
CERN

Trident

Automated system tool for collecting and analyzing hardware performance counters

Servesh Muralidharan & David Smith

IT-DI-WLCG, UP Team

CERN

Dec 2018

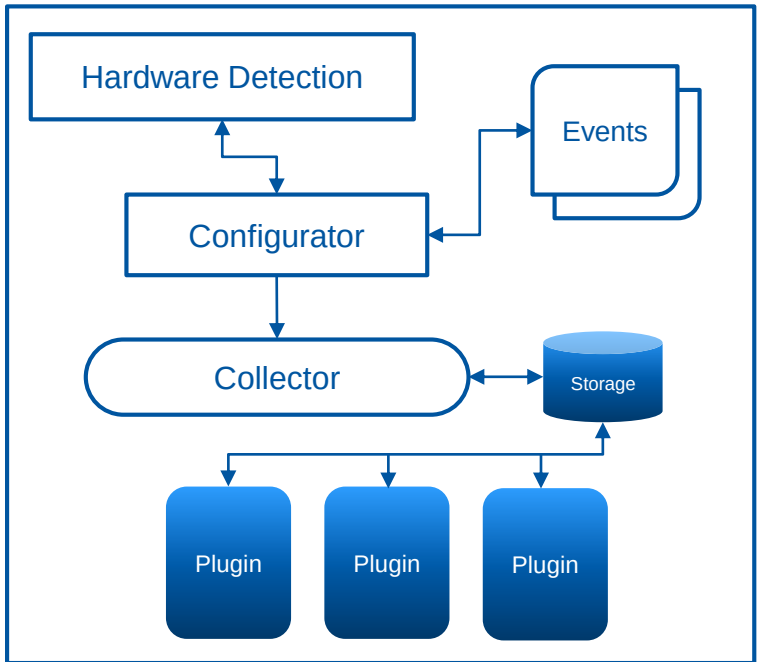
Introduction

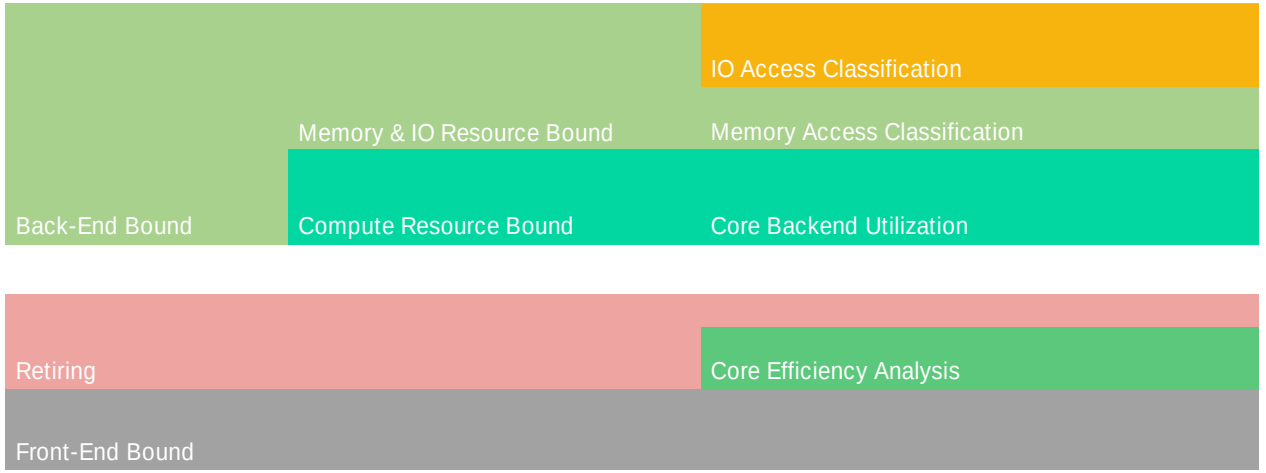
- Existing tools
  - Execution and profiling are separate steps
  - Focus more on HPC Applications
    - Fine grained analysis
    - Architecture specific optimizations
- Typical HEP Applications (As it is today!!!)
  - A cluster of several hundred algorithms
  - Complex framework interconnecting these algorithms
  - Linear instruction spread (**no hotspots**)
  - Non existence of classical numerical loops
  - Average runtime of several hours

- Proposed Solution: Trident
  - Continuous performance analysis
  - System perspective
  - Node level measurements
  - Non-virtualized environment
  - Low overhead and lightweight
  - Utilizes hardware counters to measure Core, Memory and IO metrics
  - Presents it in a HEP-human readable form

Trident Architecture

- Detects the events supported by the node
- Extensible event files for future architectures
- Configures metrics based on these events
- These metrics are sampled at preconfigured rate
- Stored temporarily in local disk
- Plugins perform transformations to the data
- Simple low overhead online analysis
- Data exported to centralized storage for offline analysis



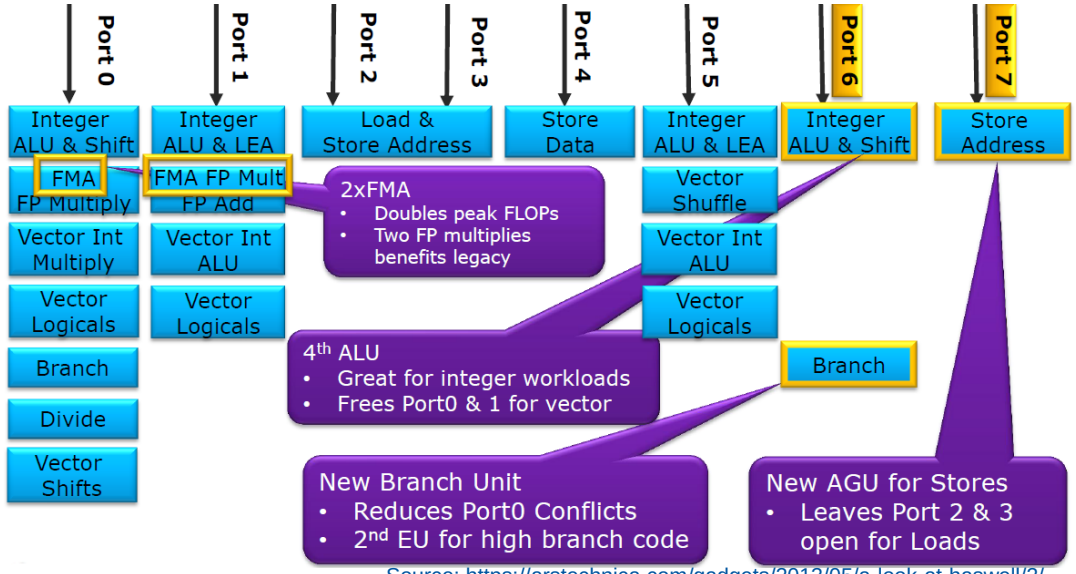


Extended Top Down Analysis

Trident - Core Performance Analysis

- Instruction Per Cycle (IPC)
  - Denotes ratio of parallel instructions executed
  - Modern processors like Intel Haswell EP can do 4 IPC
- Top down characterization
  - Identifies the resources dominated by workload
  - Front-End – fetch and decode program code
  - Back-End – monitor and execution of uOP once the dependent data operands availability
  - Retiring – Completion of the uOP
  - Bad speculation – uOPs that are cancelled before retirement due to branch misprediction
- Execution Unit Port Utilization
  - Determines how many cycles the port was busy
  - Identifies broadly the pressure from different types of uOPs
  - INT & FP operations, address calculation, etc.,

Intel Haswell EP



Source: https://arstechnica.com/gadgets/2013/05/a-look-at-haswell/2/

	Expected Range of Pipeline Slots in This Category, for a Hotspot in a Well-Tuned:		
Category	Client/Desktop Application	Server/Database/Distributed application	High Performance Computing (HPC) application
Retiring	20-50%	10-30%	30-70%
Back-End Bound	20-40%	20-60%	20-40%
Front-End Bound	5-10%	10-25%	5-10%
Bad Speculation	5-10%	5-10%	1-5%

Source: https://software.intel.com/en-us/vtune-amplifier-help-tuning-applications-using-a-top-down-microarchitecture-analysis-method

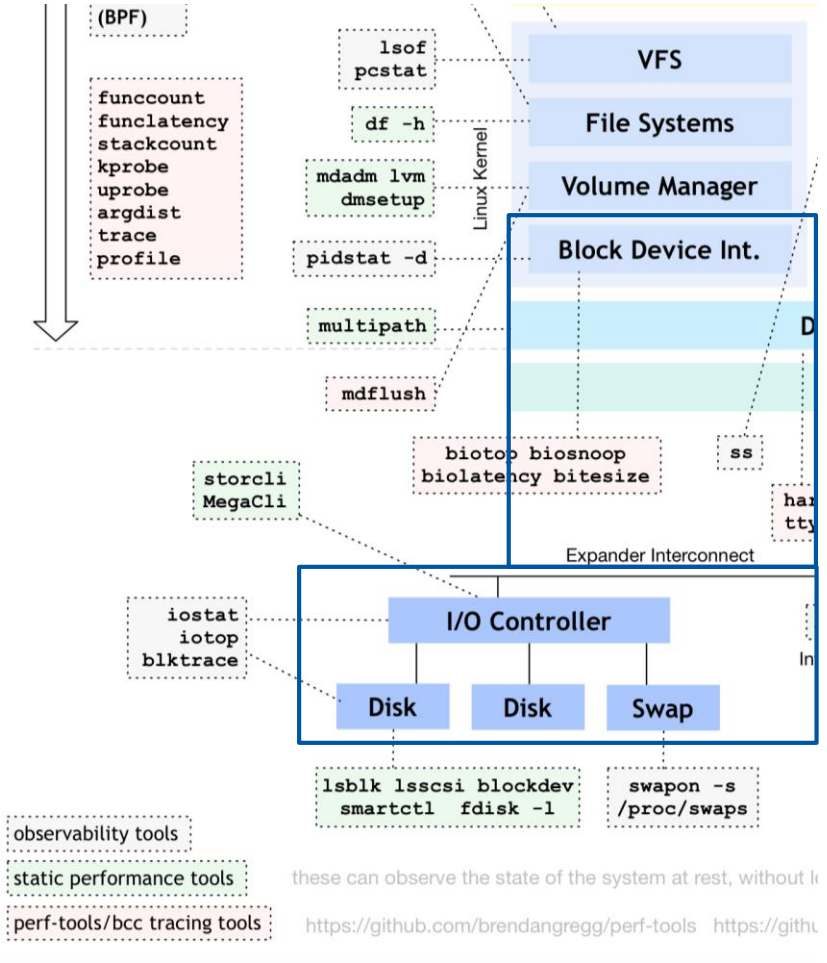
Trident - Memory Performance Analysis

- Memory bandwidth usage
  - Amount of data read and written to main memory
- Memory transaction classification
  - Memory transaction occurs through pages from memory bank
  - Page-Hit
    - Memory bank in open state
    - Lowest access latency (Open)
    - Sequential memory access usually have high page hits
  - Page-Empty
    - Memory bank is idle and needs to be activated
    - Moderate access latency (Usually 2x of page hit)
    - Random memory access usually have high page empty counts
  - Page-Miss
    - Memory to be accessed requires closing of a page in the same bank
    - Worst access latency (Usually 3x of page hit)

