Summer
SOC´14

WattDB – a Rocky Road to Energy Proportionality in Databases

Theo Härder

haerder@cs.uni-kl.de
www.haerder.de

WattDB

TECHNISCHE UNIVERSITÄT
KAISERSLAUTERN

DBIS
Datenbanken und Informationssysteme

Some Facts Enforcing Improved Energy Efficiency

In 2007 the Energy supply of ICT produced CO_2-output at level of 25% of worldwide cars

- ICT today
  - >10% of generated energy (>50 Mio. servers, ? PCs, etc.)
  - >25% CO_2 of cars
- Tomorrow (2020)
  - 100% of generated energy (level of today)
  - >> CO_2 of cars
- It is claimed that energy supply for ICT is larger than that of the entire air traffic

Some Trends ~2004 → 2014

- Magnetic disks

Capacity	400 GB	**x 15**	6 TB
GB/$	0.05	**x 600**	30
IOPS	200	**x 1**	200

- Solid state disks (flash memory)

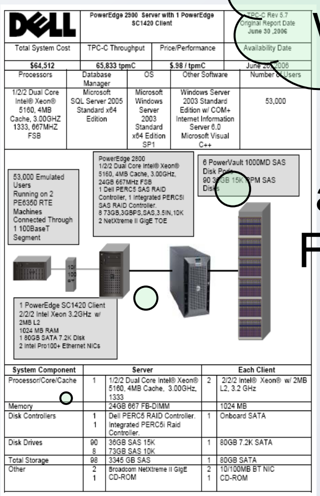
Capacity	16 GB	**x 30**	480 GB
GB/$	0.0005	**x 3,000**	>1.5
IOPS (4KB read)	1,000 (SCSI)	**x 1,000**	1,000,000+ (PCIe)
			5,000+ (SATA)
IOPS (4KB write)	50 (SCSI)	**x 10,000**	500,000+ (PCIe+RAM)

- Phase-change memory

Capacity			1 GB chip (20-nm)
IOPS (64B read)			20,000,000+ (1 chip)
IOPS (64B write)			1,000,000+ (1 chip)

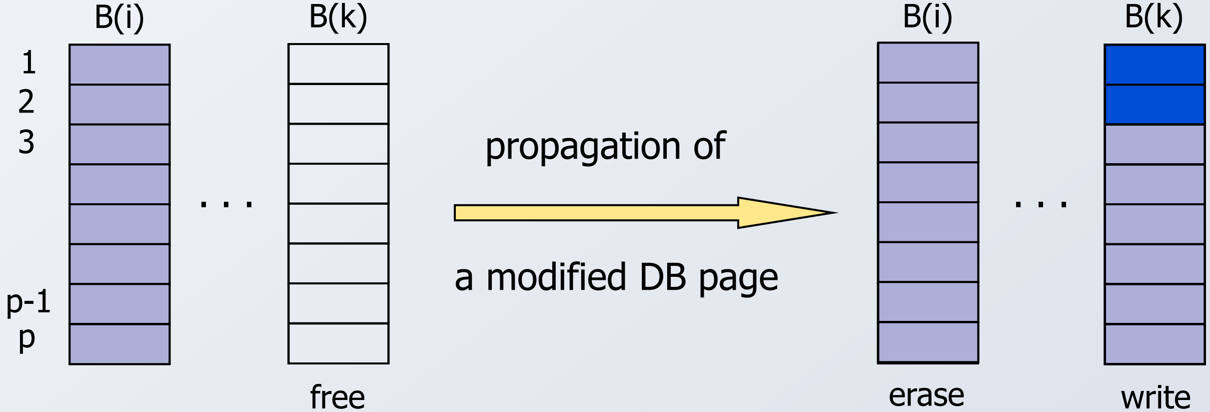
DBIS Datenbanken und Informationssysteme

Flash Usage in DB Servers?



replace with 1 10TB disk and 3 FLASH disks

- Guarantees persistent data with (almost) zero-energy needs when idle or turned off
- A flash block B, much larger than a disk block,
  - contains p (typically 32 – 128) fixed-size flash pages with 512 B – 2 KB
  - NAND logic does not enable direct update of pages → erasure
- Reads of individual flash pages
- Update of flash pages not possible;
  only overwrites of entire blocks where erasure is needed first



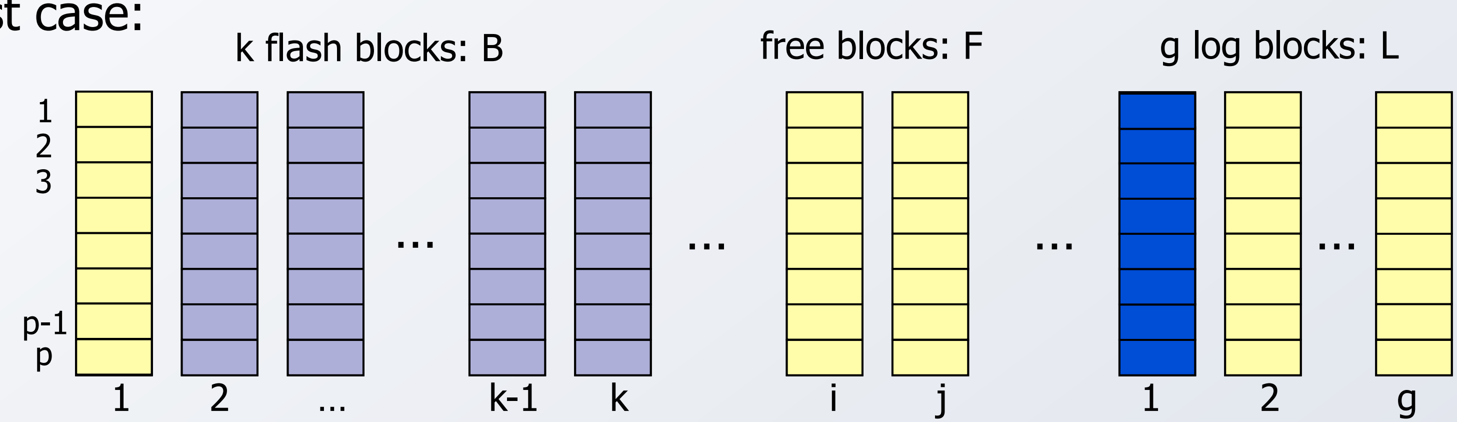
Wear leveling automatically allocates a new flash block and preserves cluster property

Built-in Wear Leveling

- Flash-internal write optimization
- Simplest form of mapping: 1:1 - block level

Metadata (flash directory) must be in RAM

Best case:



Switch: L1 becomes B1, Erase old B1 → 1 erasure

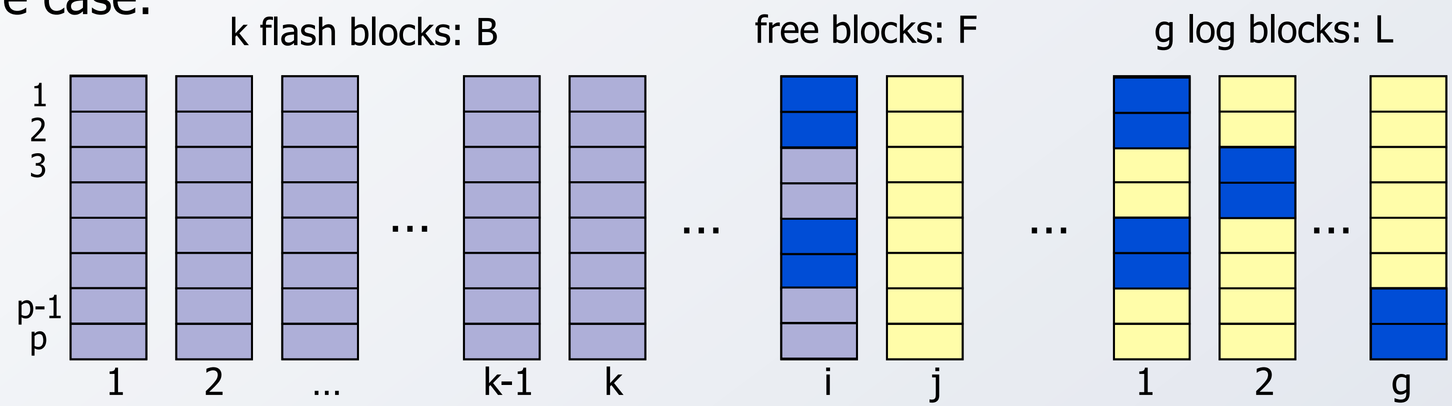
http://hothardware.com/News/OCZ-and-Indilinx-Collaborate-On-New-SSD-Garbage-Collection-Scheme/

Built-In Wear Leveling (2)

- Simplest form of mapping: 1:1 - block level

  Metadata (flash directory) must be in RAM

Some case:



Merge: L1 and B1 to Fi

Erase B1 and L1 → 2 erasures

Other forms of mapping: n:1 (n:m) - block level (page level)

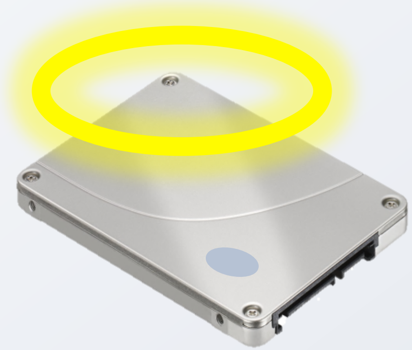
Merge of n flash blocks and one log block to Fi → n+1 erasures

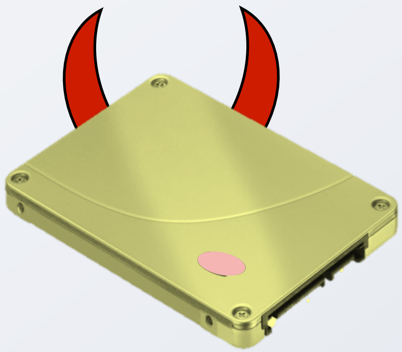
Growing flexibility ←→ higher complexity of flash management and block merge

SSDs are a disruptive I/O technology
Performance behavior
Energy consumption
Energy-proport. computing
Benchmarking/ measurements
Architectural concepts of WattDB
WattDB – how to achieve the goal?
Experiments – processing layer
Experiments – symmetric cluster
Comparison: big server vs cluster

