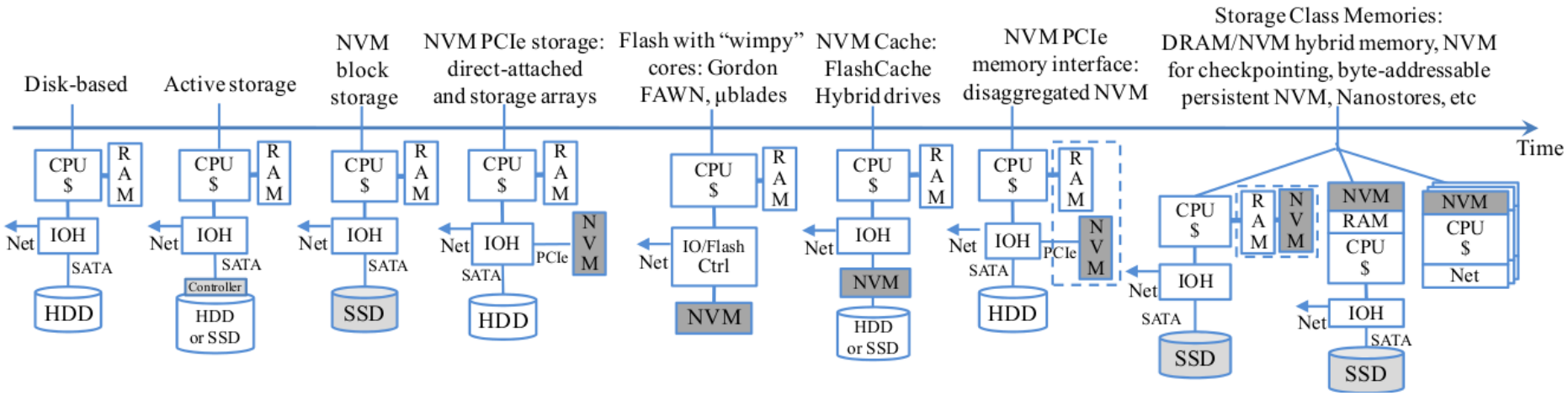


Figure 1: Taxonomy of processing in memory.

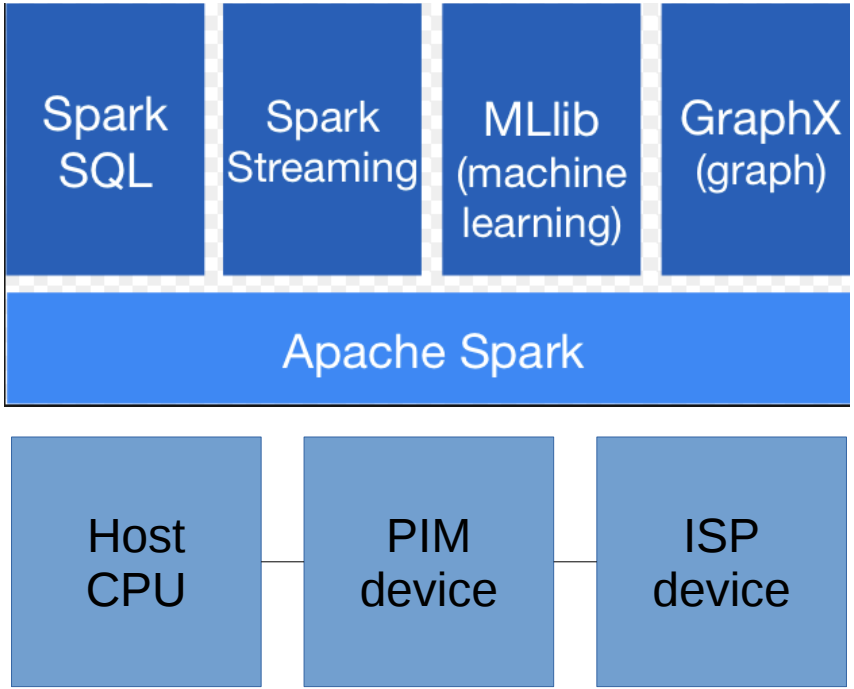
Loh et al. A processing in memory taxonomy and a case for studying fixedfunction pim. In Workshop on Near-Data Processing (WoNDP), 2013.

Trends of Integrating NVM in the System Architecture ?



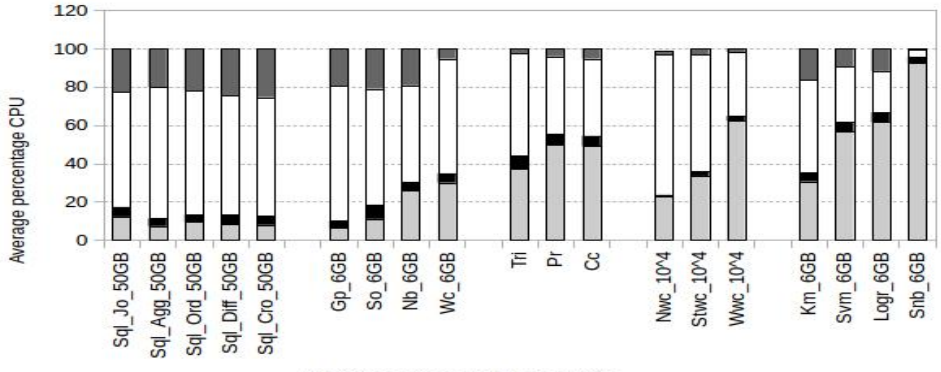
Chang et al. A limits study of benefits from nanostore-based future data-centric system architectures. In Computing Frontiers 2012