Further info: https://www.anandtech.com/show/3851/everything-you-always-wanted-to-know-about-sdram-memory-but-were-afraid-to-ask/5

Trident - IO Performance Analysis

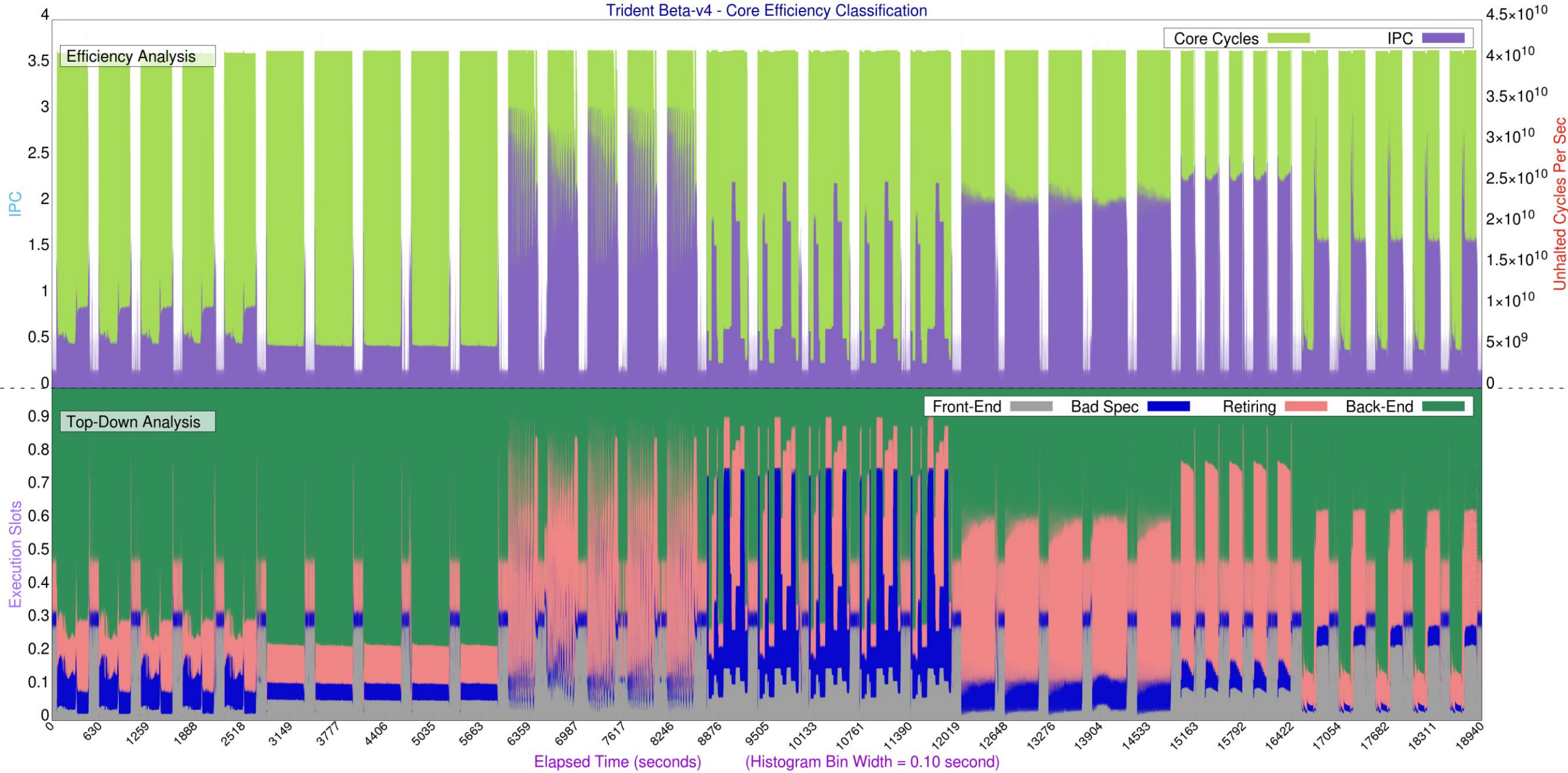
- Data recorded form ProcFS
- Transfer Rate Analysis
  - Amount of data read and written to storage
  - Limited by interface and type of memory
  - Sustained Vs Bursts
- Operation Rate Analysis
  - Amount of operations performed
  - Limited by controller for fast memory
  - Limited by disk head for HDDs
  - Random Vs Sequential accesses



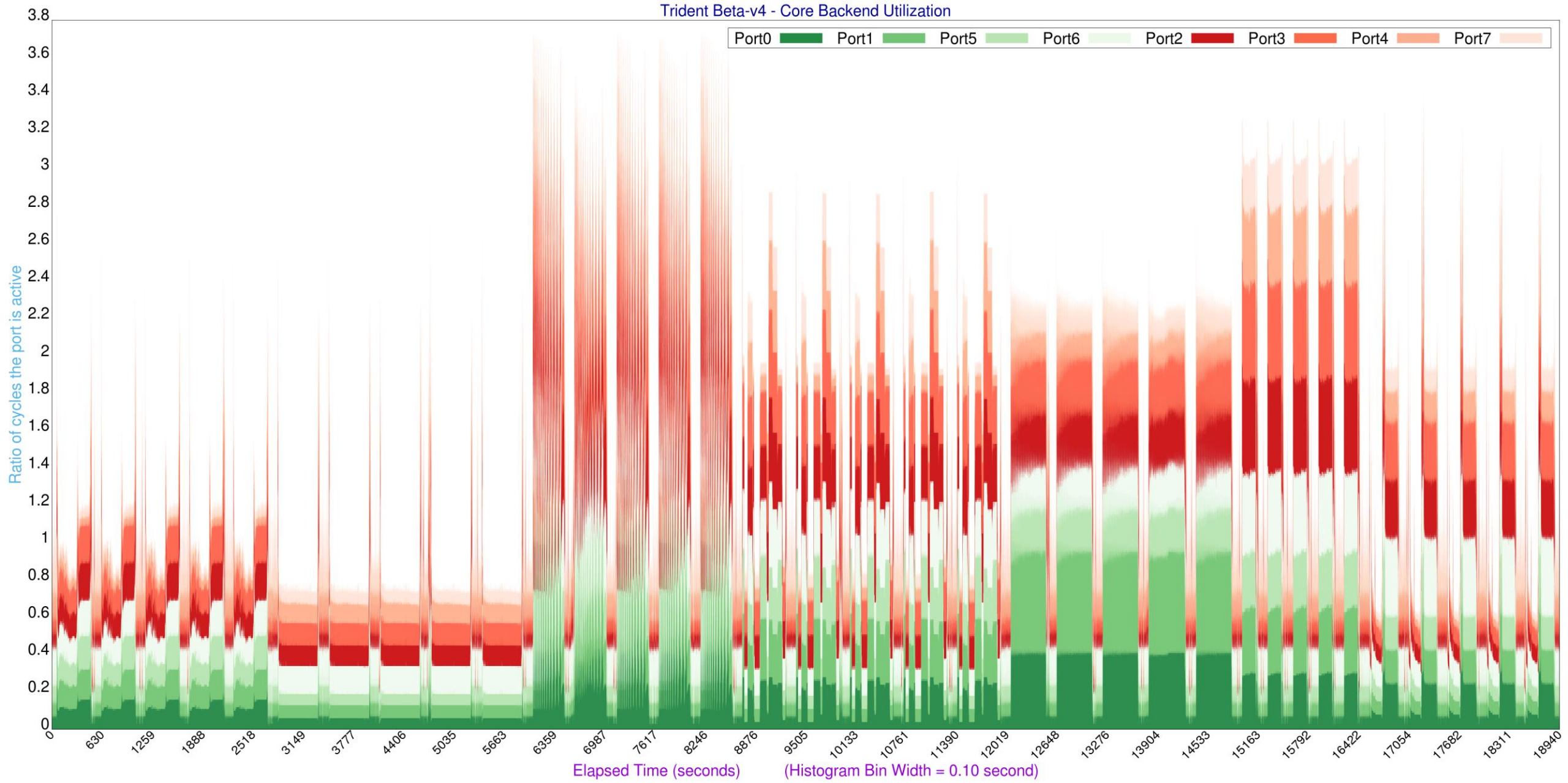
Experiment setup

- Test System : 2x Intel(R) Xeon(R) E5-2630 v3@2.40GHz / 64GB DDR4-2133 RAM / Centos 7 (3.10.0-862.el7.x86_64)
- Workloads
  - HEPSPEC06 Benchmark Suite
    - 450.soplex,471.omnetpp,447.dealII,473.astar,444.namd,453.povray,483.xalancbmk
  - ATLAS Job 1 - Geant4 MC simulation (CPU Intensive)
    - Executes: Sim_tf (Geant4)
    - Input: EVNT files from event generation / Output: HITS files
  - ATLAS Job 2 - MC digitization and reco (CPU+I/O intensive)
    - Executes: Reco_tf with several sub-steps
      - HITtoRDO (Digitization)
      - RDOtoRDOTrigger (Trigger simulation)
      - RAWtoESD + ESDtoAOD (MC reconstruction)
      - POOLMergeAthenaMPAOD0 (merge some small outputs)
    - Input: HITS files from MC simulation + low and high pile-up HITS for digitization / Output: AOD files
  - ATLAS Job 3 - Derivation production (I/O Intensive)
    - Executes: Reco_tf in AODtoDAOD mode
      - AODtoDAOD
      - DAODs are used as the basis of ~all physics analysis in ATLAS
      - DAODs produced from AOD by removing whole events (skimming), whole reconstructed objects (thinning) and variables (slimming)
    - Input: AOD / Output: multiple DAODs in a train