Flash Translation Layer

- The good
  - Wear leveling
  - Garbage collection
  - Performance gain





- The bad
  - Black box

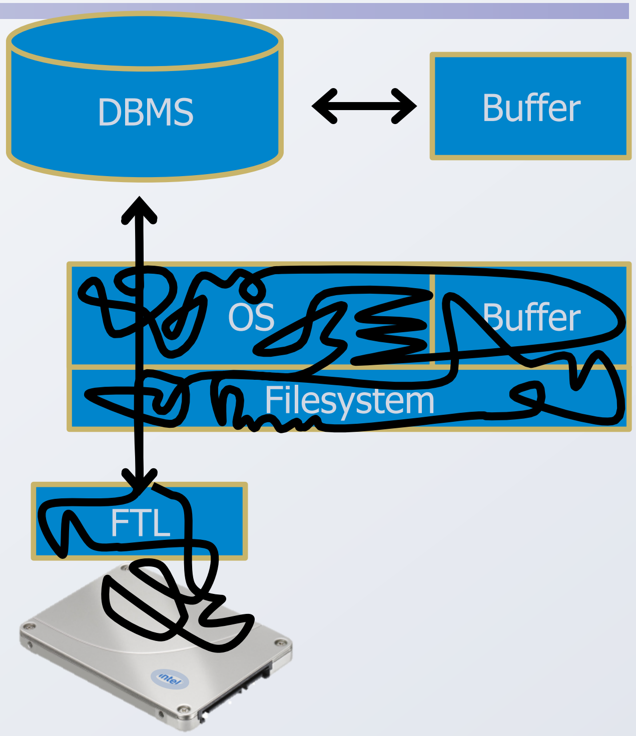
- The ugly
  - Behavior is totally unpredictable

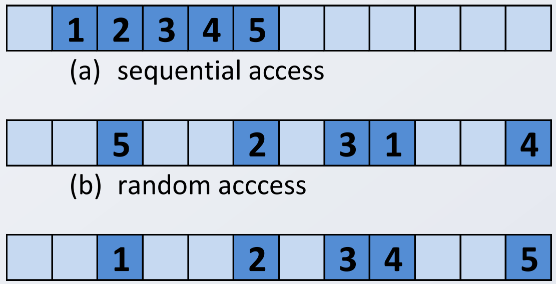
➔ DBMS people are "control freaks"

SSDs are a disruptive I/O technology
Performance behavior
Energy consumption
Energy-proport. computing
Benchmarking/ measurements
Architectural concepts of WattDB
WattDB – how to achieve the goal?
Experiments – processing layer
Experiments – symmetric cluster
Comparison: big server vs cluster

Measurement Setup

- Direct I/O
- Read & write
- IOmeter & our own tool
- 5 SSDs
  - SSD1: SuperTalent
  - SSD2: Mtron
  - SSD3: Intel Generation 1
  - SSD4: Intel Generation 2
  - SSD5: Crucial RealSSD
- 3 access patterns





(a) sequential access

(b) random acccess

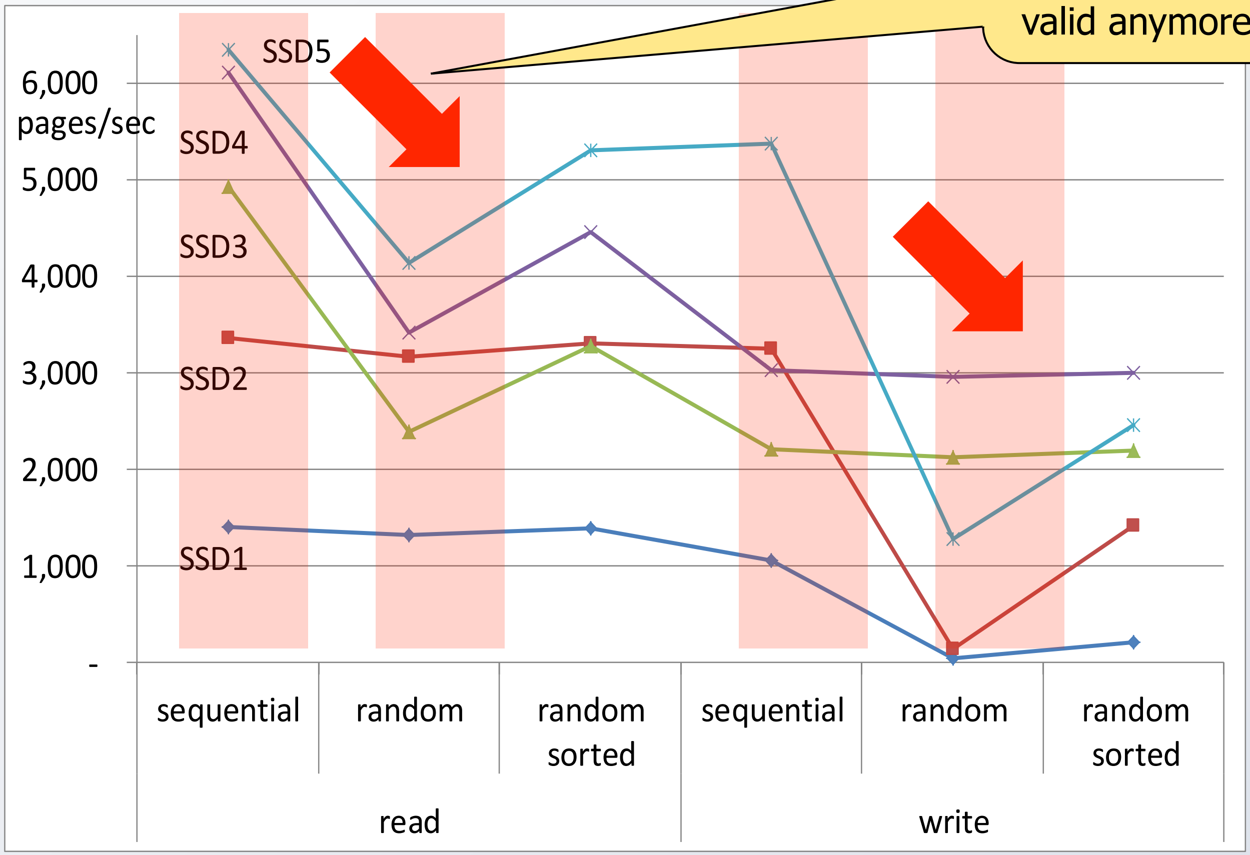
(c) skip-sequential access

- different page sizes
- empty vs. full devices

Myth 1: Random Access is as Fast as Sequential Access

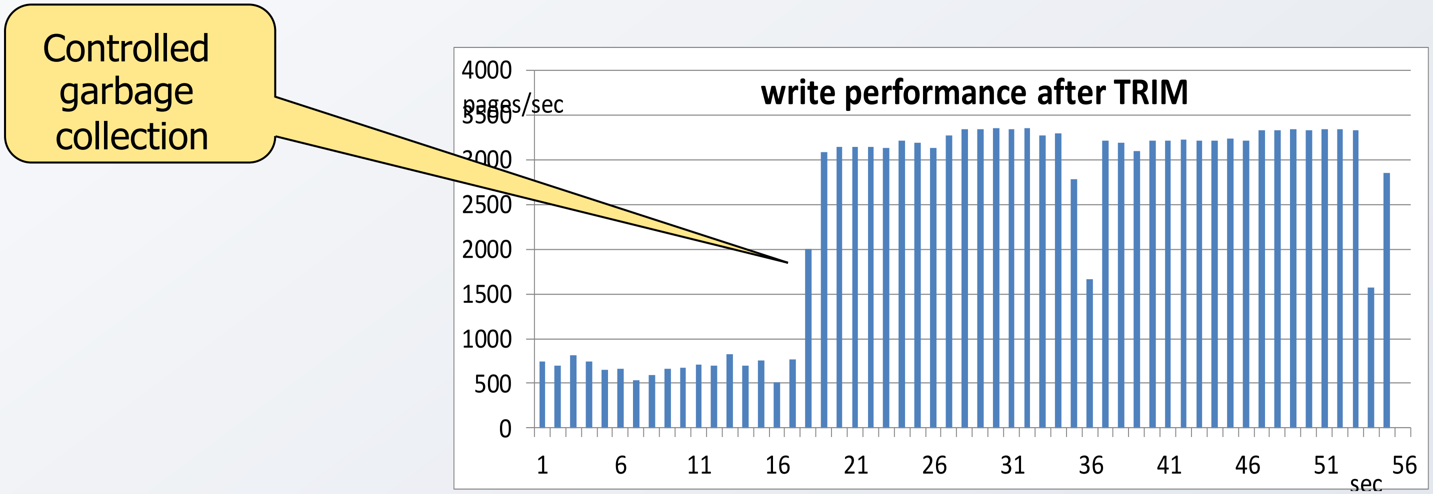
- Random Access

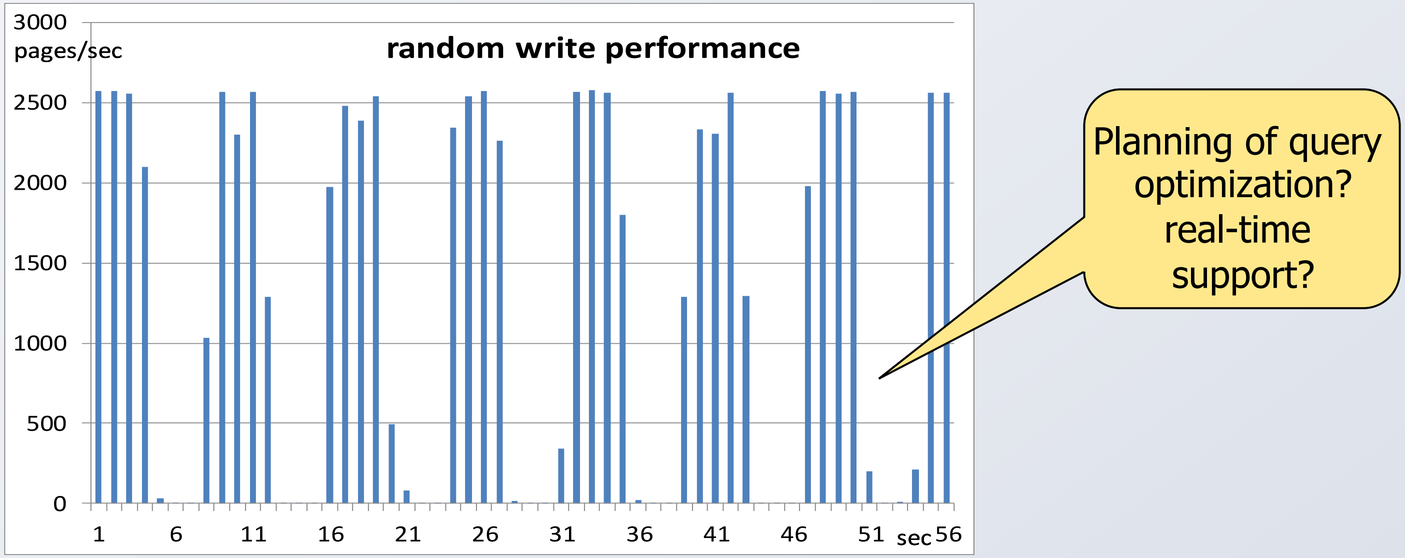
Rule of thumb of conventional systems is not valid anymore!