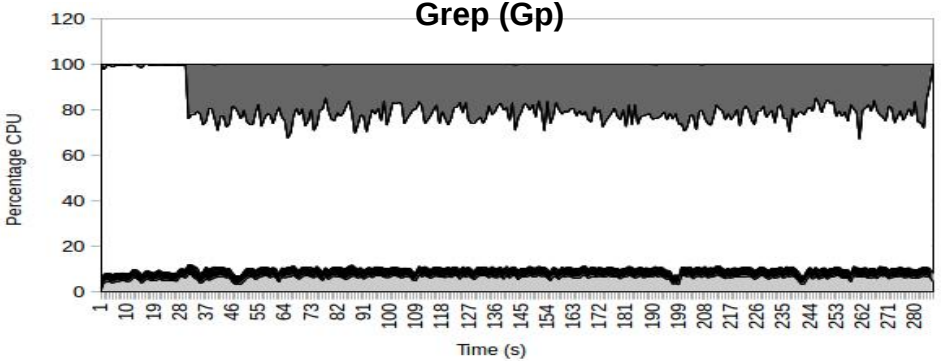
Can Spark workloads benefit from Near data processing ?

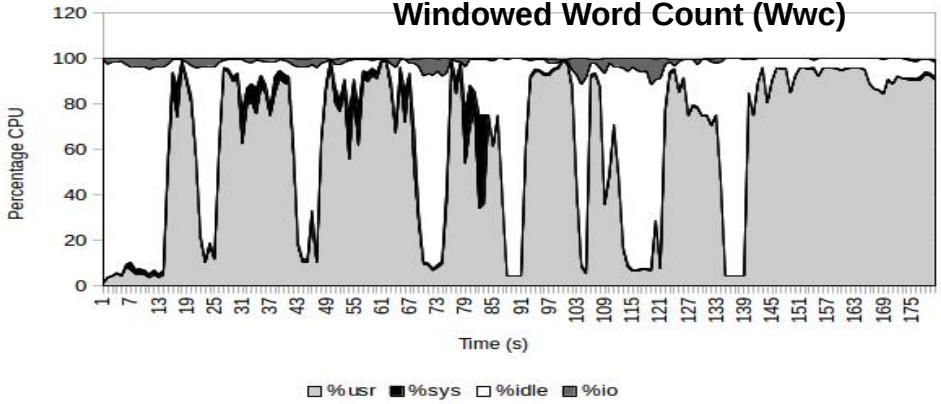


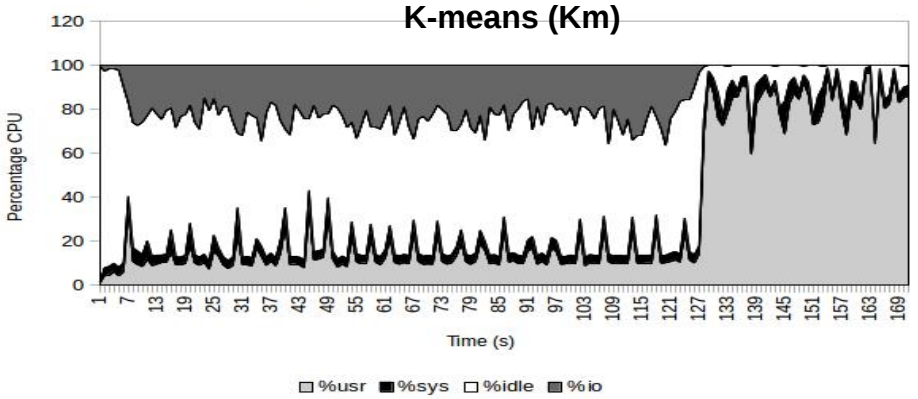
Project: NightKing

The case for in-storage processing ?









The case for 2D integrated PIM instead of 3D Stacked PIM ?

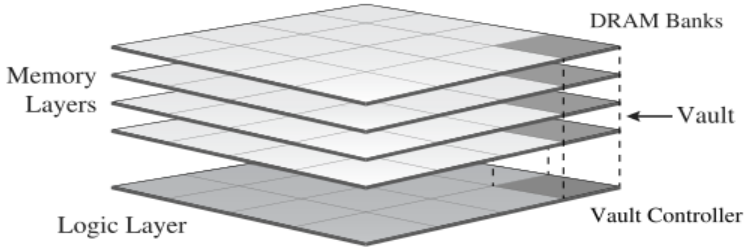
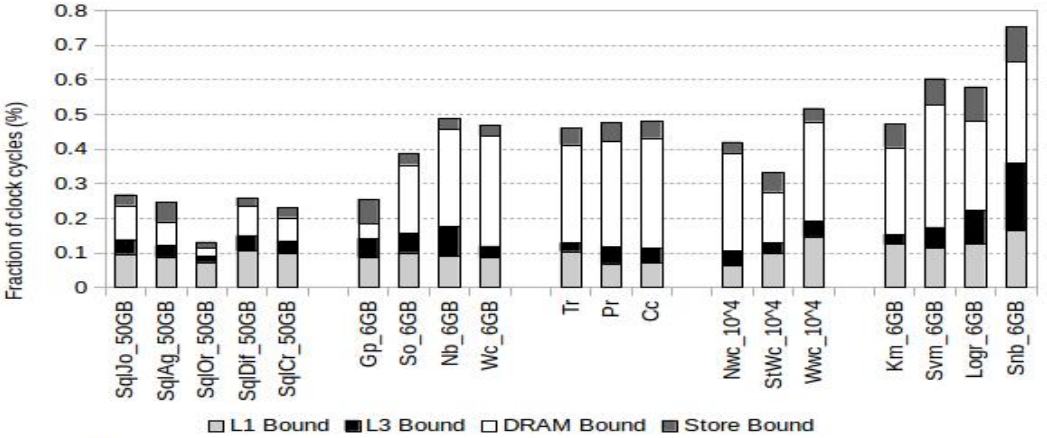