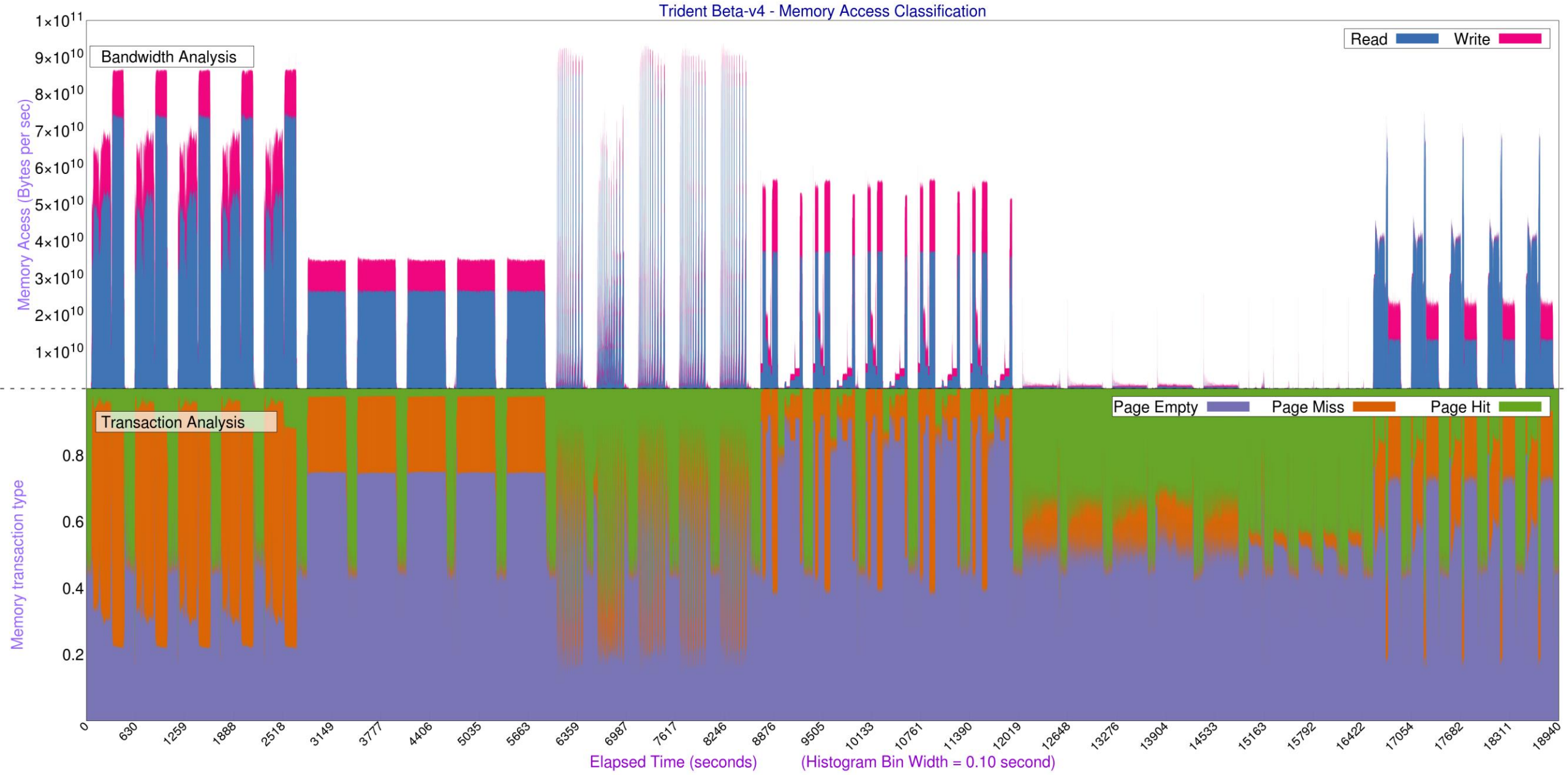
HS06


Trident Beta-v4 - Core Efficiency Classification
Efficiency Analysis
Core Cycles
IPC
IPC
Unhalted Cycles Per Sec
Top-Down Analysis
Front-End
Bad Spec
Retiring
Back-End
Execution Slots
Elapsed Time (seconds)
(Histogram Bin Width = 0.10 second)

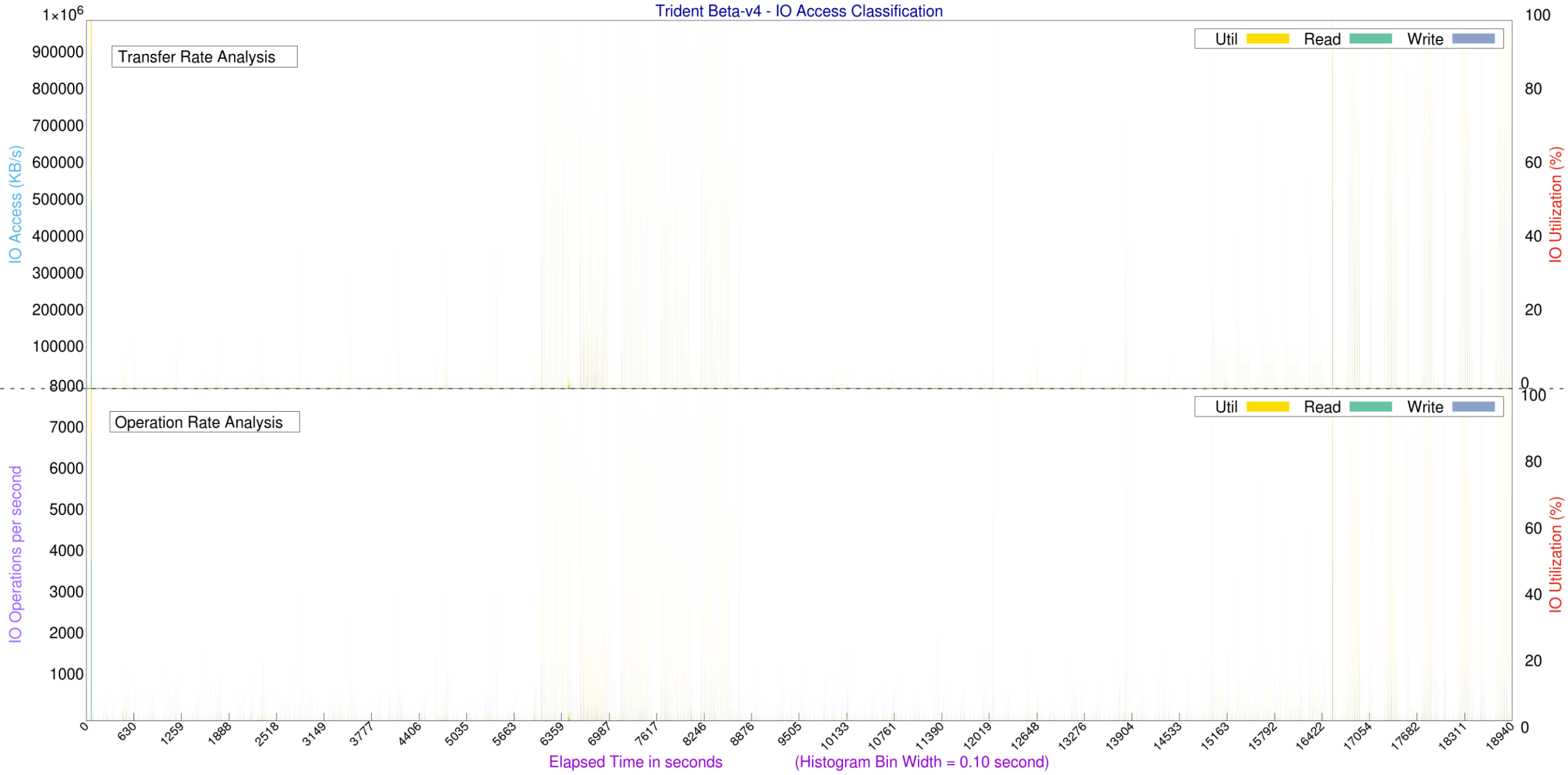
HS06


Trident Beta-v4 - Core Backend Utilization
Port0
Port1
Port5
Port6
Port2
Port3
Port4
Port7
Ratio of cycles the port is active
3.8
3.6
3.4
3.2
3
2.8
2.6
2.4
2.2
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
0
630
1259
1888
2518
3149
3777
4406
5035
5663
6359
6987
7617
8246
8876
9505
10133
10761
11390
12019
12648
13276
13904
14533
15163
15792
16422
17054
17682
18311
18940
Elapsed Time (seconds)
(Histogram Bin Width = 0.10 second)

HS06


Trident Beta-v4 - Memory Access Classification
Bandwidth Analysis
Read
Write
Memory Acess (Bytes per sec)
1×10^11
9×10^10
8×10^10
7×10^10
6×10^10
5×10^10
4×10^10
3×10^10
2×10^10
1×10^10
Transaction Analysis
Page Empty
Page Miss
Page Hit
Memory transaction type
0.8
0.6
0.4
0.2
0
630
1259
1888
2518
3149
3777
4406
5035
5663
6359
6987
7617
8246
8876
9505
10133
10761
11390
12019
12648
13276
13904
14533
15163
15792
16422
17054
17682
18311
18940
Elapsed Time (seconds)
(Histogram Bin Width = 0.10 second)