Myth 2: SSDs are Incredibly Fast

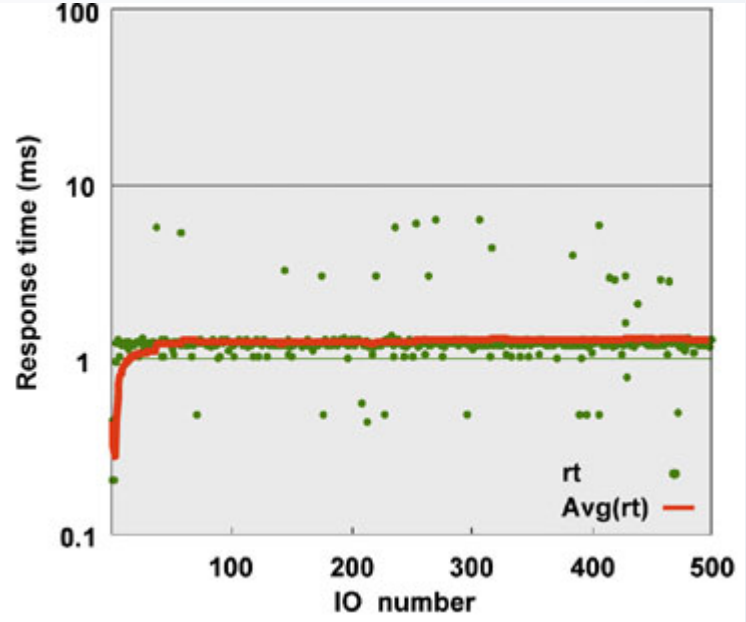
- Unstable Behavior

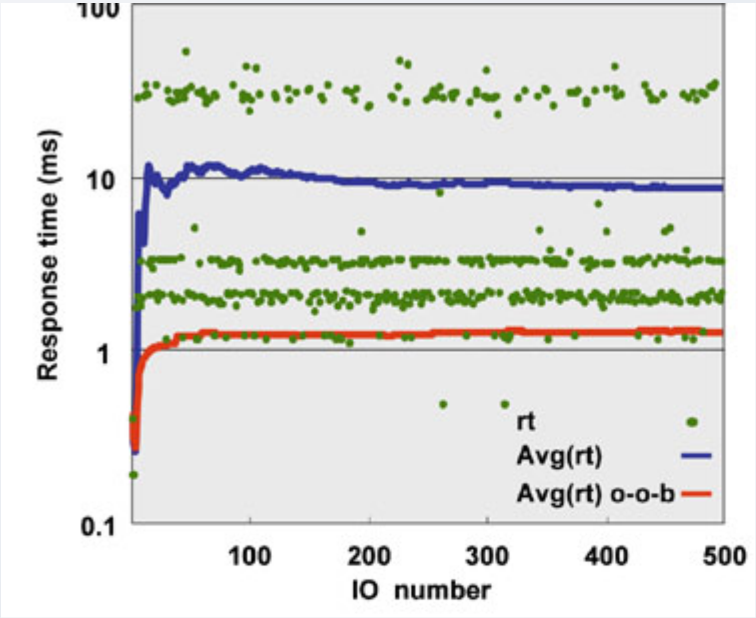




Myth 3: Predictable Performance?

- Device state: random writes (see also Bonnet, Bouganim, Koltsidas, Viglas, VLDB 2011)





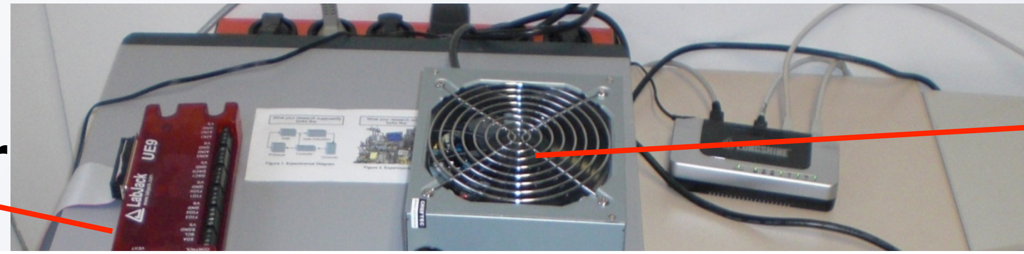
SSD state: out of the box after filling the device

- Performance behavior is dependent on
  - Device type (technology)
  - FTL mapping (manufacturer)
  - Device state (workload)
  - Aging, …

What about Energy?

What our research *actually* looks like:

A/D Converter



Power Supply

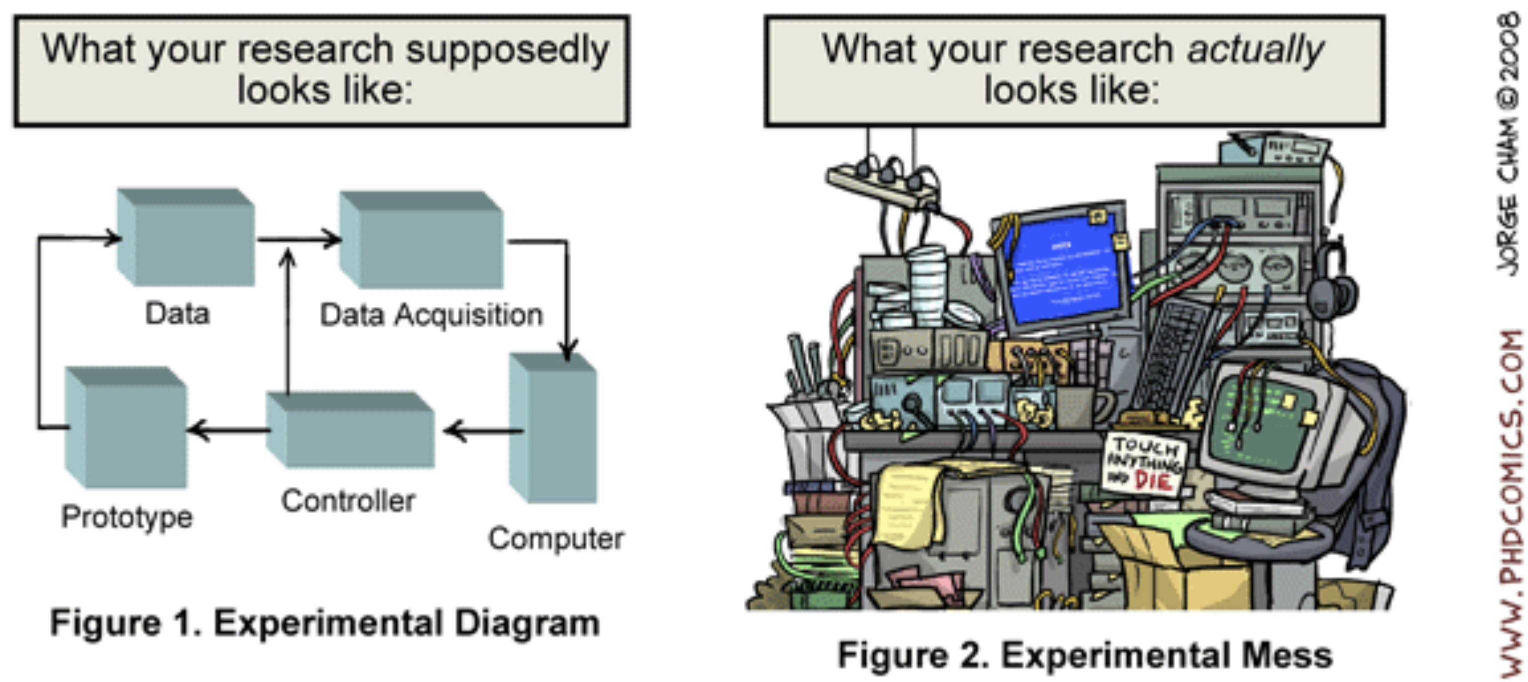
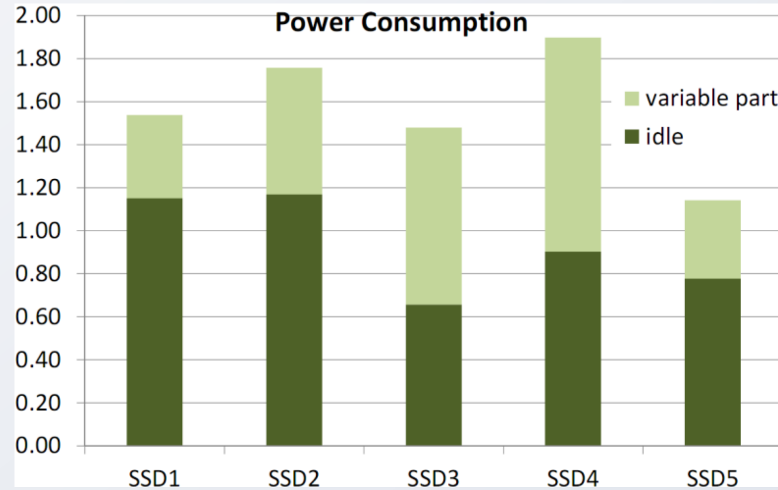


Figure 1. Experimental Diagram

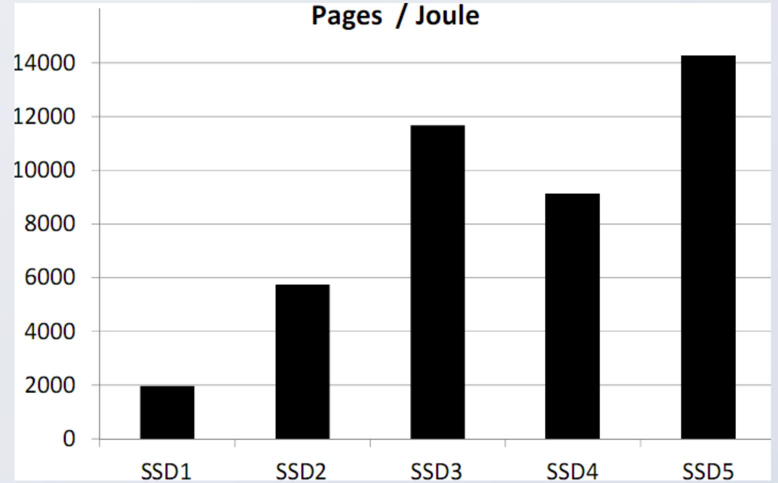
Figure 2. Experimental Mess

Energy-related SSD Measurements

- Absolute power consumption (W): sequential read pattern
  - Not as energy saving as expected
  - Higher consumption for writes
  - Consumer disks: ~4 – 6 W
  - High-end disks: ~9 – 14 W
  - HDD3/SSD3:
    ~15 (idle), ~8 (peak)



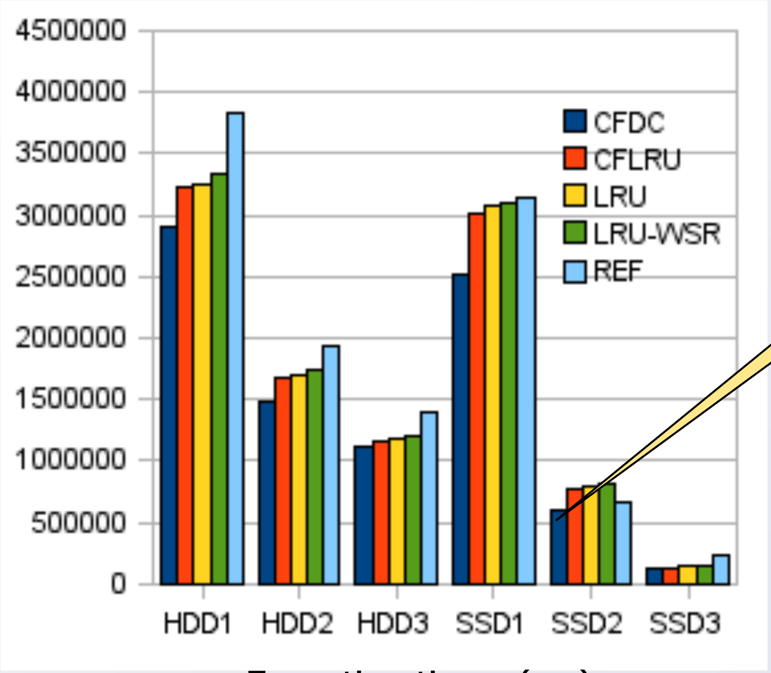
- Pages read per Joule (W * s)
  - Energy efficiency is constantly improved
  - On disks:
    ~600 – 1800 pages/J