Fig. 1. Structure of an HMC.



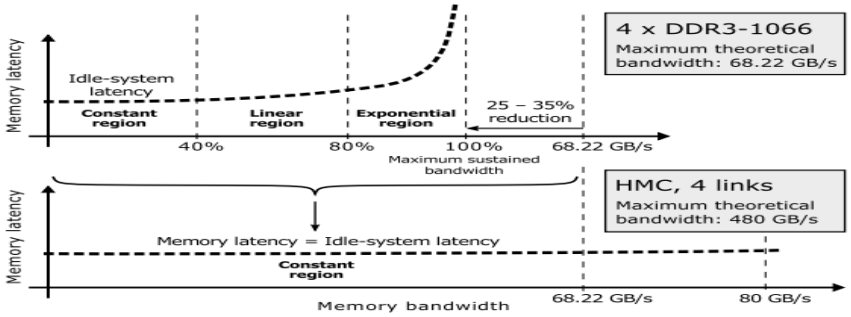
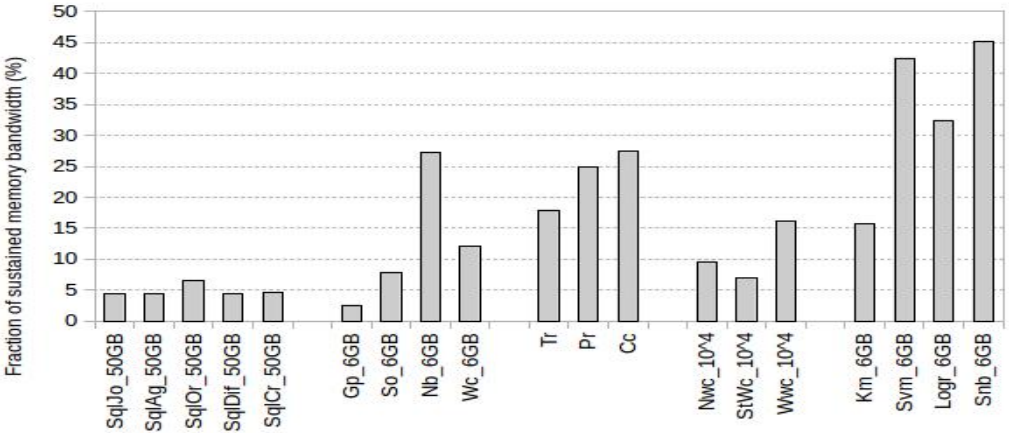


Figure 4: Bandwidth-latency curves of DDR3 and HMC systems



SPARK SUMMIT EUROPE 2017

M. Radulovic et al. Another Trip to the Wall: How Much Will Stacked DRAM Benefit HPC?

A refined hypothesis based on workload characterization ?