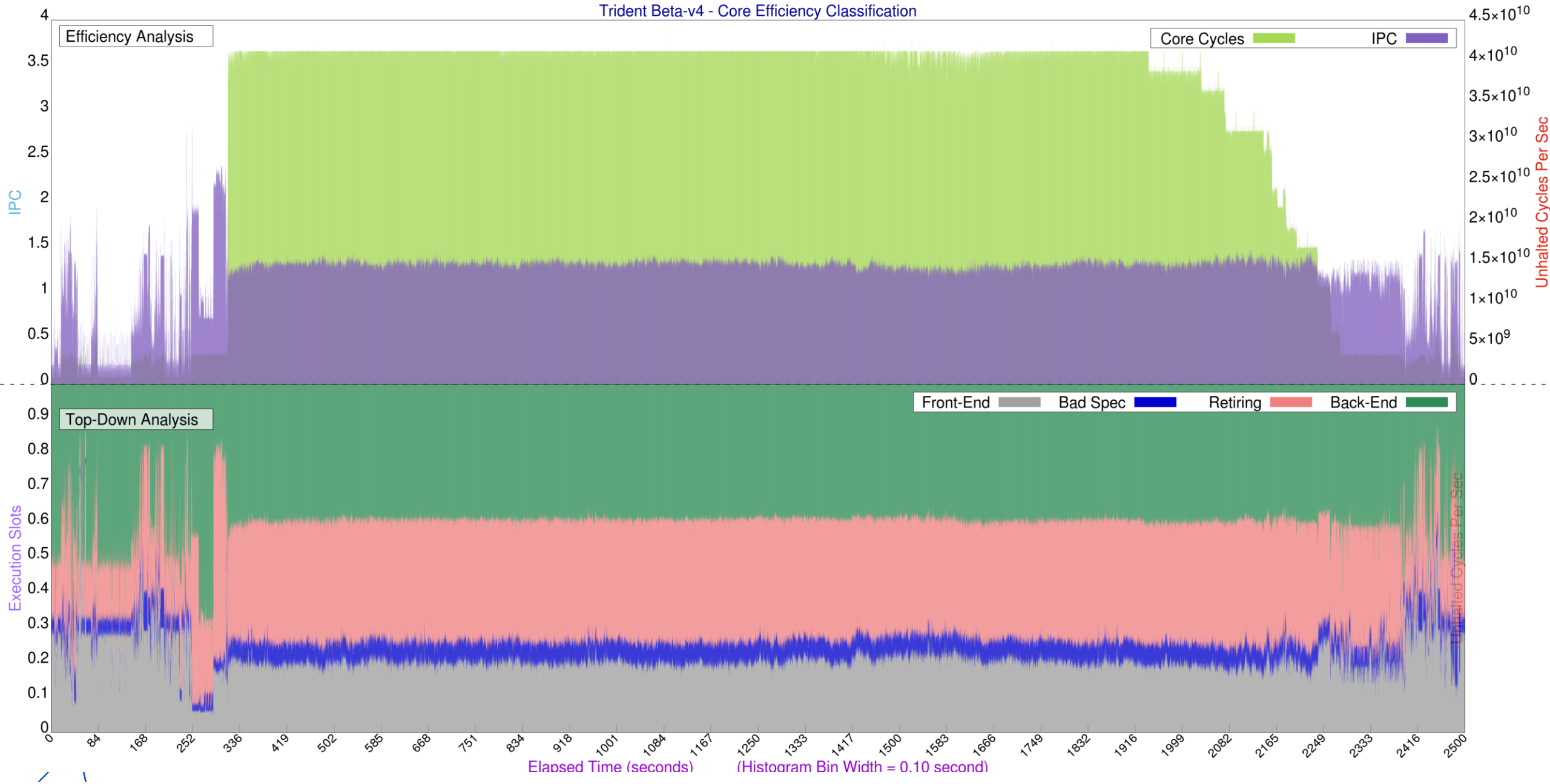
HS06


Trident Beta-v4 - IO Access Classification
Transfer Rate Analysis
Util
Read
Write
IO Access (KB/s)
IO Utilization (%)
1×10^6
900000
800000
700000
600000
500000
400000
300000
200000
100000
8000
100
80
60
40
20
0
Operation Rate Analysis
IO Operations per second
7000
6000
5000
4000
3000
2000
1000
0
630
1259
1888
2518
3149
3777
4406
5035
5663
6359
6987
7617
8246
8876
9505
10133
10761
11390
12019
12648
13276
13904
14533
15163
15792
16422
17054
17682
18311
18940
Elapsed Time in seconds
(Histogram Bin Width = 0.10 second)

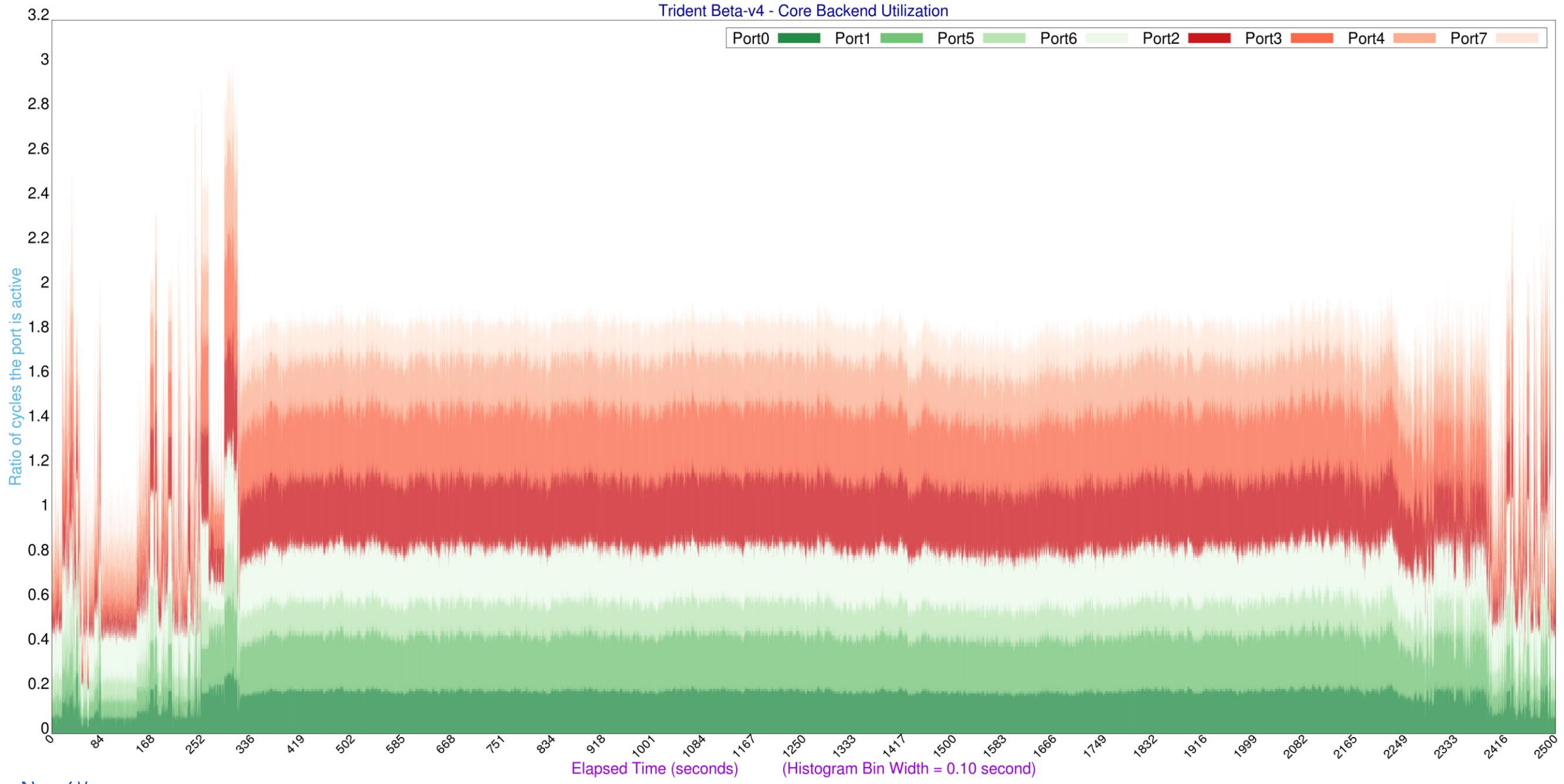
Experiment setup

- Test System : 2x Intel(R) Xeon(R) E5-2630 v3@2.40GHz / 64GB DDR4-2133 RAM / Centos 7 (3.10.0-862.el7.x86_64)
- Workloads
  - HEPSPEC06 Docker Container – Cloud Benchmark Suite (Thanks Domenico Giordano!!!)
    - 450.soplex,471.omnetpp,447.dealII,473.astar,444.namd,453.povray,483.xalancbmk
  - ATLAS Job 1 - Geant4 MC simulation (CPU Intensive)
    - Executes: Sim_tf (Geant4)
    - Input: EVNT files from event generation / Output: HITS files
  - ATLAS Job 2 - MC digitization and reco (CPU+I/O intensive)
    - Executes: Reco_tf with several sub-steps
      - HITtoRDO (Digitization)
      - RDOtoRDOTrigger (Trigger simulation)
      - RAWtoESD + ESDtoAOD (MC reconstruction)
      - POOLMergeAthenaMPAOD0 (merge some small outputs)
    - Input: HITS files from MC simulation + low and high pile-up HITS for digitization / Output: AOD files
  - ATLAS Job 3 - Derivation production (I/O Intensive)
    - Executes: Reco_tf in AODtoDAOD mode
      - AODtoDAOD
      - DAODs are used as the basis of ~all physics analysis in ATLAS
      - DAODs produced from AOD by removing whole events (skimming), whole reconstructed objects (thinning) and variables (slimming)
    - Input: AOD / Output: multiple DAODs in a train