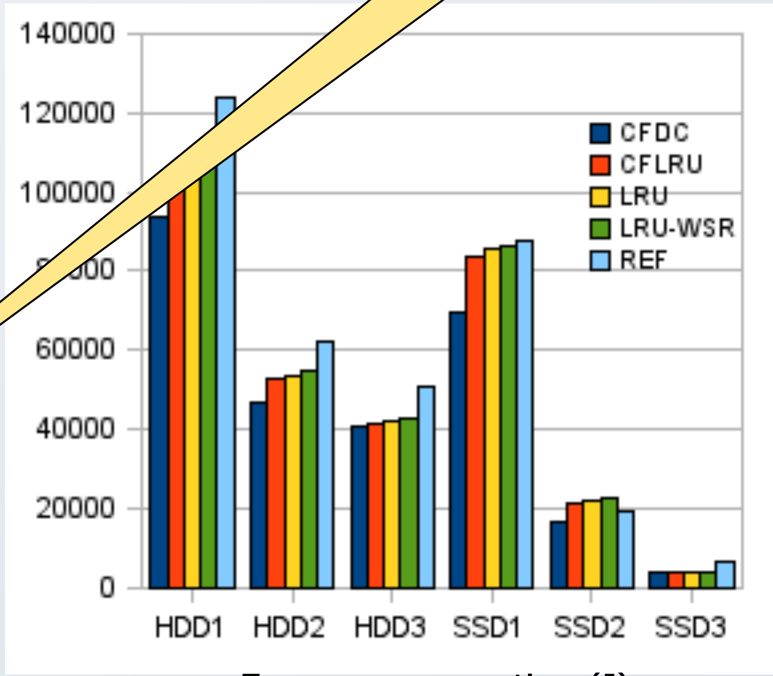
Findings in DBMS Buffer Management

- Objectives to exploit read/write asymmetry
  - Replacement decision for read-only or modified pages
  - Basic principles of CFDC (Clean-First Dirty-Clustered)
    - Minimize number of physical writes
    - Address write pattern to improve write efficiency
    - Keep a relatively high hit ratio (even with SSDs)
  - Behavior of algorithms in heterogeneous environments
  - How energy efficient is a given algorithm?

Performance of CFDC is remarkable

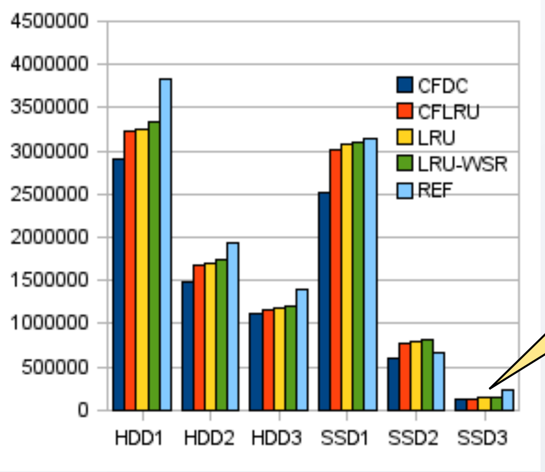


Execution times (ms)

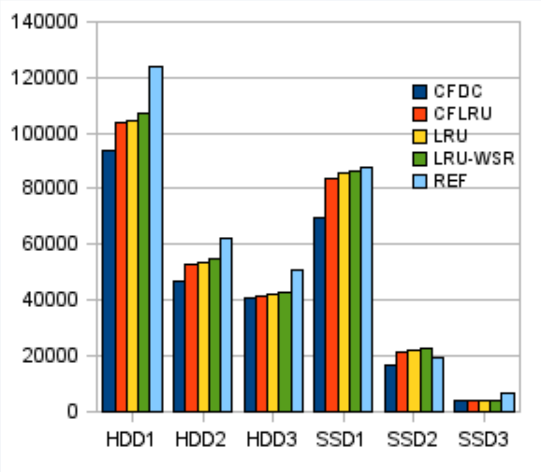


Energy consumption (J)

Key Observations



Buffer opt. less significant; larger device cache on SSD3

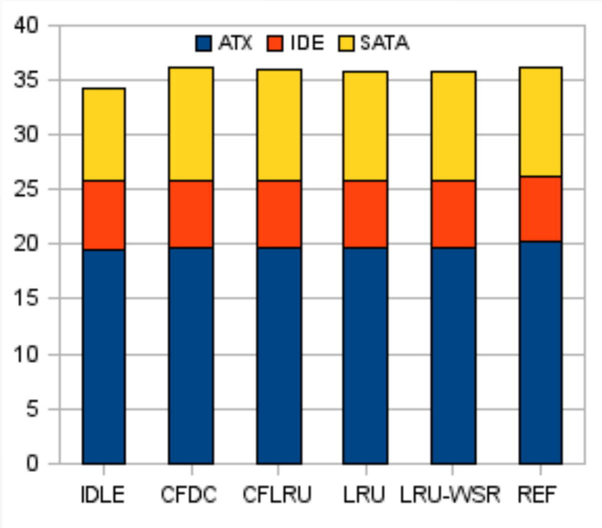


- This optimization behavior is indicative for what we can expect under the different storage device settings!
  - IO cost is steadily reduced
  - SSD saving potential more and more disappears
- Observation confirms general thesis (Tsirogiannis et al., 2010)

➔ **The most energy-efficient configuration is typically the highest performing one within a single node intended for use in scale-out architectures**

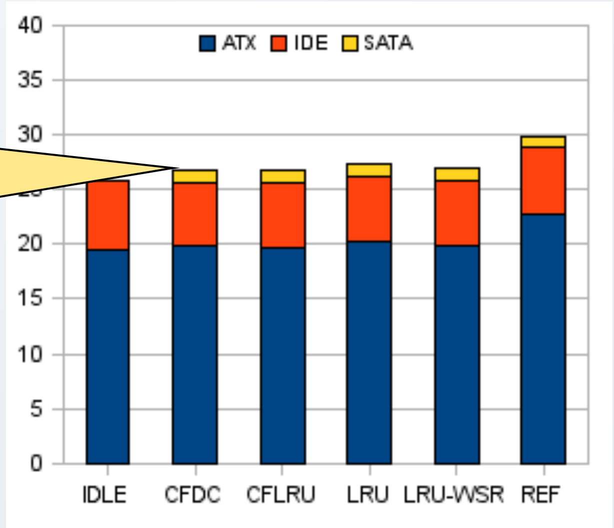
Break-down of Average Power (W)

HDD3



Energy optimization potential disappears

SSD3



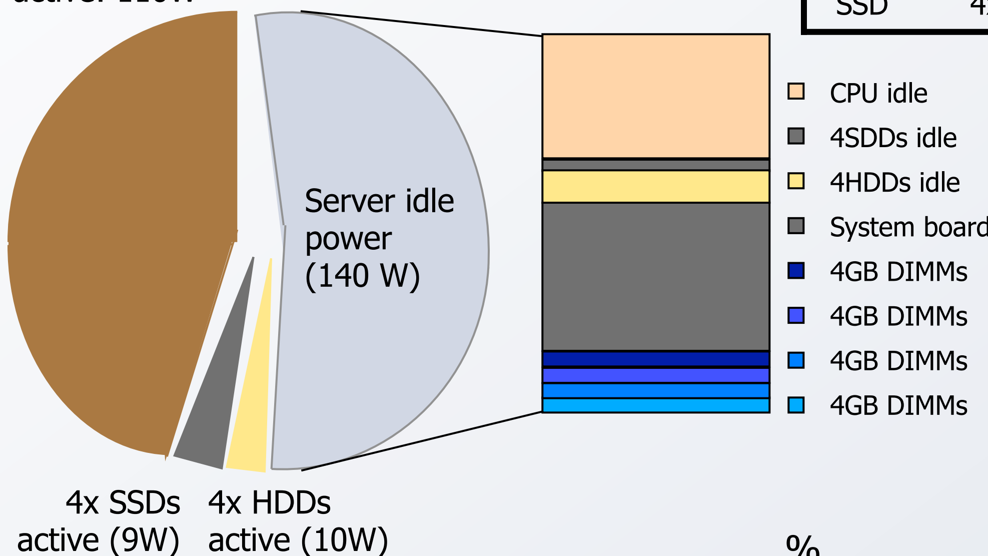
- Identical environments: ATX and IDE remain unchanged
  - Ideally, power consumption should linearly depend on system utilization
  - No difference between "idle", "working", and "peak"
  - Avg. system utilization varies for the individual trace executions

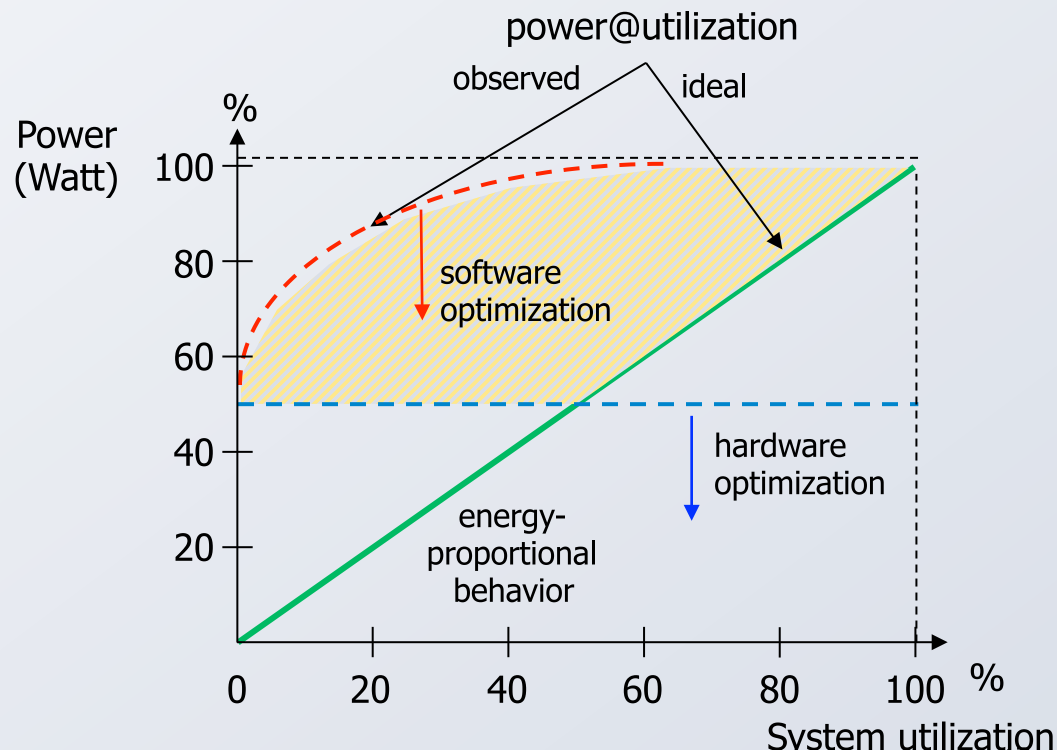
➔ Components are not energy proportional

Optimization increases "idle" times in low-utilization environments
→ overall energy efficiency may not be improved!