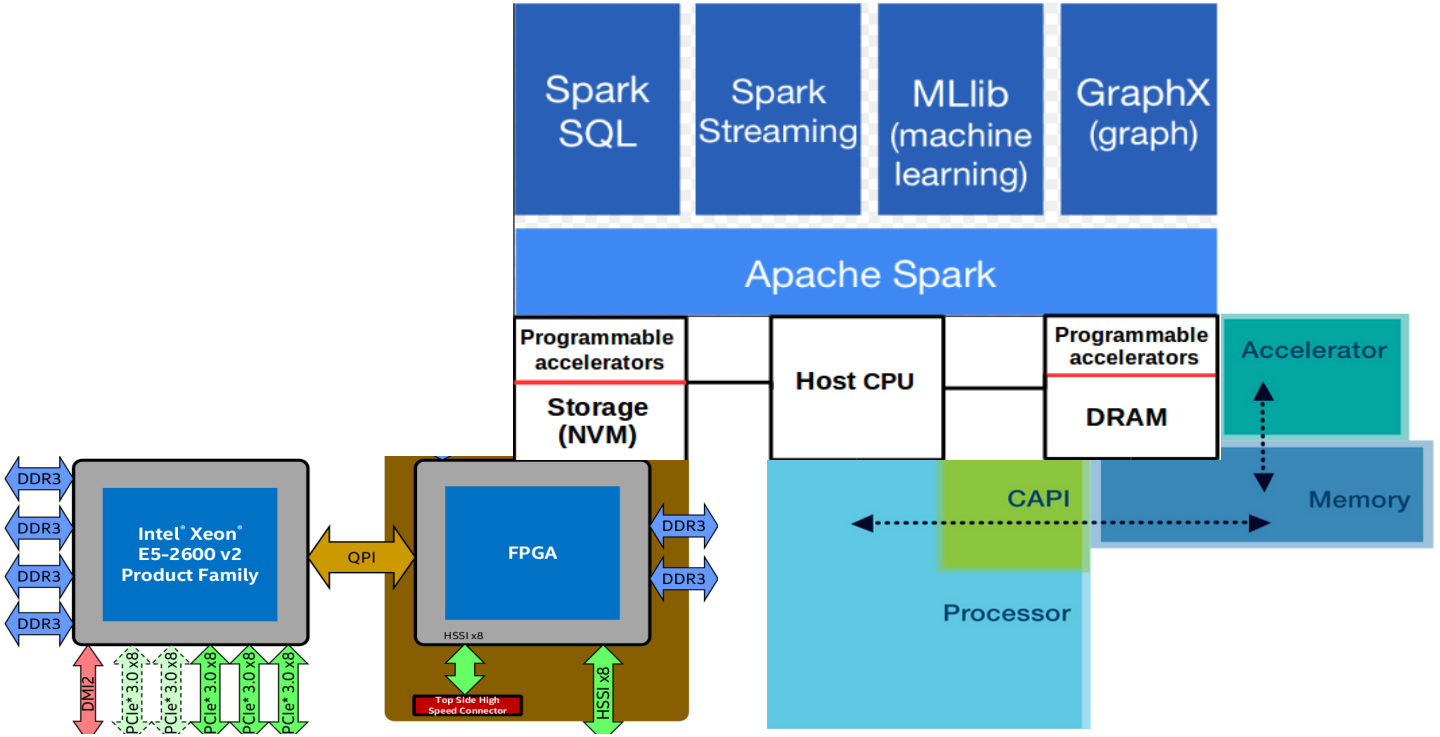
- Spark workloads, which are ***not iterative*** and have ***high ratio of I/O wait time / CPU time*** like join, aggregation, filter, word count and sort are ideal candidates for ***ISP*****.**
- Spark workloads, which have ***low ratio of I/O wait time / CPU time*** like stream processing and iterative graph processing workloads are bound by ***latency*** of frequent accesses to ***DRAM*** and are ideal candidates for ***2D integrated PIM***.
- Spark workloads, which are ***iterative*** and have ***moderate ratio of I/O wait time / CPU time*** like K-means, have both ***I/O bound*** *and* ***memory bound*** phases and hence will benefit from ***hybrid*** 2D integrated PIM and ISP.
- In order to satisfy the varying compute demands of Spark workloads, we envision an NDC architecture with ***programmable logic*** based hybrid ISP and 2D integrated PIM.

How to test the refined hypothesis ?

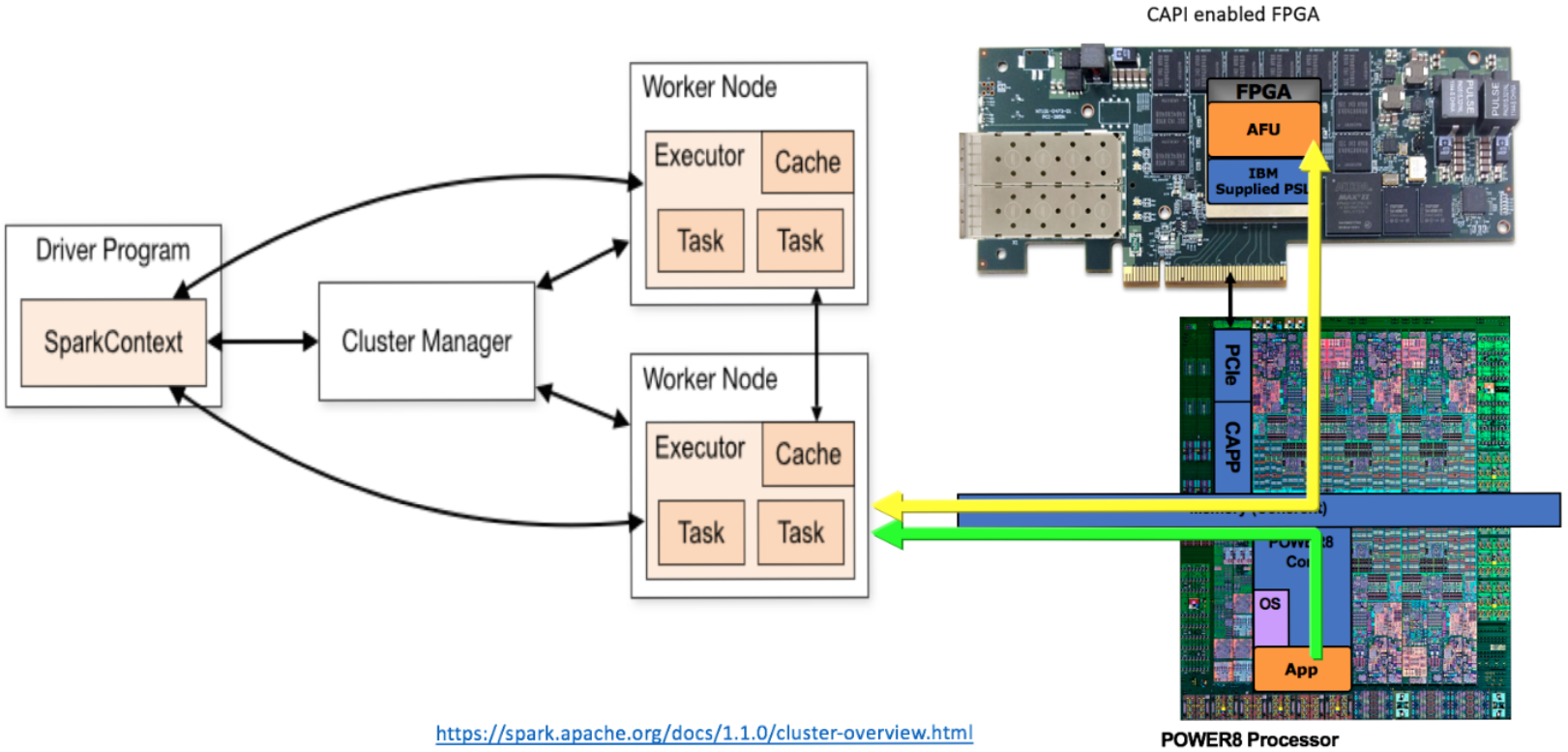
- Simulation Approach
  - Very slow for big data applications :(
- Modeling Approach
  - Overly estimated numbers :(
- Emulation Approach
  - A lot of development :(