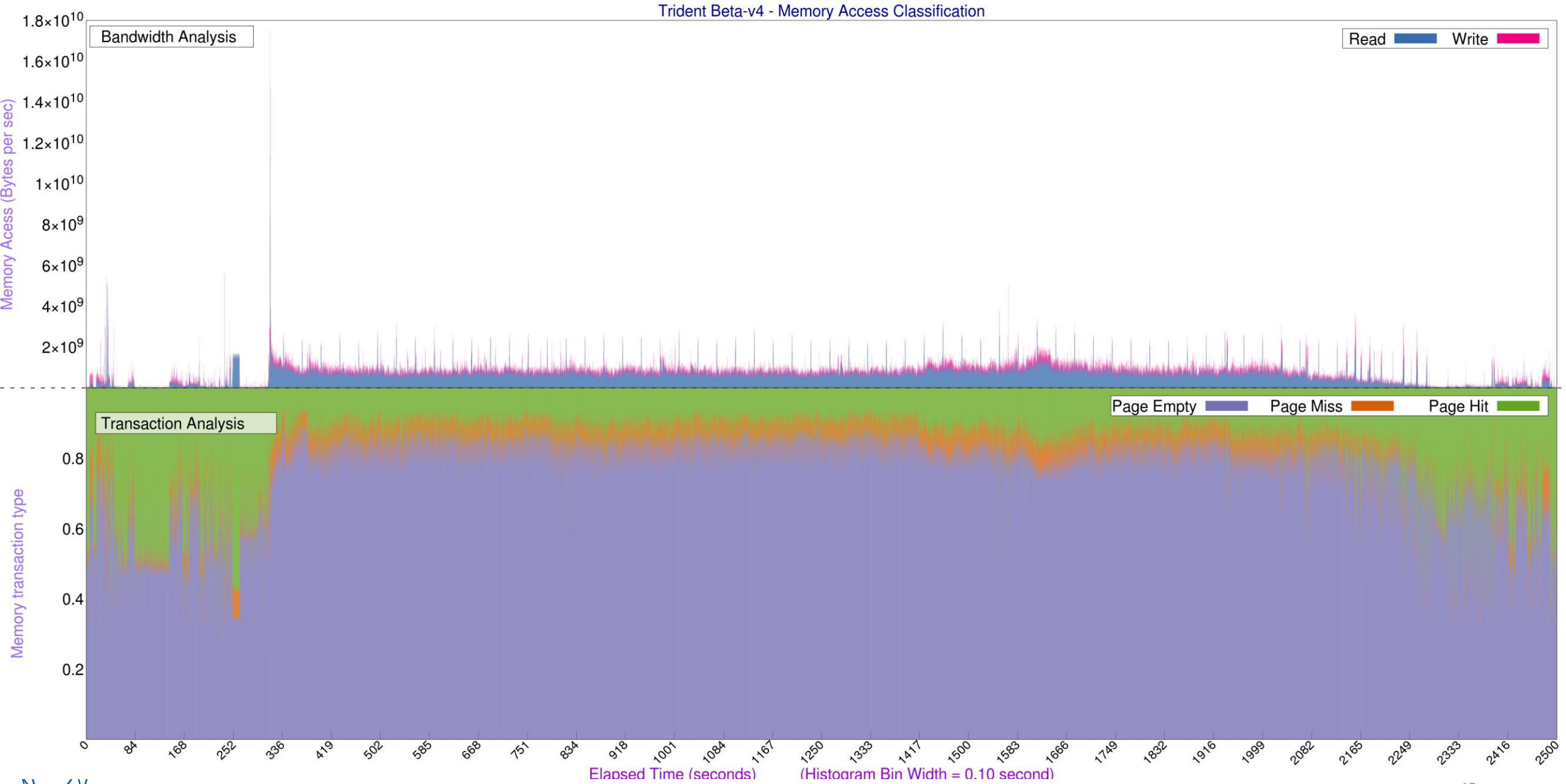
Geant4


Trident Beta-v4 - Core Efficiency Classification
Efficiency Analysis
Core Cycles
IPC
IPC
Unhalted Cycles Per Sec
Top-Down Analysis
Front-End
Bad Spec
Retiring
Back-End
Execution Slots
Unhalted Cycles Per Sec
Elapsed Time (seconds)
(Histogram Bin Width = 0.10 second)

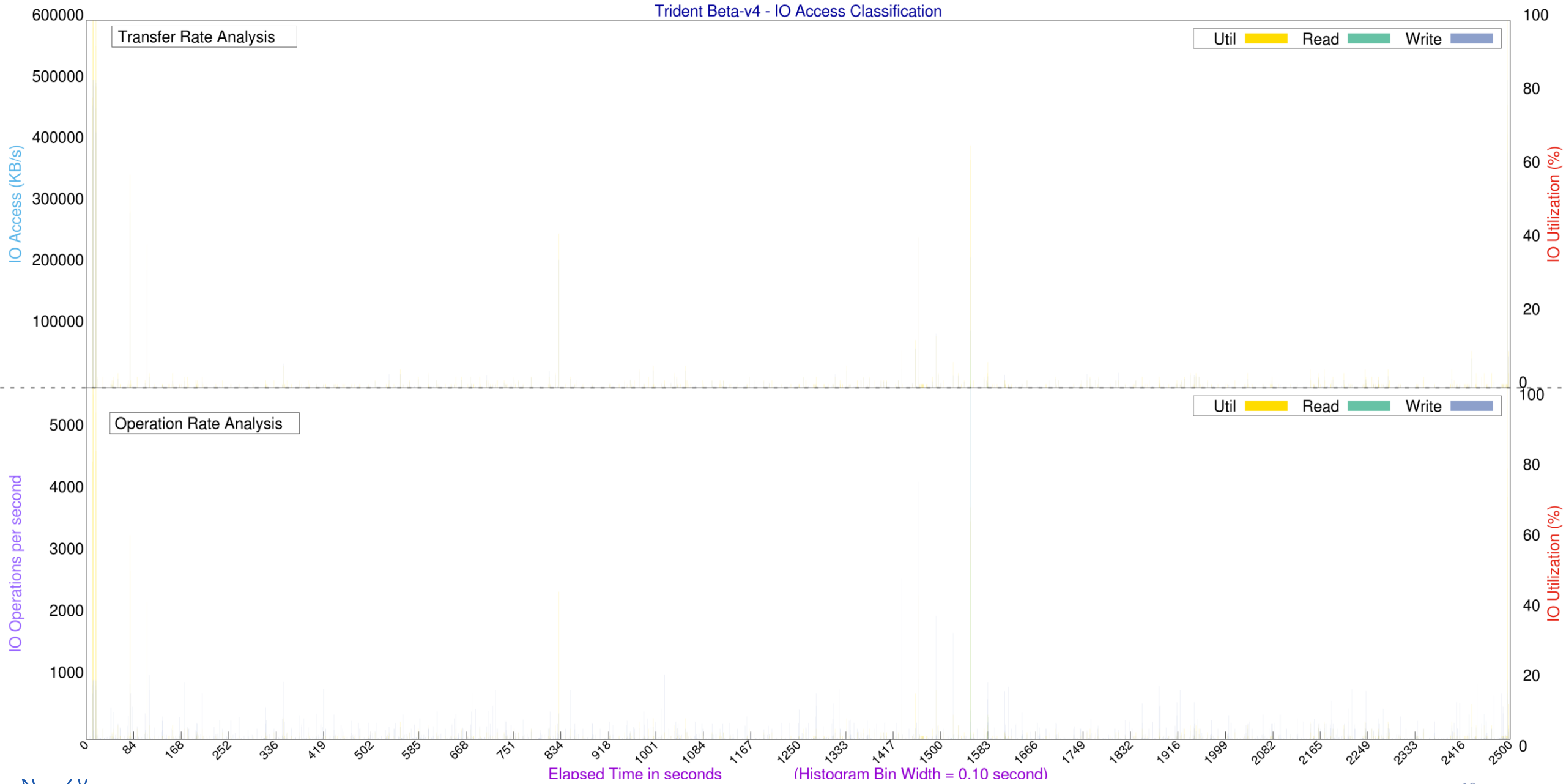
Geant4


Trident Beta-v4 - Core Backend Utilization
Port0
Port1
Port5
Port6
Port2
Port3
Port4
Port7
Ratio of cycles the port is active
3.2
3
2.8
2.6
2.4
2.2
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
0
84
168
252
336
419
502
585
668
751
834
918
1001
1084
1167
1250
1333
1417
1500
1583
1666
1749
1832
1916
1999
2082
2165
2249
2333
2416
2500
Elapsed Time (seconds)
(Histogram Bin Width = 0.10 second)

Geant4


Trident Beta-v4 - Memory Access Classification
Bandwidth Analysis
Read
Write
Memory Acess (Bytes per sec)
1.8×10^10
1.6×10^10
1.4×10^10
1.2×10^10
1×10^10
8×10^9
6×10^9
4×10^9
2×10^9
Transaction Analysis
Page Empty
Page Miss
Page Hit
Memory transaction type
0.8
0.6
0.4
0.2
0
84
168
252
336
419
502
585
668
751
834
918
1001
1084
1167
1250
1333
1417
1500
1583
1666
1749
1832
1916
1999
2082
2165
2249
2333
2416
2500
Elapsed Time (seconds)
(Histogram Bin Width = 0.10 second)

Geant4


Trident Beta-v4 - IO Access Classification
Transfer Rate Analysis
Util
Read
Write
IO Access (KB/s)
IO Utilization (%)
600000
500000
400000
300000
200000
100000
100
80
60
40
20
0
Operation Rate Analysis
IO Operations per second
5000
4000
3000
2000
1000
0
0
84
168
252
336
419
502
585
668
751
834
918
1001
1084
1167
1250
1333
1417
1500
1583
1666
1749
1832
1916
1999
2082
2165
2249
2333
2416
2500
Elapsed Time in seconds
(Histogram Bin Width = 0.10 second)