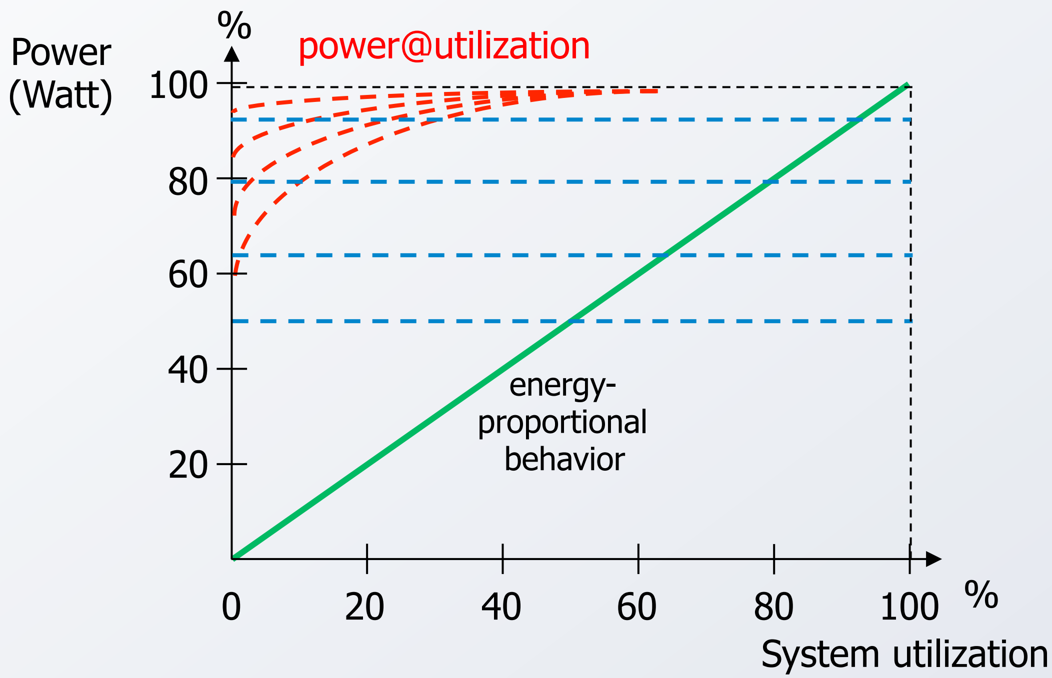
The Entire Picture

CPU	2xIntel Xeon 4-Core 2.66GHz	48W	160W
RAM	8x4GB FB-DIMMS	80W	80W
HDD	4x250GB Seagate 10K 2.5"	16W	24W
SSD	4x64GB Intel X-25E 2.5"	0.2W	10W





Is Growth of Main Memory the Solution?



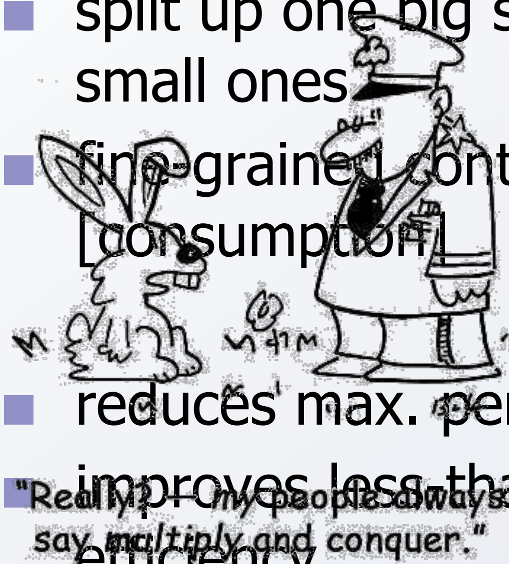
➔ In-memory data management assumes continuous peak loads!

➔ Energy consumption of memory linearly grows with size and dominates all other components across all levels of system utilization

An Energy-Proportional DBMS

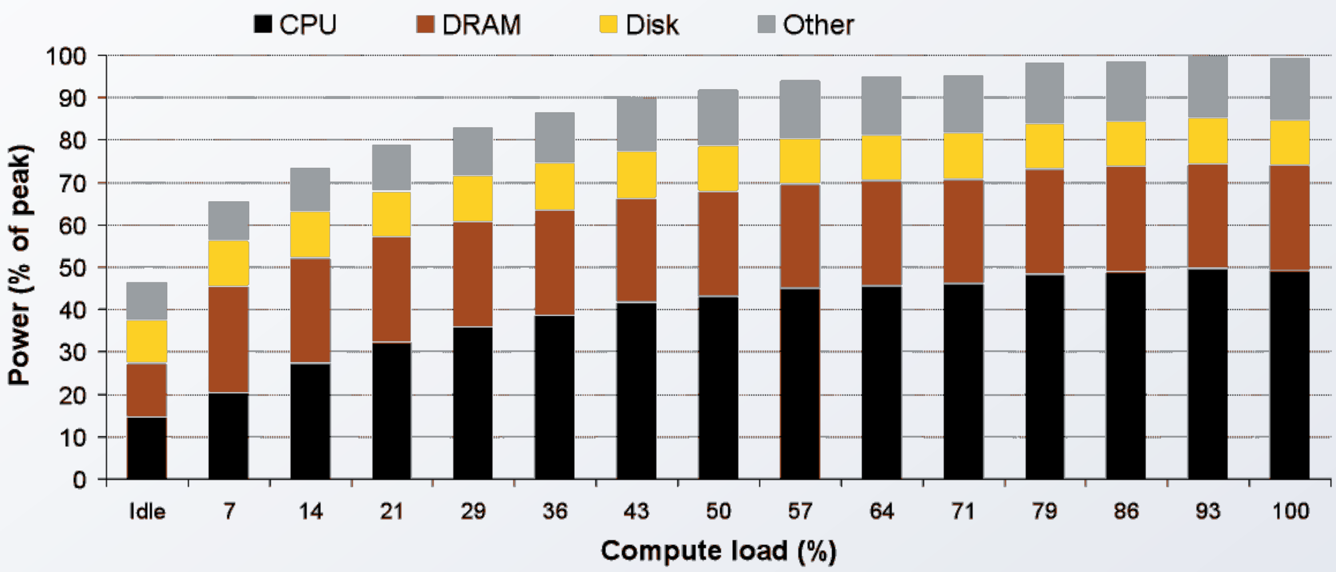
Divide and conquer

- split up one big server into a cluster of small ones
- fine-grained control of power [consumption]
- reduces max. performance
- improves less-than-peak energy efficiency



DBIS
Datenbanken und Informationssysteme

Energy Efficiency of DBMS

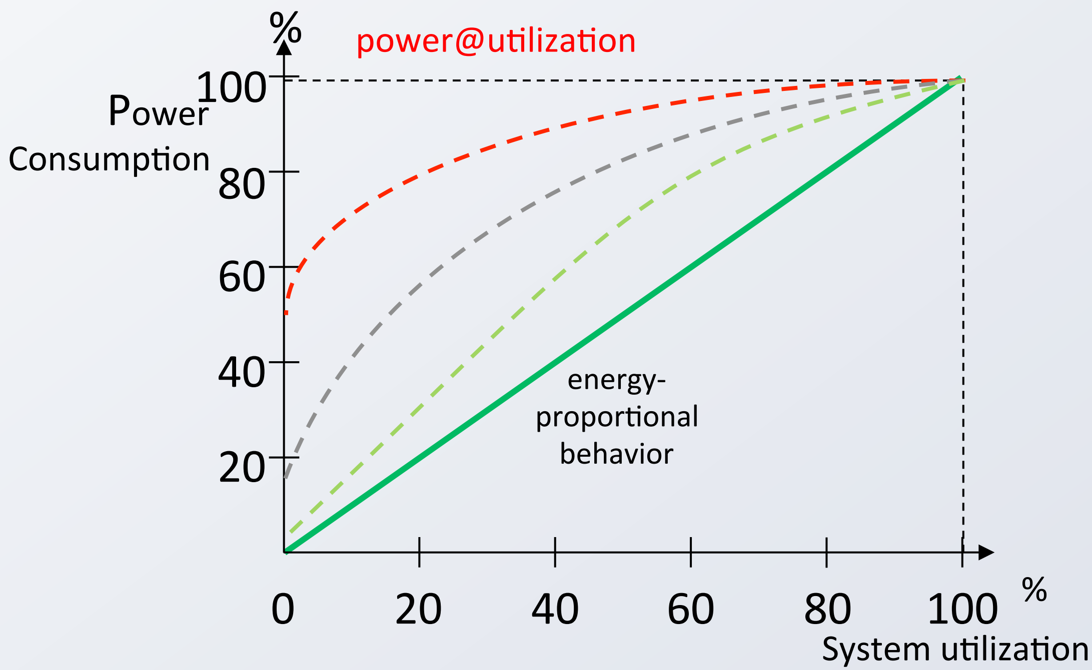


- EE = (relative load / rel. energy consumption)
- Best performance at 100% load, i.e. efficiency = 100%
- Efficiency quickly drops:

load	90 %	70 %	50 %	30 %	10 %
efficiency	90 %	73 %	55 %	35 %	14 %

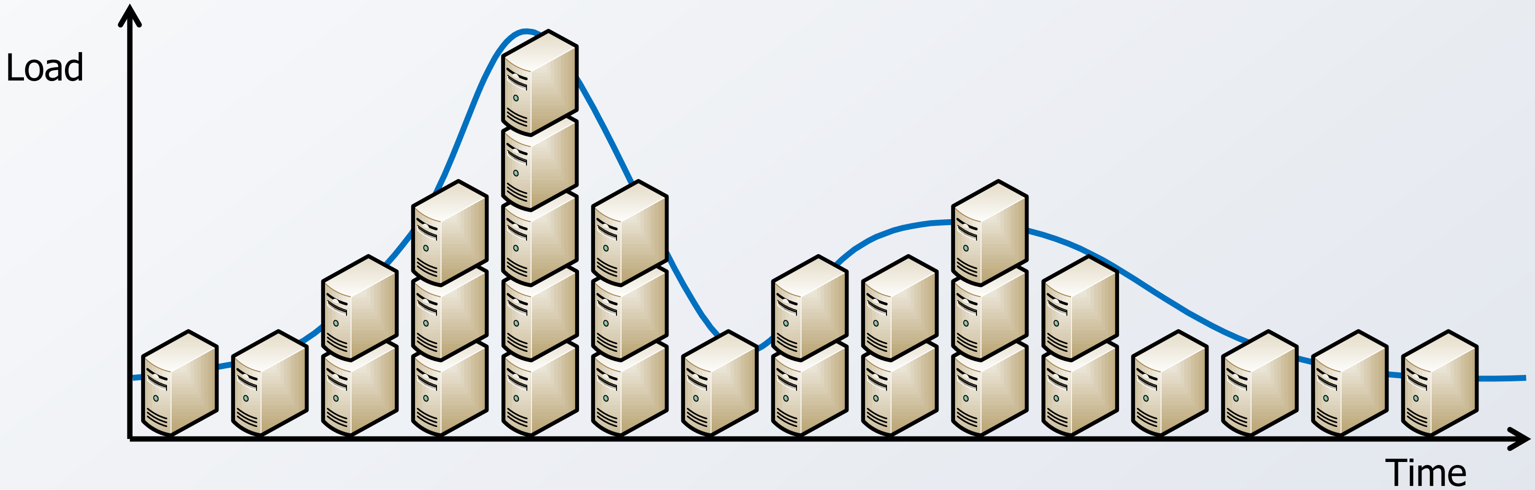
~~Energy Efficiency~~ Energy Proportionality

load	**90 %**	**70 %**	**50 %**	**30 %**	**10 %**
efficiency	100 %	100 %	100 %	100 %	100 %