How about a combination of Modeling and partial Emulation ?

Can existing tightly coupled servers be used as emulators ?

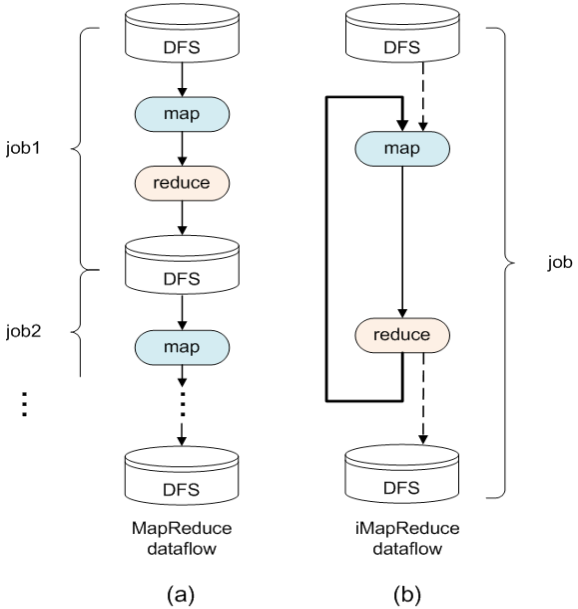


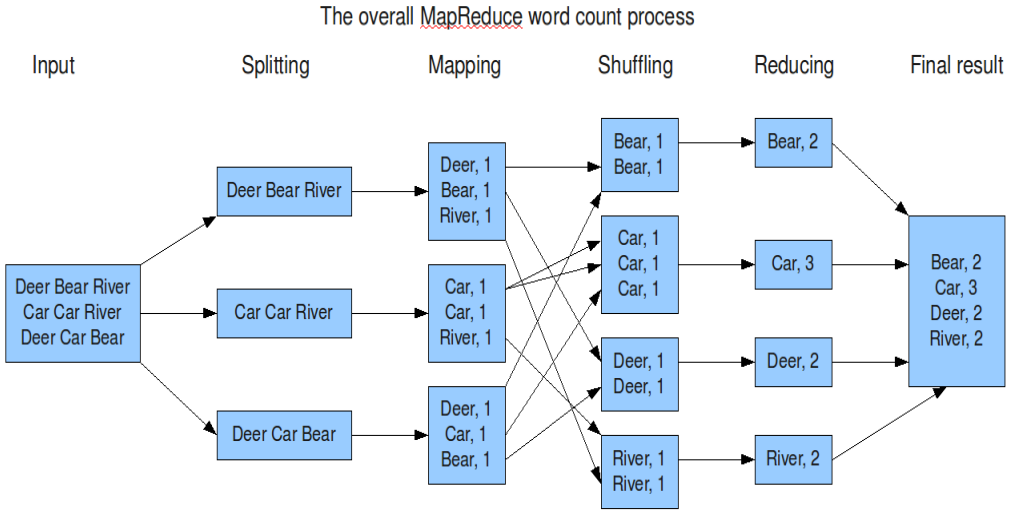
Our System Design ?