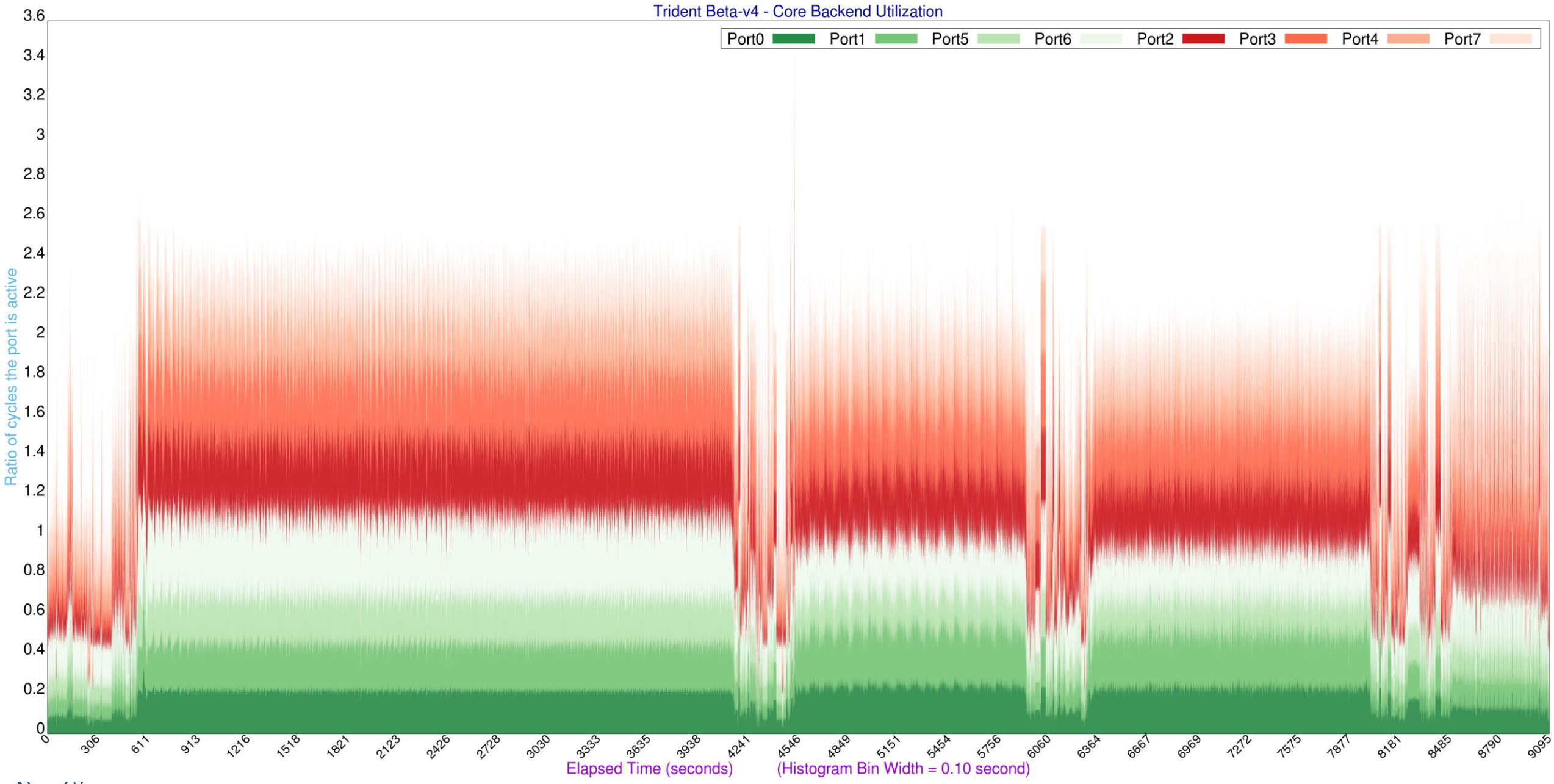
Experiment setup

- Test System : 2x Intel(R) Xeon(R) E5-2630 v3@2.40GHz / 64GB DDR4-2133 RAM / Centos 7 (3.10.0-862.el7.x86_64)
- Workloads
  - HEPSPEC06 Docker Container – Cloud Benchmark Suite (Thanks Domenico Giordano!!!)
    - 450.soplex,471.omnetpp,447.dealII,473.astar,444.namd,453.povray,483.xalancbmk
  - ATLAS Job 1 - Geant4 MC simulation (CPU Intensive)
    - Executes: Sim_tf (Geant4)
    - Input: EVNT files from event generation / Output: HITS files
  - ATLAS Job 2 - MC digitization and reco (CPU+I/O intensive)
    - Executes: Reco_tf with several sub-steps
      - HITtoRDO (Digitization)
      - RDOtoRDOTrigger (Trigger simulation)
      - RAWtoESD + ESDtoAOD (MC reconstruction)
      - POOLMergeAthenaMPAOD0 (merge some small outputs)
    - Input: HITS files from MC simulation + low and high pile-up HITS for digitization / Output: AOD files
  - ATLAS Job 3 - Derivation production (I/O Intensive)
    - Executes: Reco_tf in AODtoDAOD mode
      - AODtoDAOD
      - DAODs are used as the basis of ~all physics analysis in ATLAS
      - DAODs produced from AOD by removing whole events (skimming), whole reconstructed objects (thinning) and variables (slimming)
    - Input: AOD / Output: multiple DAODs in a train

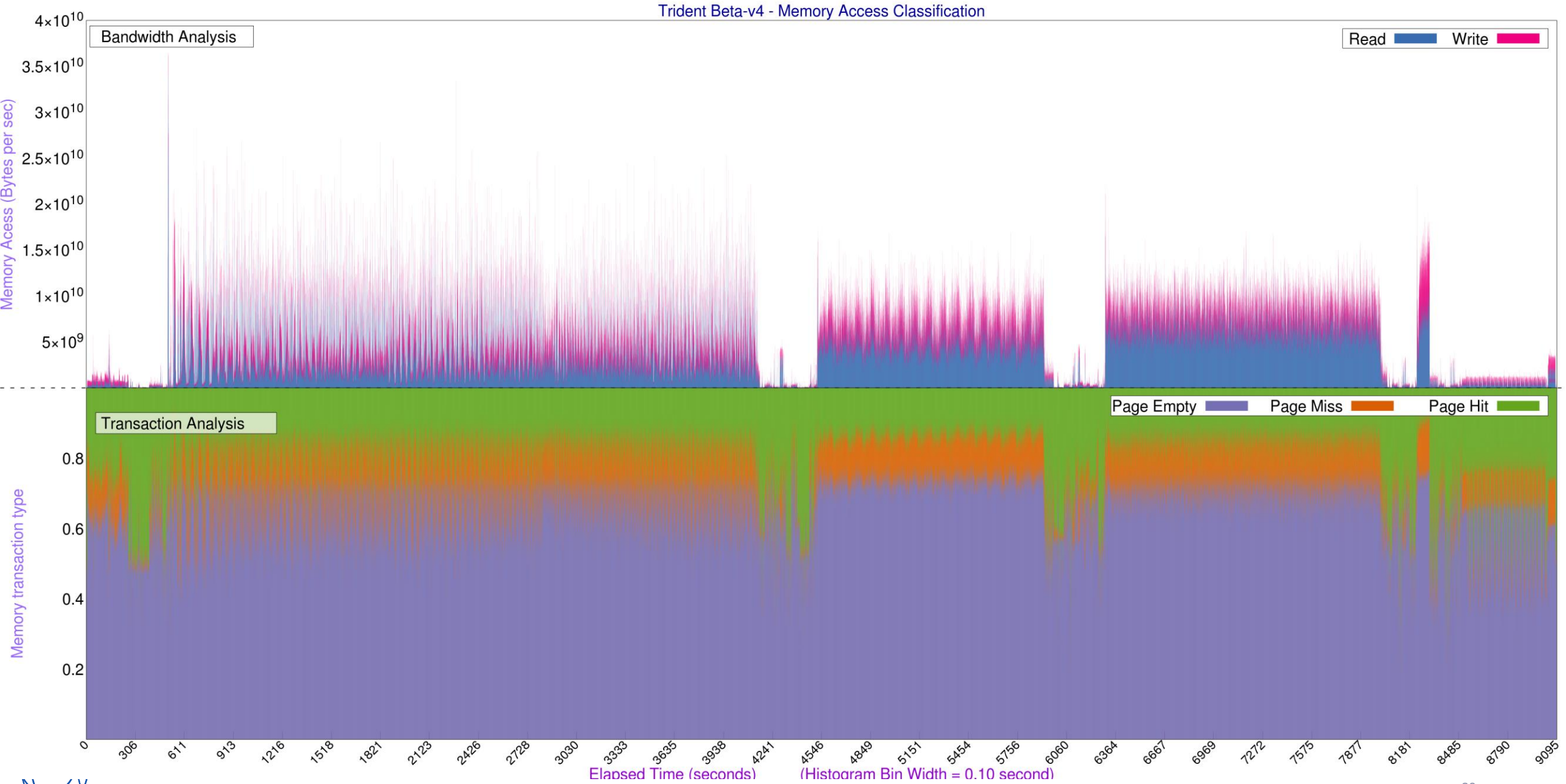
MC Digi + Reco


Trident Beta-v4 - Core Efficiency Classification
Efficiency Analysis
Core Cycles
IPC
IPC
Unhalted Cycles Per Sec
Top-Down Analysis
Front-End
Bad Spec
Retiring
Back-End
Execution Slots
Elapsed Time (seconds)
(Histogram Bin Width = 0.10 second)

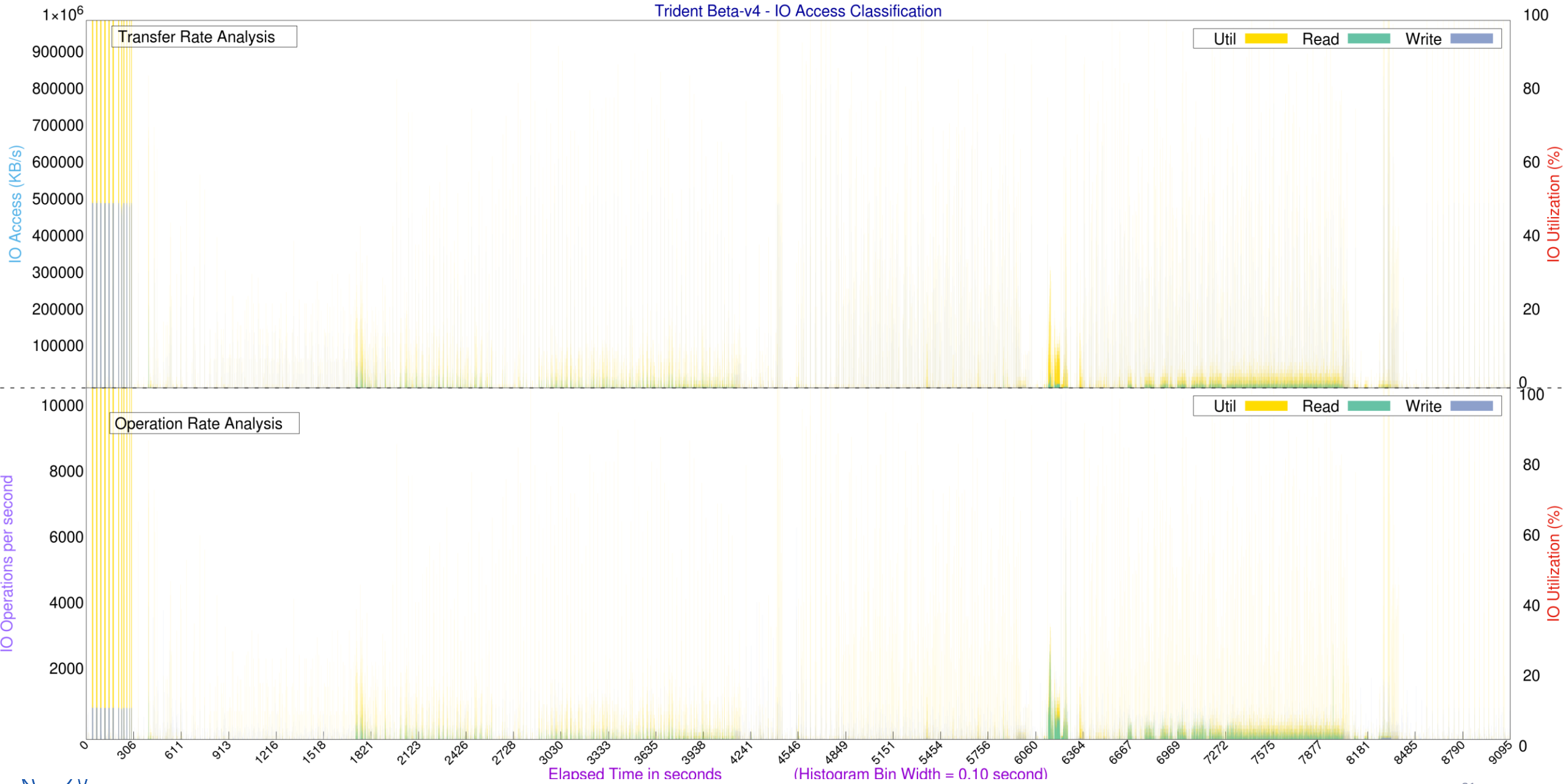
MC Digi + Reco


Trident Beta-v4 - Core Backend Utilization
Port0
Port1
Port5
Port6
Port2
Port3
Port4
Port7
Ratio of cycles the port is active
3.6
3.4
3.2
3
2.8
2.6
2.4
2.2
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
0
306
611
913
1216
1518
1821
2123
2426
2728
3030
3333
3635
3938
4241
4546
4849
5151
5454
5756
6060
6364
6667
6969
7272
7575
7877
8181
8485
8790
9095
Elapsed Time (seconds)
(Histogram Bin Width = 0.10 second)