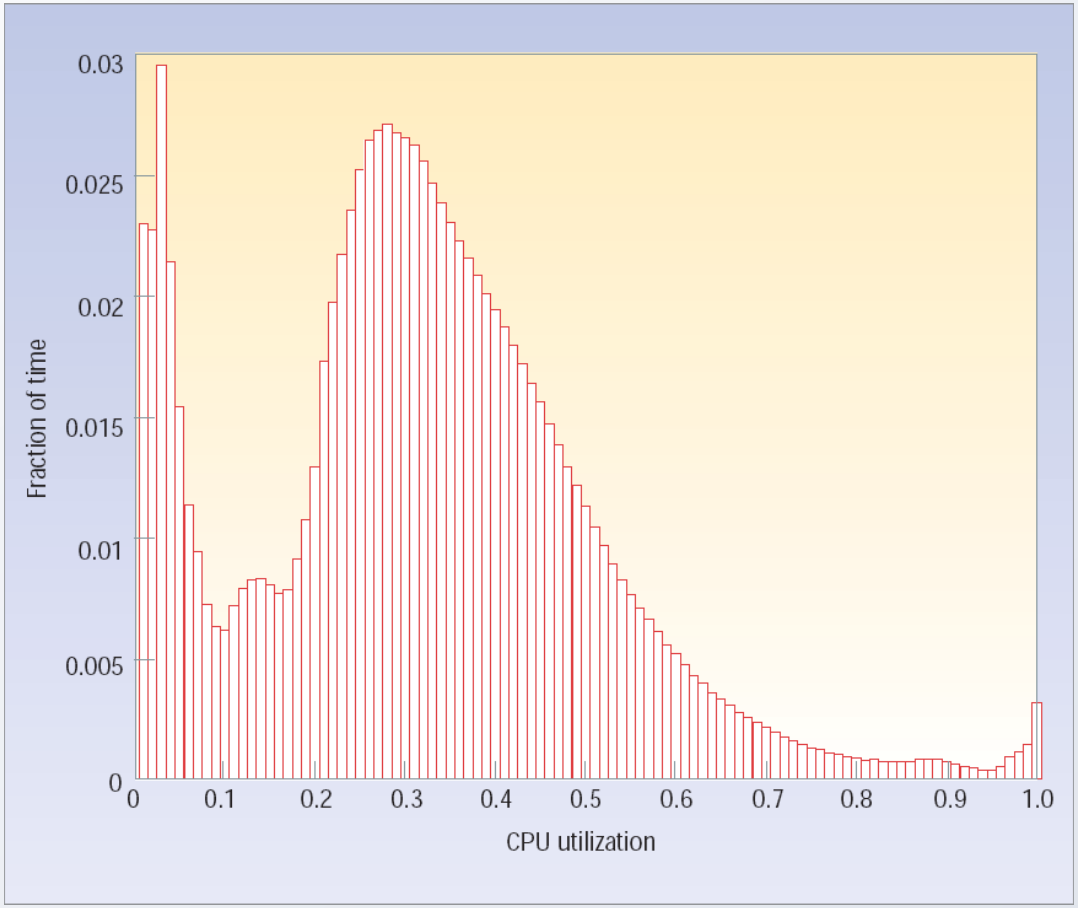
- Energy consumption proportional to system utilization

A Dynamic Cluster of Wimpy nodes



- Key questions
  - Time span to disconnect low-utilized nodes from the cluster or to re-activate switched-off nodes in case of overload
  - Flexible physiological partitioning of DB data