https://spark.apache.org/docs/1.1.0/cluster-overview.html

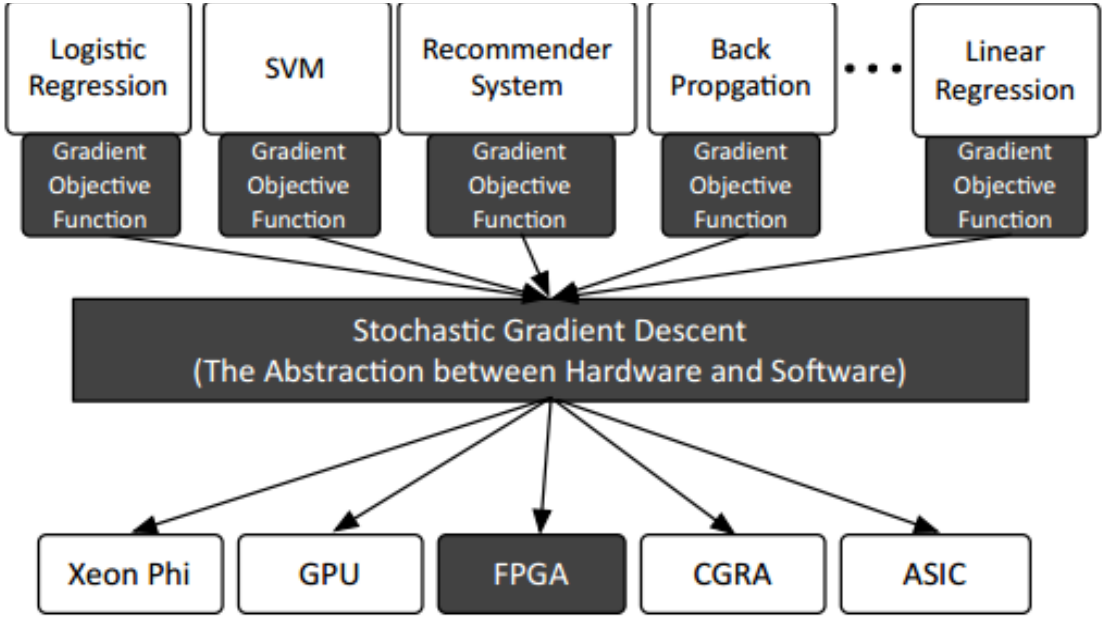
Which programming model ?





Iterative MapReduce

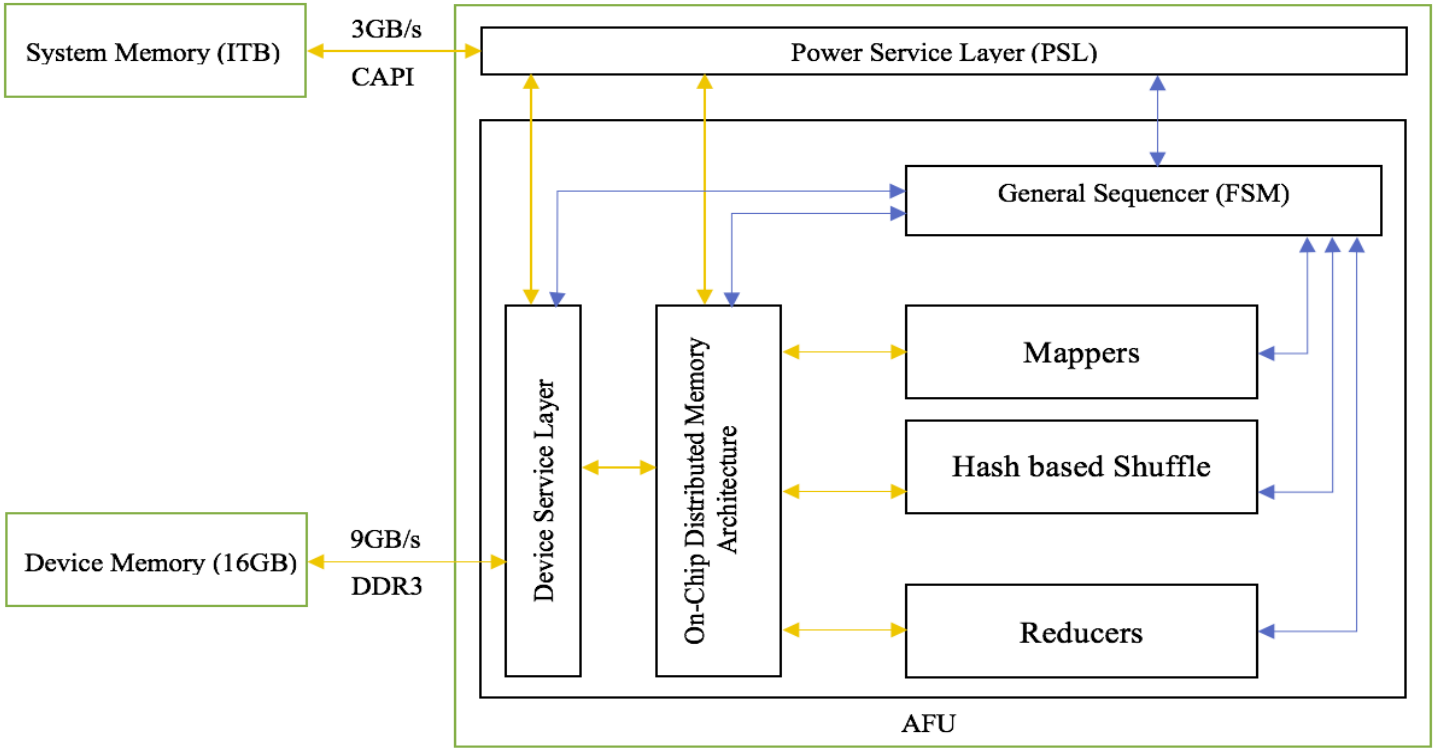
Which workloads ?



K-means and SGD

Mahan et al. TABLA: A unified template-based framework for accelerating statistical machine learning

Our programmable accelerators ?