MC Digi + Reco


Trident Beta-v4 - Memory Access Classification
Bandwidth Analysis
Read
Write
Memory Acess (Bytes per sec)
4×10^10
3.5×10^10
3×10^10
2.5×10^10
2×10^10
1.5×10^10
1×10^10
5×10^9
Transaction Analysis
Page Empty
Page Miss
Page Hit
Memory transaction type
0.8
0.6
0.4
0.2
0
306
611
913
1216
1518
1821
2123
2426
2728
3030
3333
3635
3938
4241
4546
4849
5151
5454
5756
6060
6364
6667
6969
7272
7575
7877
8181
8485
8790
9095
Elapsed Time (seconds)
(Histogram Bin Width = 0.10 second)

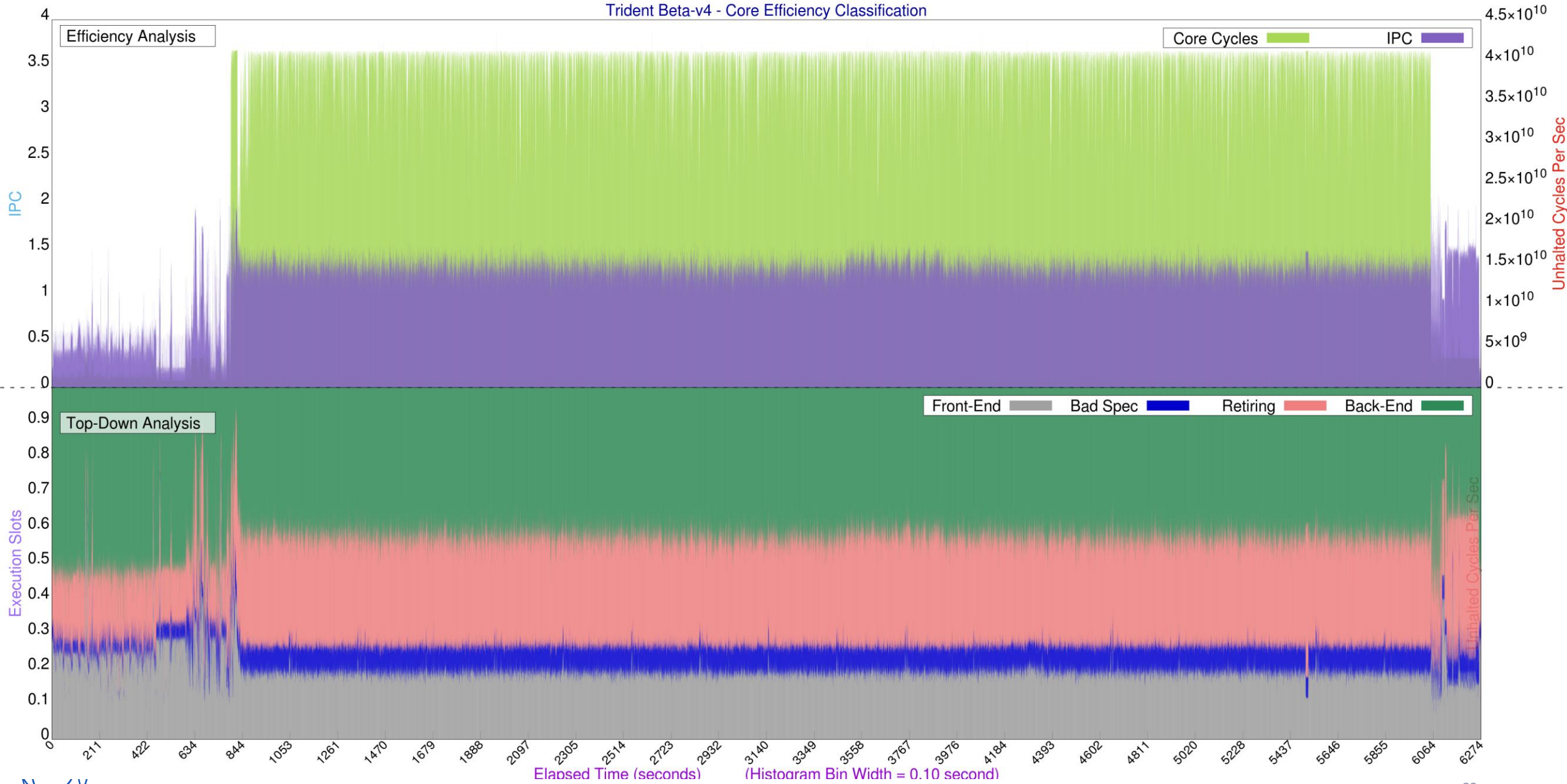
MC Digi + Reco


Trident Beta-v4 - IO Access Classification
Transfer Rate Analysis
Util
Read
Write
IO Access (KB/s)
IO Utilization (%)
Operation Rate Analysis
IO Operations per second
Elapsed Time in seconds
(Histogram Bin Width = 0.10 second)

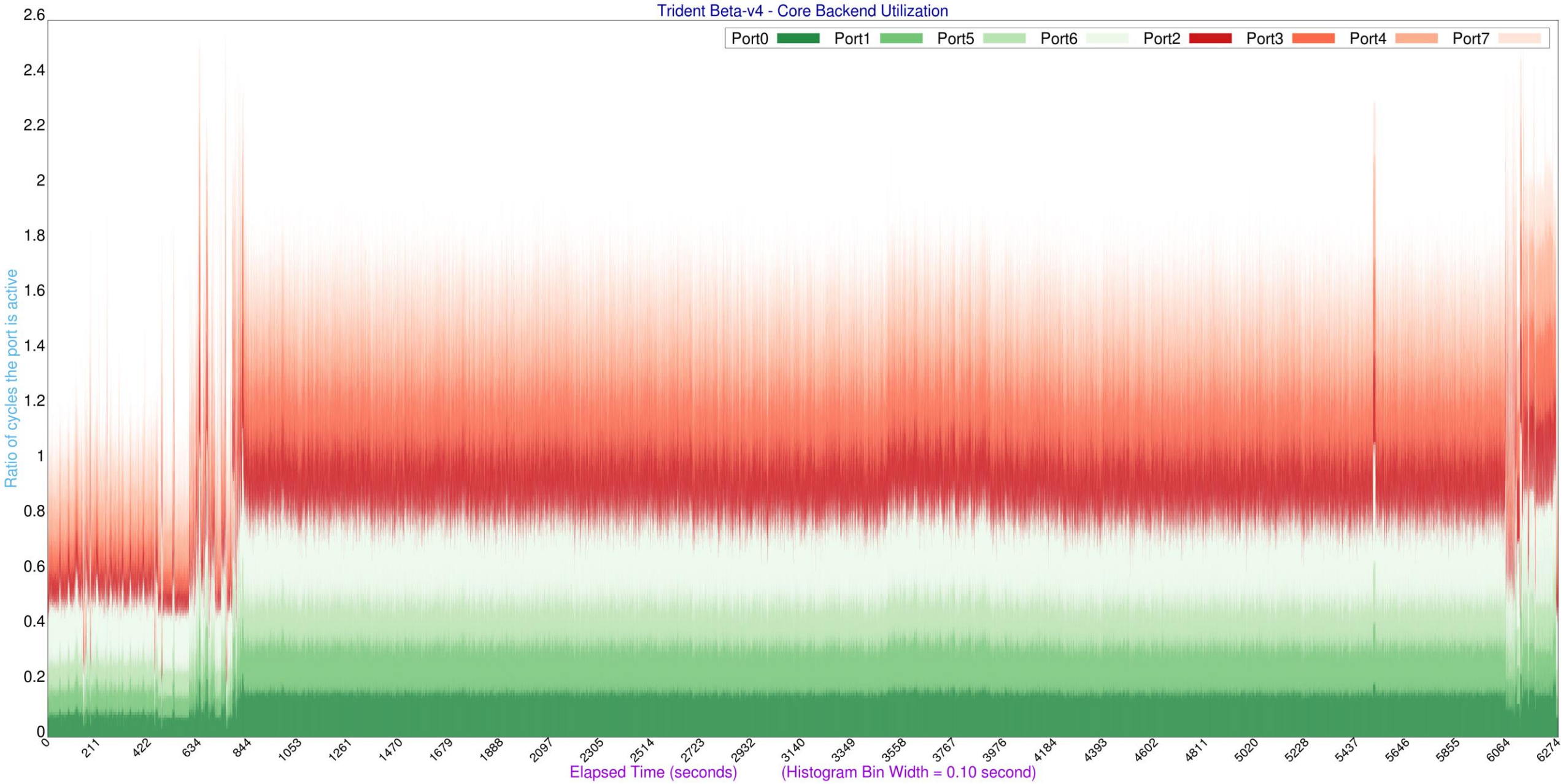
Experiment setup

- Test System : 2x Intel(R) Xeon(R) E5-2630 v3@2.40GHz / 64GB DDR4-2133 RAM / Centos 7 (3.10.0-862.el7.x86_64)
- Workloads
  - HEPSPEC06 Docker Container – Cloud Benchmark Suite (Thanks Domenico Giordano!!!)
    - 450.soplex,471.omnetpp,447.dealII,473.astar,444.namd,453.povray,483.xalancbmk
  - ATLAS Job 1 - Geant4 MC simulation (CPU Intensive)
    - Executes: Sim_tf (Geant4)
    - Input: EVNT files from event generation / Output: HITS files
  - ATLAS Job 2 - MC digitization and reco (CPU+I/O intensive)
    - Executes: Reco_tf with several sub-steps
      - HITtoRDO (Digitization)
      - RDOtoRDOTrigger (Trigger simulation)
      - RAWtoESD + ESDtoAOD (MC reconstruction)
      - POOLMergeAthenaMPAOD0 (merge some small outputs)
    - Input: HITS files from MC simulation + low and high pile-up HITS for digitization / Output: AOD files
  - ATLAS Job 3 - Derivation production (I/O Intensive)
    - Executes: Reco_tf in AODtoDAOD mode
      - AODtoDAOD
      - DAODs are used as the basis of ~all physics analysis in ATLAS
      - DAODs produced from AOD by removing whole events (skimming), whole reconstructed objects (thinning) and variables (slimming)
    - Input: AOD / Output: multiple DAODs in a train