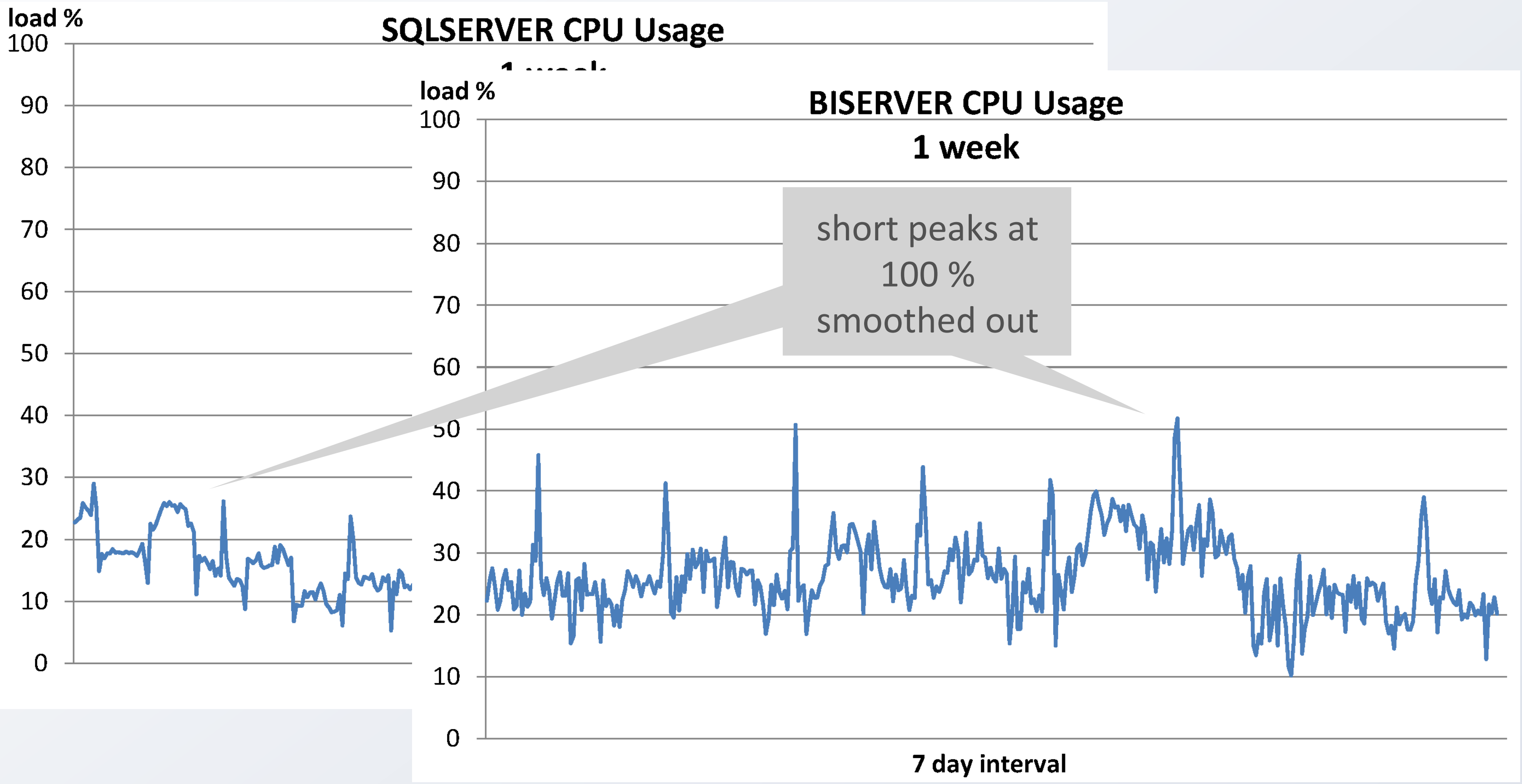
CPU Utilization of DBMS



Average CPU utilization of more than 5,000 servers,

see A. Barroso and U. Hölzle: The Case for Energy-Proportional Computing

CPU Utilization of DBMS

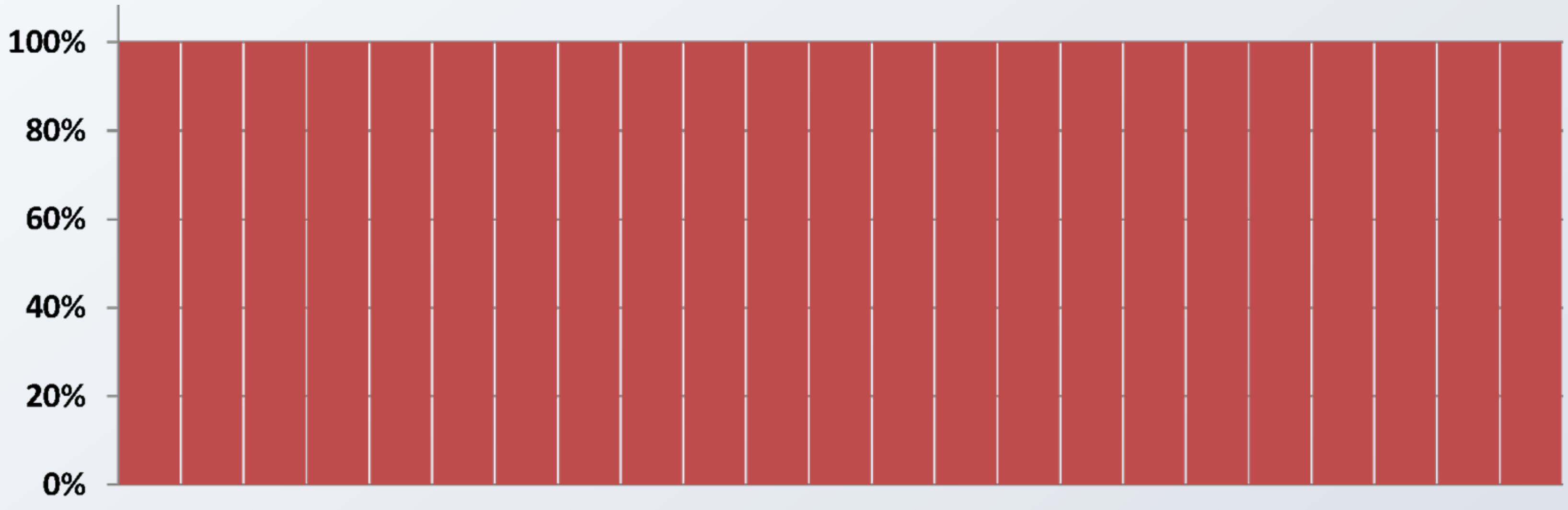


Study by SPH AG, Stuttgart

Monitoring for 1 week, ERP backend & analysis servers

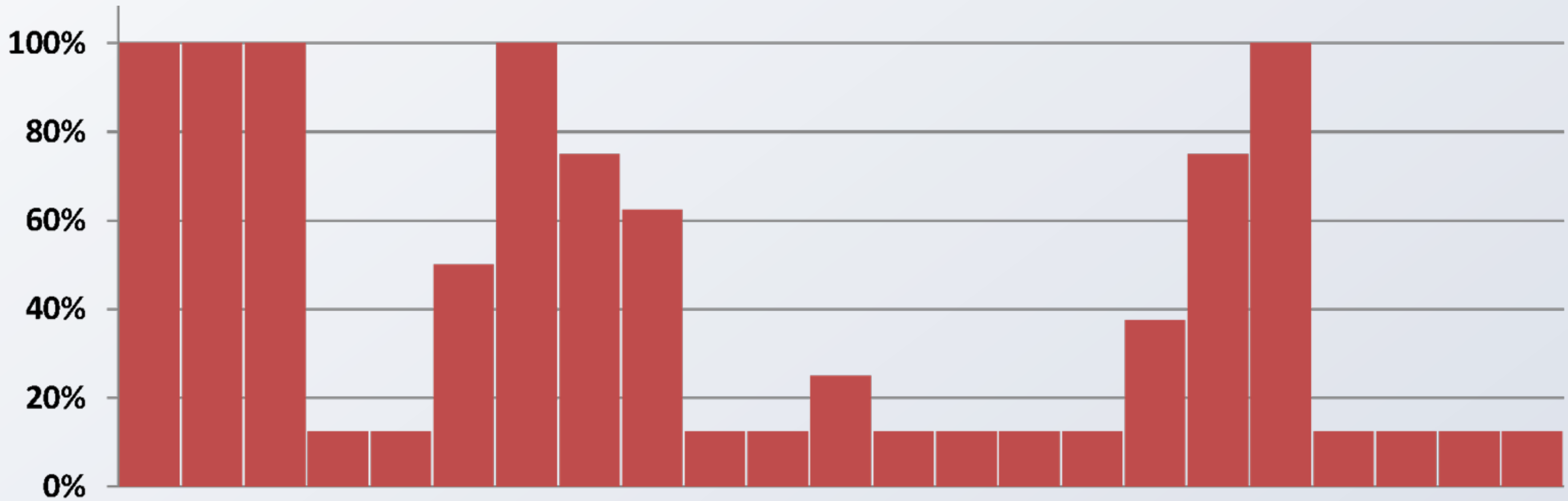
Benchmarking for Energy Efficiency

- Traditional Benchmarking Paradigms
  - Measure performance
  - Run at 100% utilization all the time
  - The more, the better



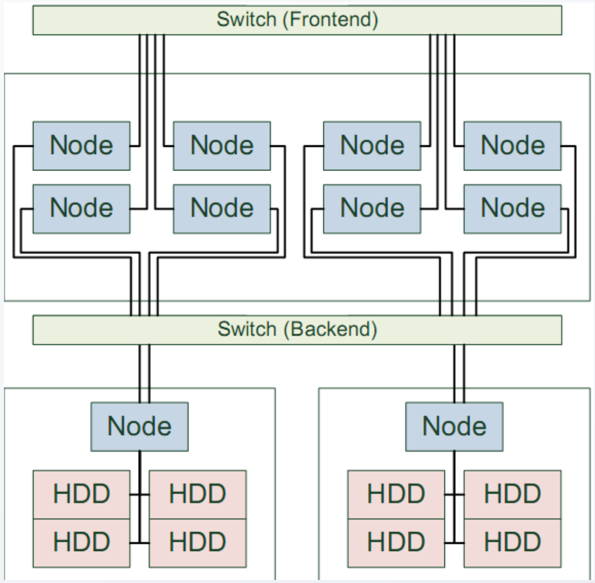
Benchmarking for Energy Efficiency (2)

- Benchmarking for energy efficiency
  - Define realistic workloads
  - Introduce idle/non-peak times
  - Honor energy savings w.r.t. typical utilization patterns



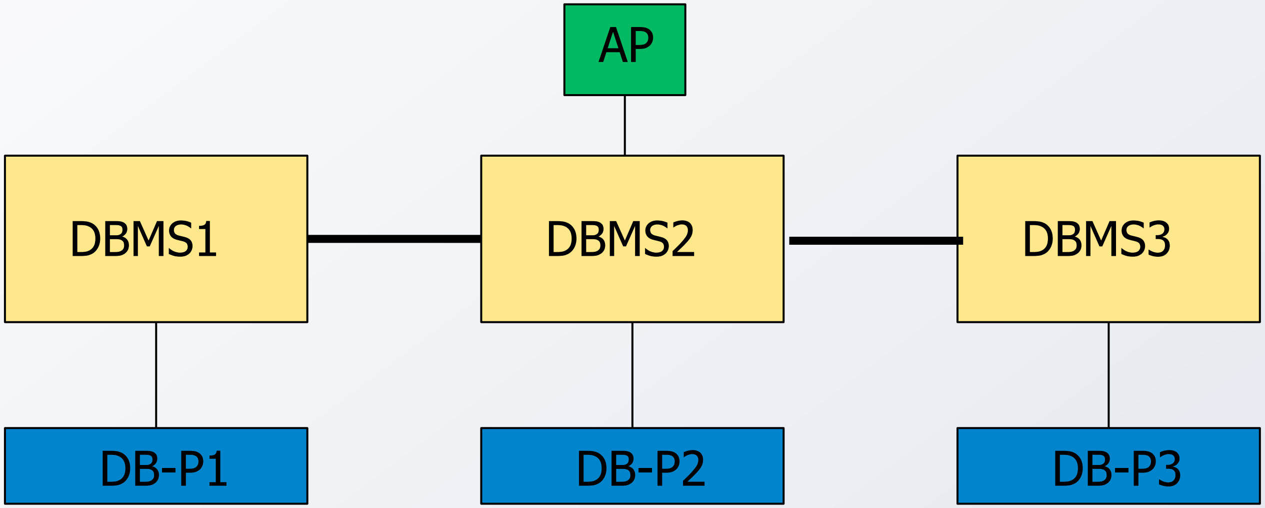
DBIS Datenbanken und Informationssysteme

Architecture Overview



- Minimal system configuration: a single node
  - storage mapping, query processing, cluster coordination
  - HW is Amdahl-balanced (node with 2GB memory < 30 W)
- All storage devices (SSD, HDD)
  - Are dynamically shared by all nodes
  - Shared-Nothing processing architecture of the cluster has to be supported by an emulated Shared-Disk I/O architecture

Cluster Design Principle – Shared Nothing

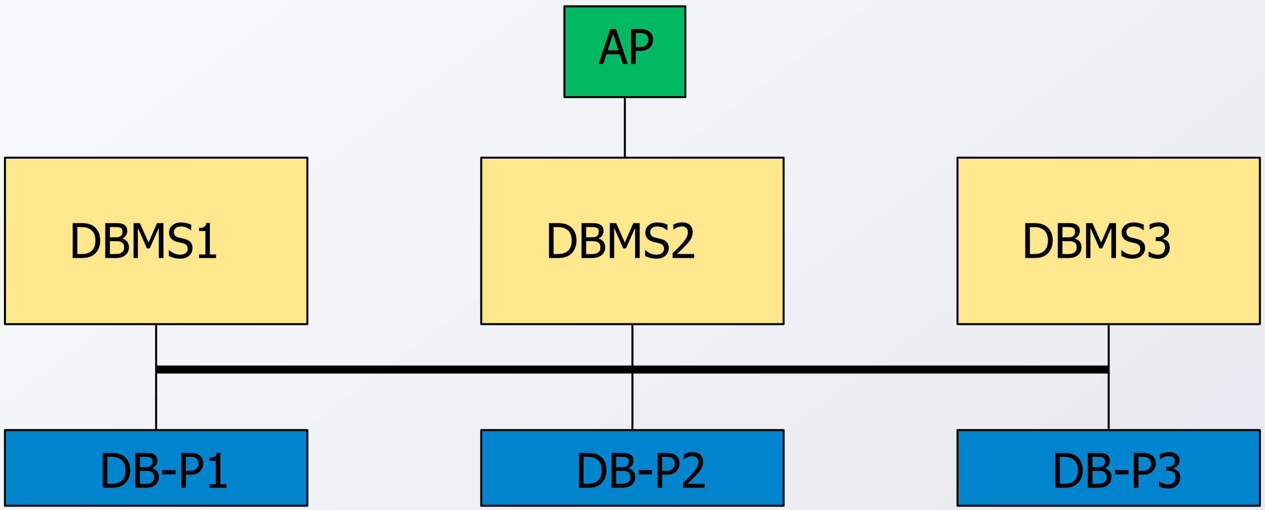


- "The load follows the data"
  - Distributed execution of a transaction
  - DBMS implementation: similar to a single server
  - Scalability of computing power! → scale-out!
  - Load Imbalance/Scalability of data??
    → physical repartitioning!
  - Data replication increases reliability and availability
    → failure handling without data redundancy?
  - Distributed control (e.g. CC) is difficult

Needs improvement: see Shared-Disk architecture

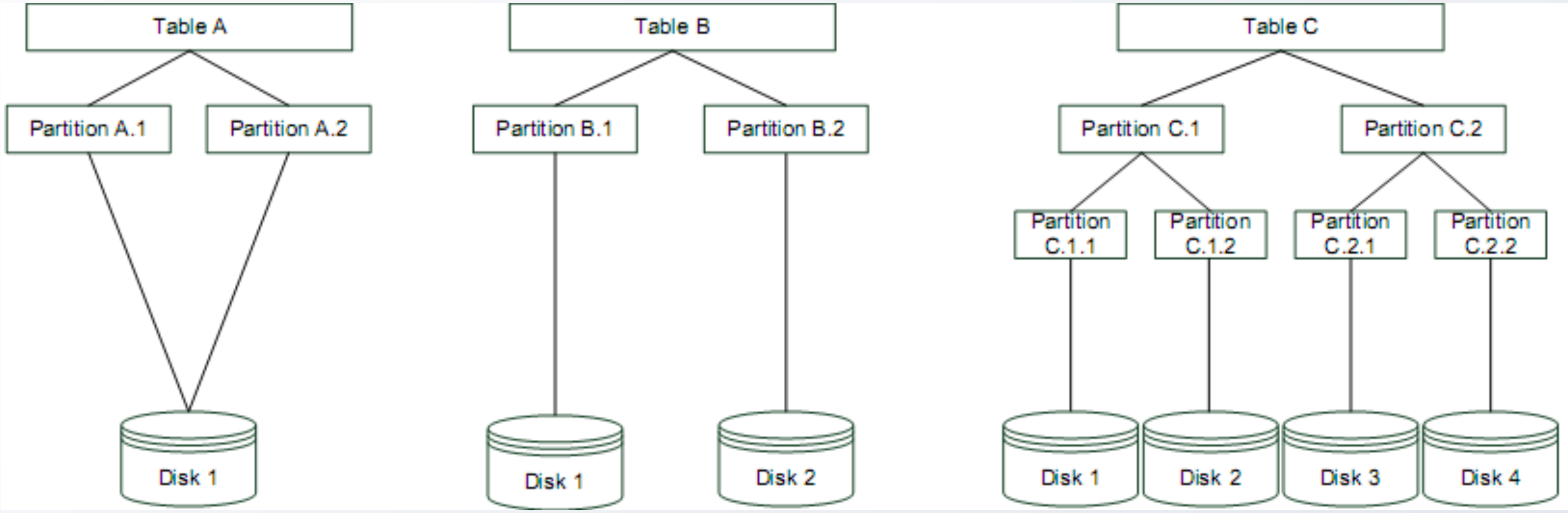
SSDs are a disruptive I/O technology
Performance behavior
Energy consumption
Energy-proport. computing
Benchmarking/ measurements
Architectural concepts of WattDB
WattDB – how to achieve the goal?
Experiments – processing layer
Experiments – symmetric cluster
Comparison: big server vs cluster