Advantages of the design ?

- Template based design to support generality.
- No of mappers and reducers can be instantiated based on the FPGA card.
- General Sequencer is a Finite State Machine whose states can be varied to meet the diverse set of workloads
- Mappers and Reducers can be programmed in C/C++ and can be synthesized using High Level Synthesis.
- Support hardware acceleration of Diverse set of workloads

How about using a roof-line model ?

Table 1: Spark MLlib Workloads

Spark Workloads	Time Complexity per iteration
K-Means	O(k*n*d)
Linear Regression	O(n*d2 + d3)
Gradient Descent	O(n*d)
SVM using SMO (libsvm)	O(d*n3)
Decision Tree Training	O(n*d*log(n))
Least Squares using SVD	O(n* d2)
Ridge Regression	O(n*d2)
Least angle regression	O(n*d2)
Alternating Least Squaes	O(k2 + n*k3)
Cholesky Factorization	O(n3)
Multi Layer Perceptron	O(n*m*d)
Stochastic Gradient Descent	O(n*d + k*n)

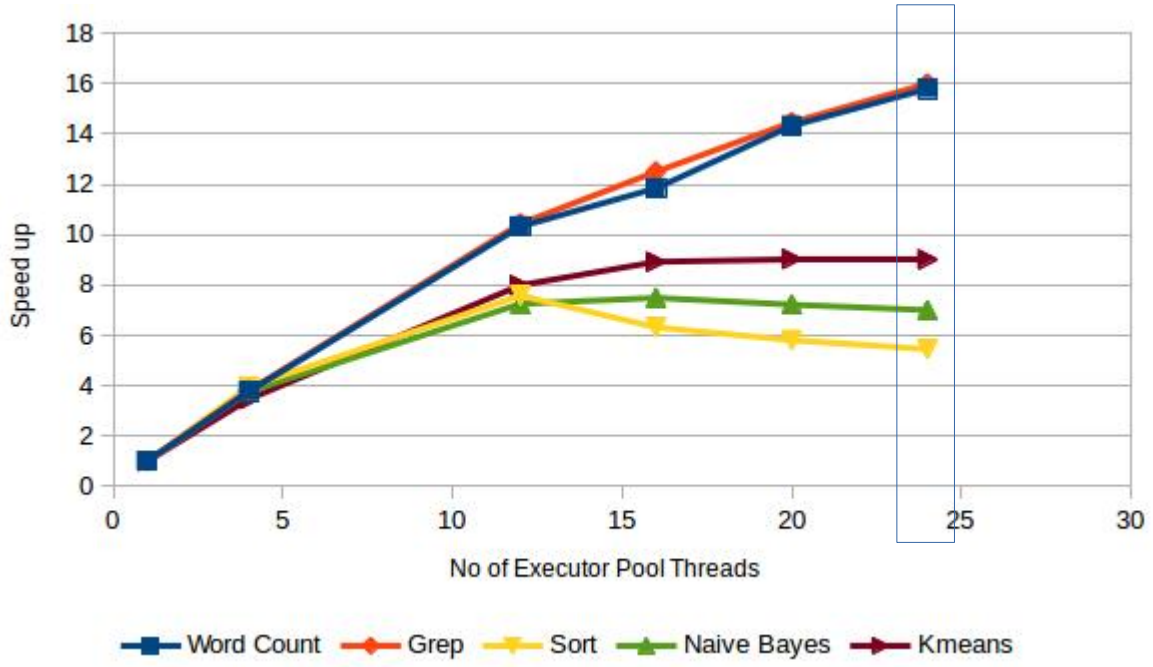
Used to estimate Arithmetic Intensities

Table 2: Machine Details

	E870
Sockets	8
Chips	8
Cores	64
Centaur chips	64
Core-clock rate (GHz)	4.35
Memory (TB)	2
Peak Performace (GFlops/s)	2227
Peak Bandwidth (GB/s)	1843