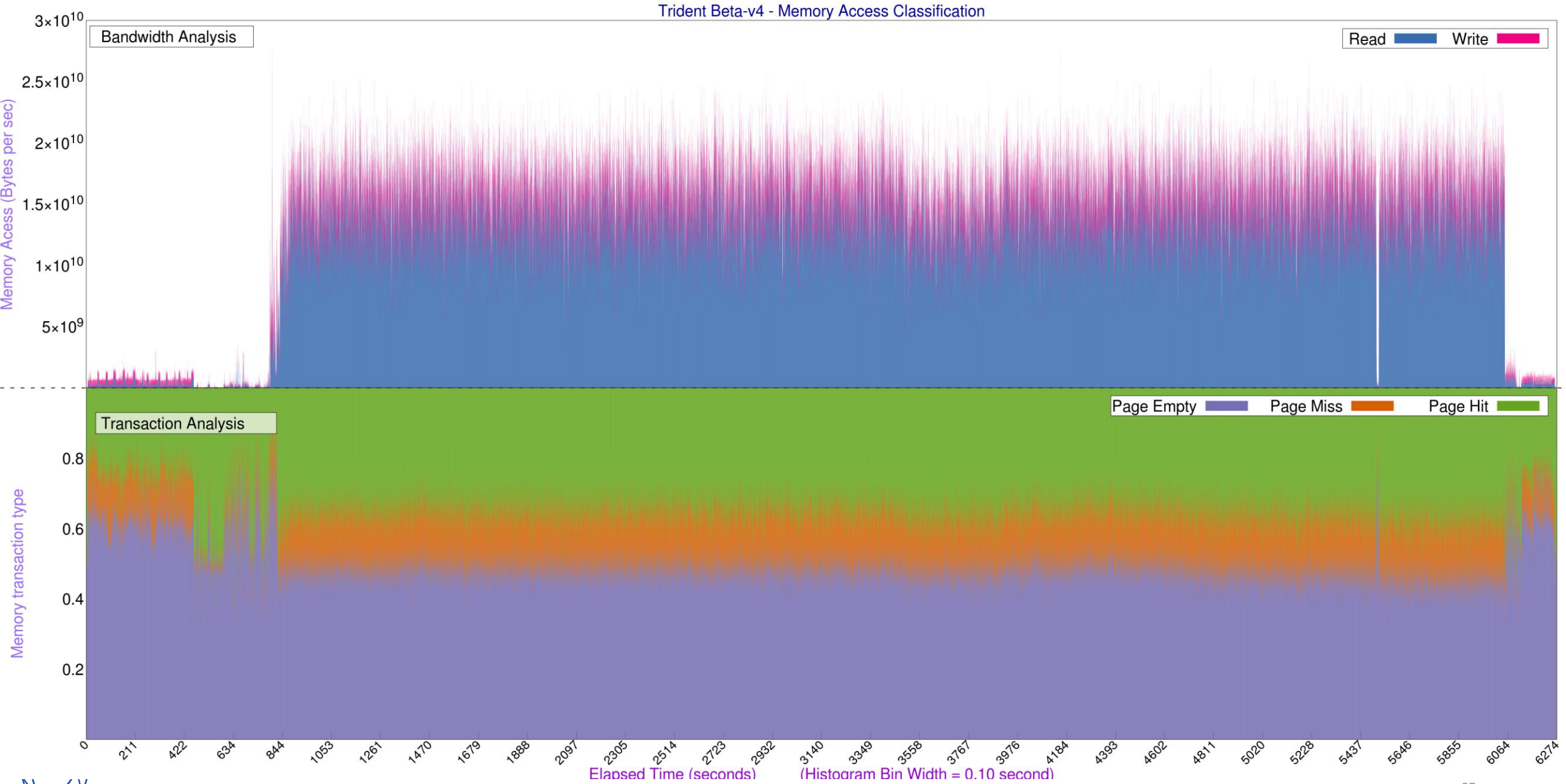
Derivation Production



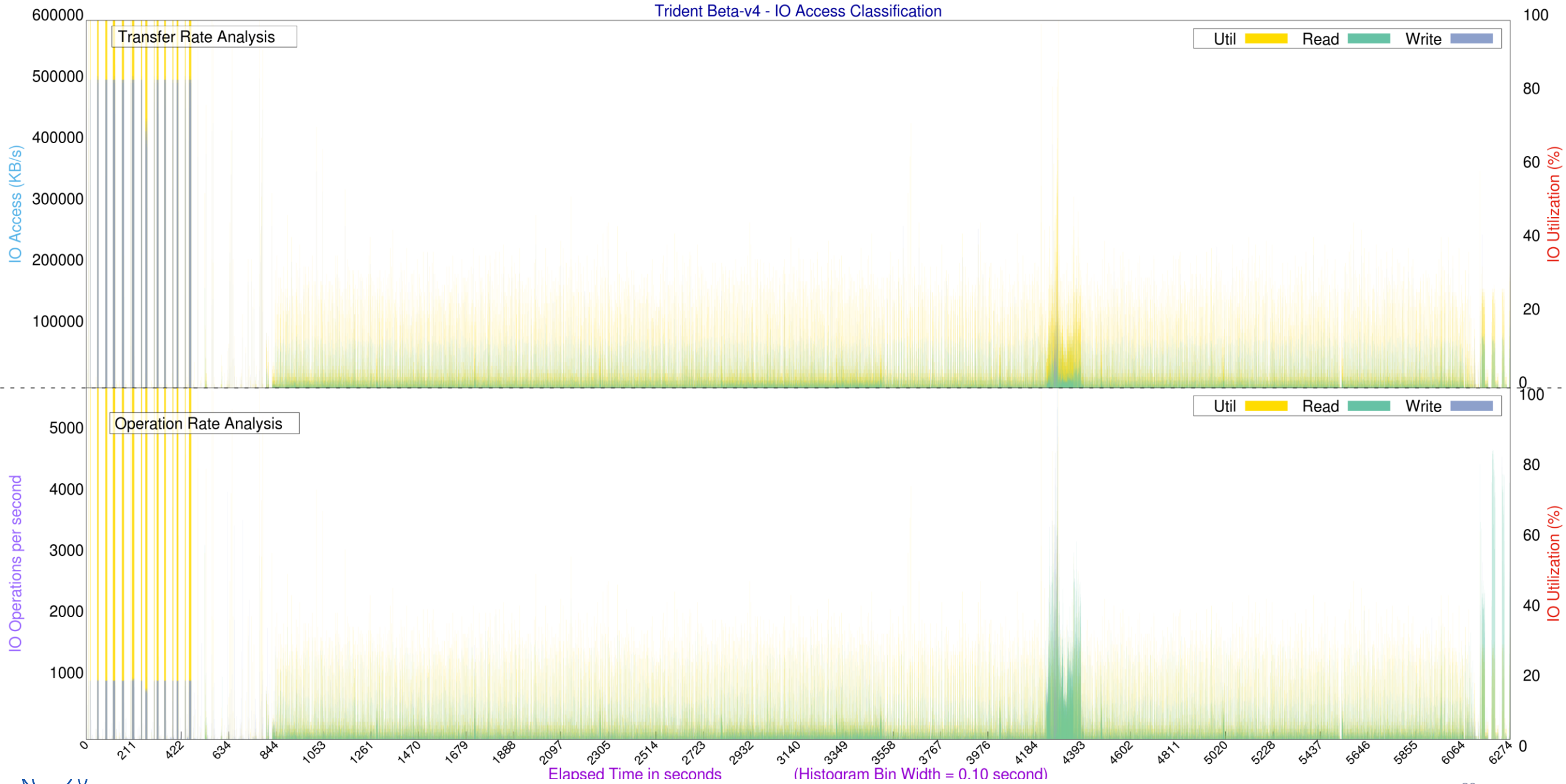
Derivation Production

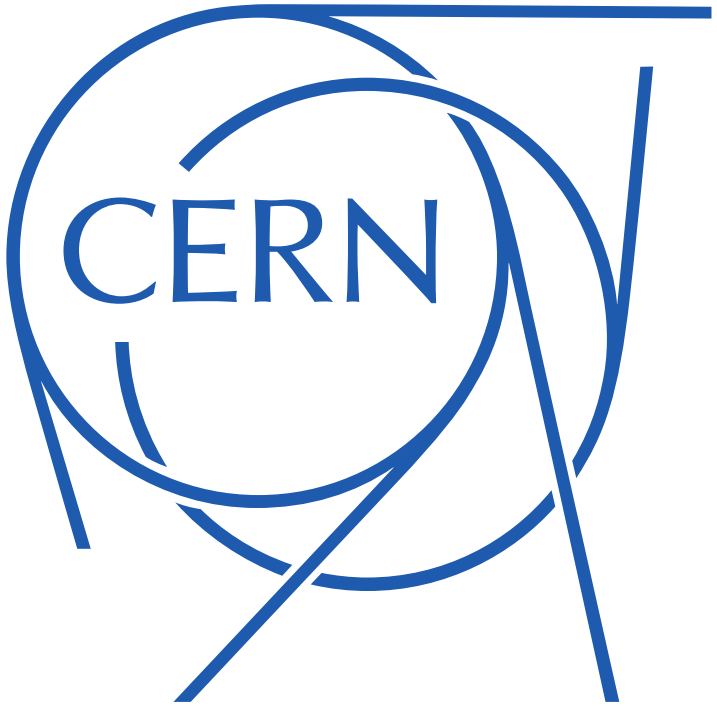


Derivation Production



Derivation Production




CERN