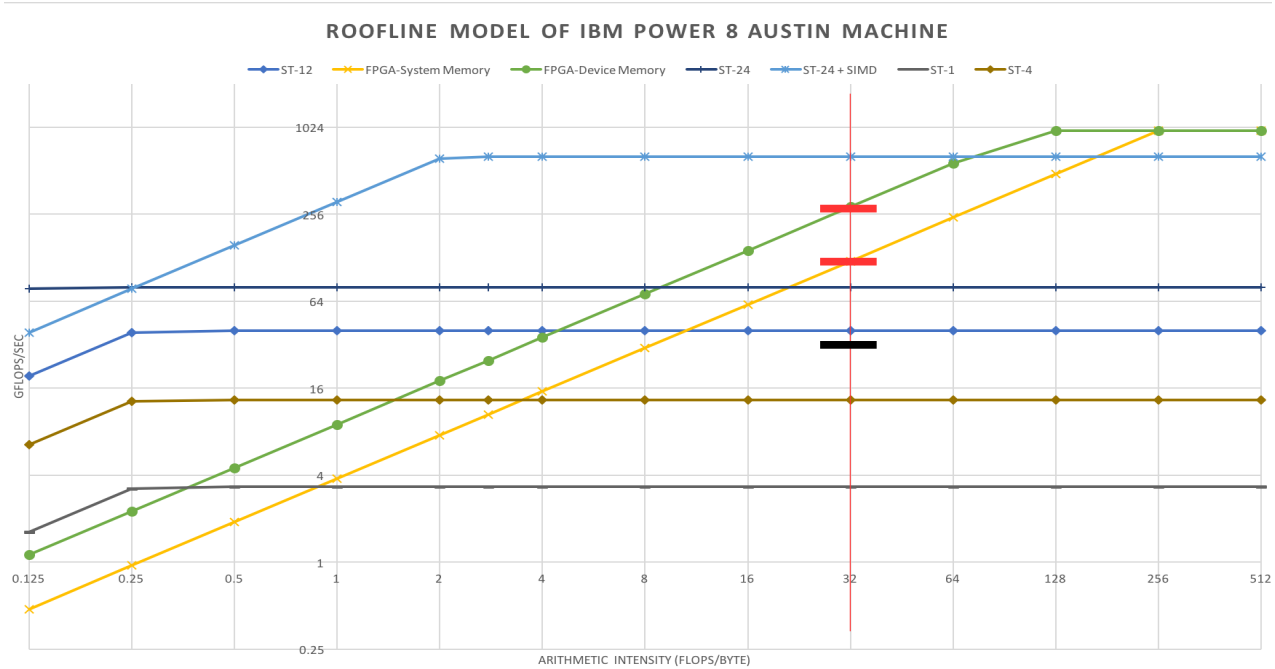
Used to generate roof-line model

Let's show some numbers ?



Poor multi-core scalability of Apache Spark

What are the opportunities ?

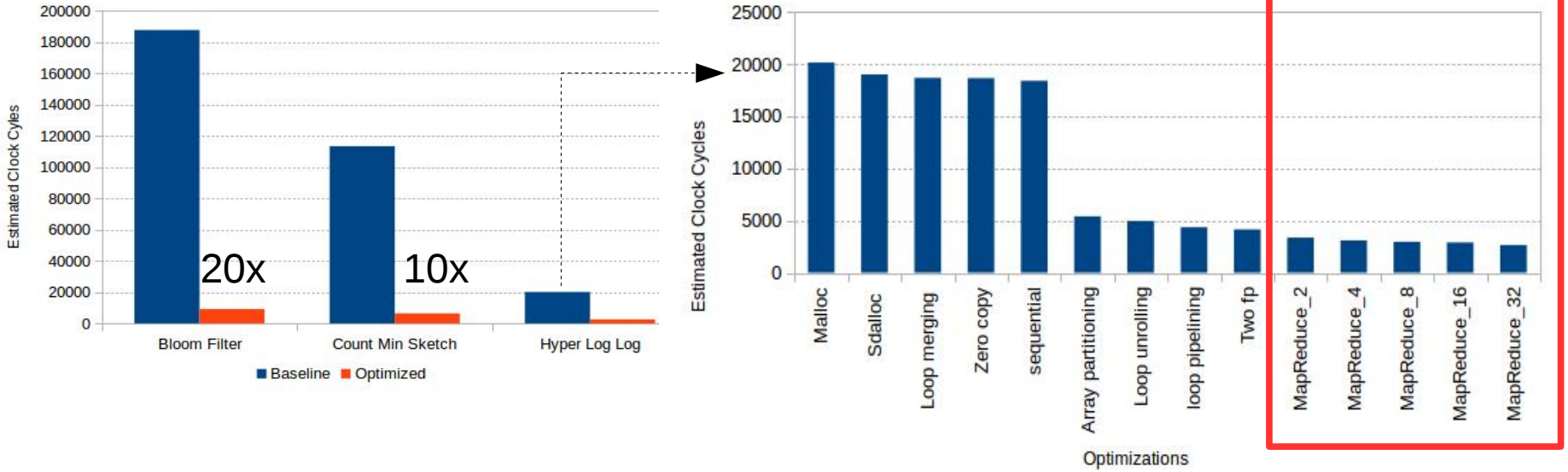


K-means (Km)

What are the challenges ?

- How to design the best hybrid CPU + FPGA ML workloads ?
- How to attain peak performance on CPU side ?
- How to attain peak performance on FPGA side ?
- How to balance load between CPU and FPGA ?
- How hide communication between JVM and FPGA ?
- How to attain peak CAPI bandwidth consumption ?
- How to design the clever ML workload accelerators using HLS tools ?

What High Level Synthesis (Xilinx SDSoC Tool Chain) can do ?



Things to remember from this talk ?

- 3D Stacked Memories are the over-kill for Apache Spark Workloads.
- Project Night-King aims at improving the single node performance of Apache Spark using programmable accelerators near DRAM and NVRAM.
- Conservatively, Near-data accelerators augmented Scale-up Servers can improve the performance of Spark MLlib by 5x.
- Never Trust the 20x speed-up claims being made in the industry. Most of the time, the reference points are wrong!
- Xilinx SDSoC Tool Chain needs to support pragmas for map-reduce programming model.

- Performance Characterization and Optimization of In-Memory Data Analytics on a Scale-up Server, PhD thesis, Ahsan Javed Awan (ISBN: 978-91-7729-584-6)

That's all for now ?

THANK YOU

Co-funded by the
Erasmus+ Programme
of the European Union

Email: ajawan@kth.se
Profile: www.kth.se/profile/ajawan/
https://se.linkedin.com/in/ahsanjavedawan