Cluster Design Principle – Shared Data

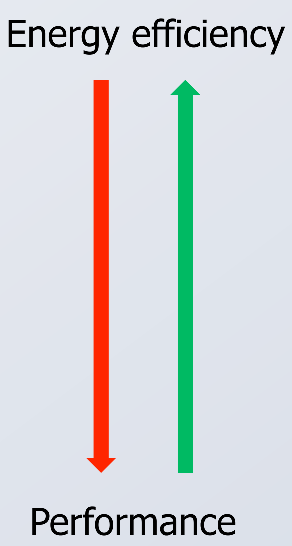


- "Data is fetched to the place of transaction execution"
  - Local transaction execution
  - Same data may be in different states in several DB buffers → buffer coherence problem!
  - Node failure handling: local repair
  - Common concurrency control over logical partitions
  - Scalability of data: no physical repartitioning!
  - Computing power: larger servers needed → scale-up!

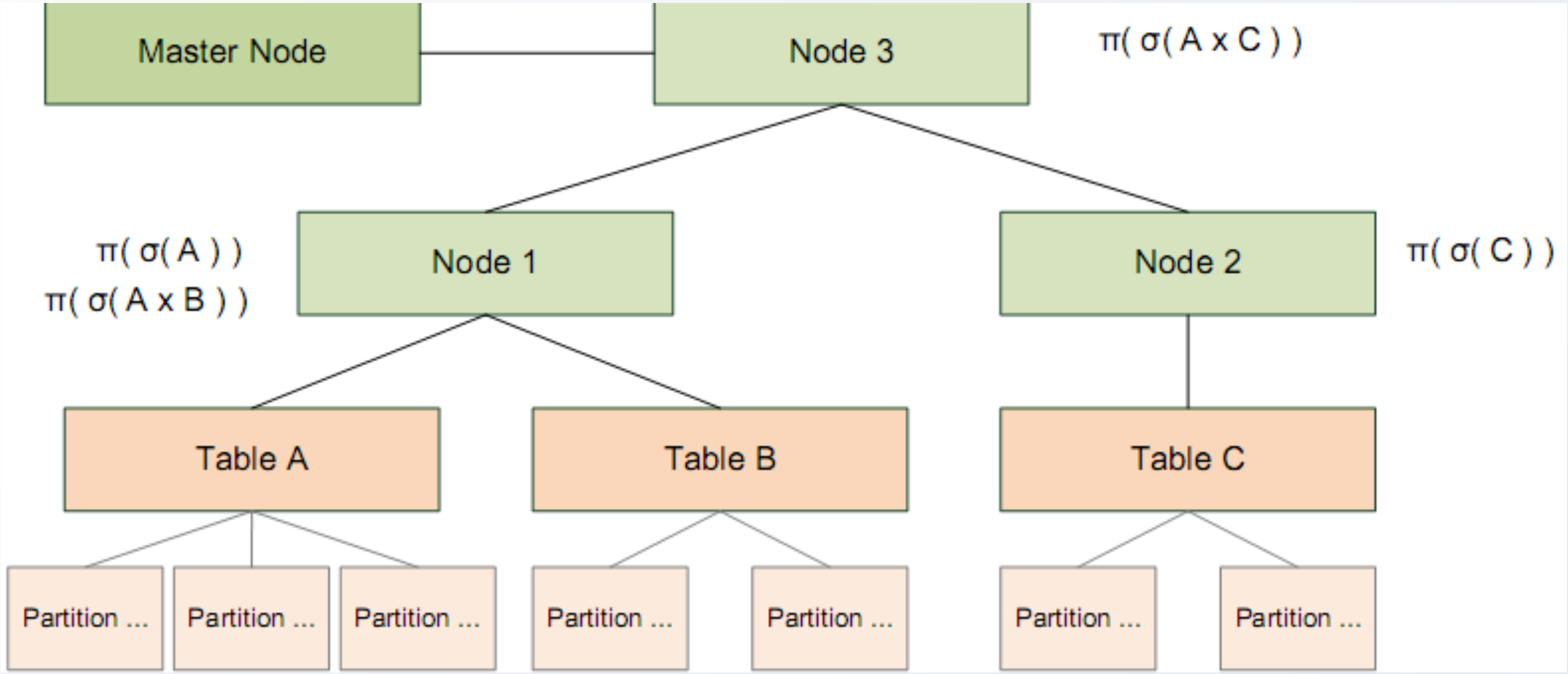
Storage Mapping and Partitioning



- Unused disks should be switched off
- Storage partitioning schemes
  - Table A: 2 partitions on a single disk
  - Table B: 2 partitions on separate disks
  - Table C: 4 partitions on 4 disks
- More flexible partitioning schemes possible
  - C, C.1, or C.1.1 assigned to a single node



Query Processing



- Node assignment for query processing
  - QEP has to reflect data partitioning schemes and their assignment to nodes
  - Subqueries access partitions, process data, and emit intermediate results

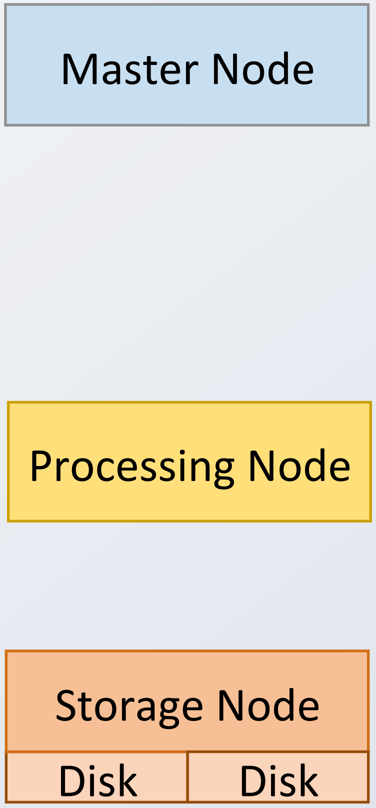
An Energy-proportional DBMS

WattDB

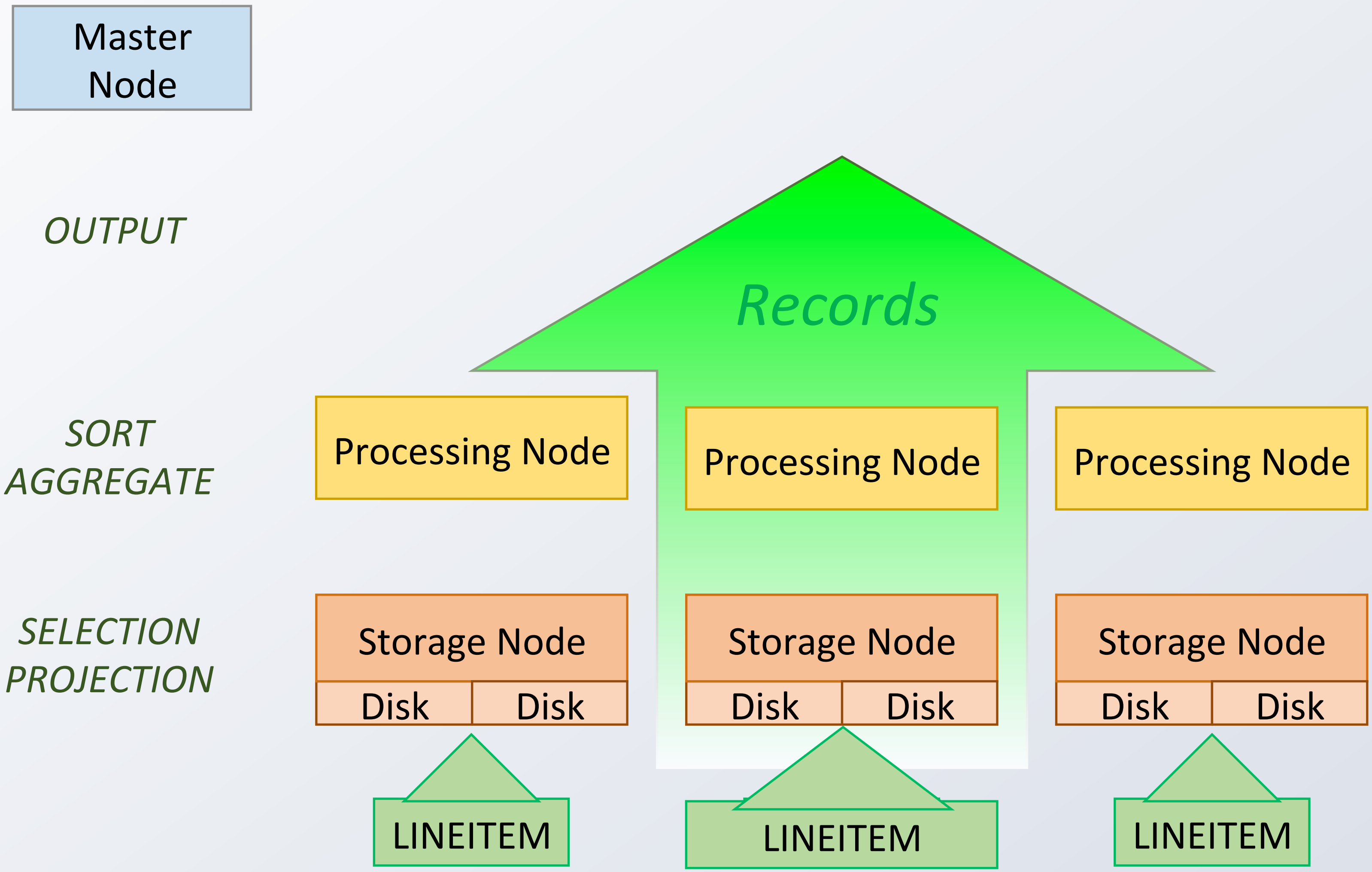
- Energy proportionality as primary objective
- Cluster of lightweight nodes
- Commodity hardware
- Amdahl-balanced

WattDB

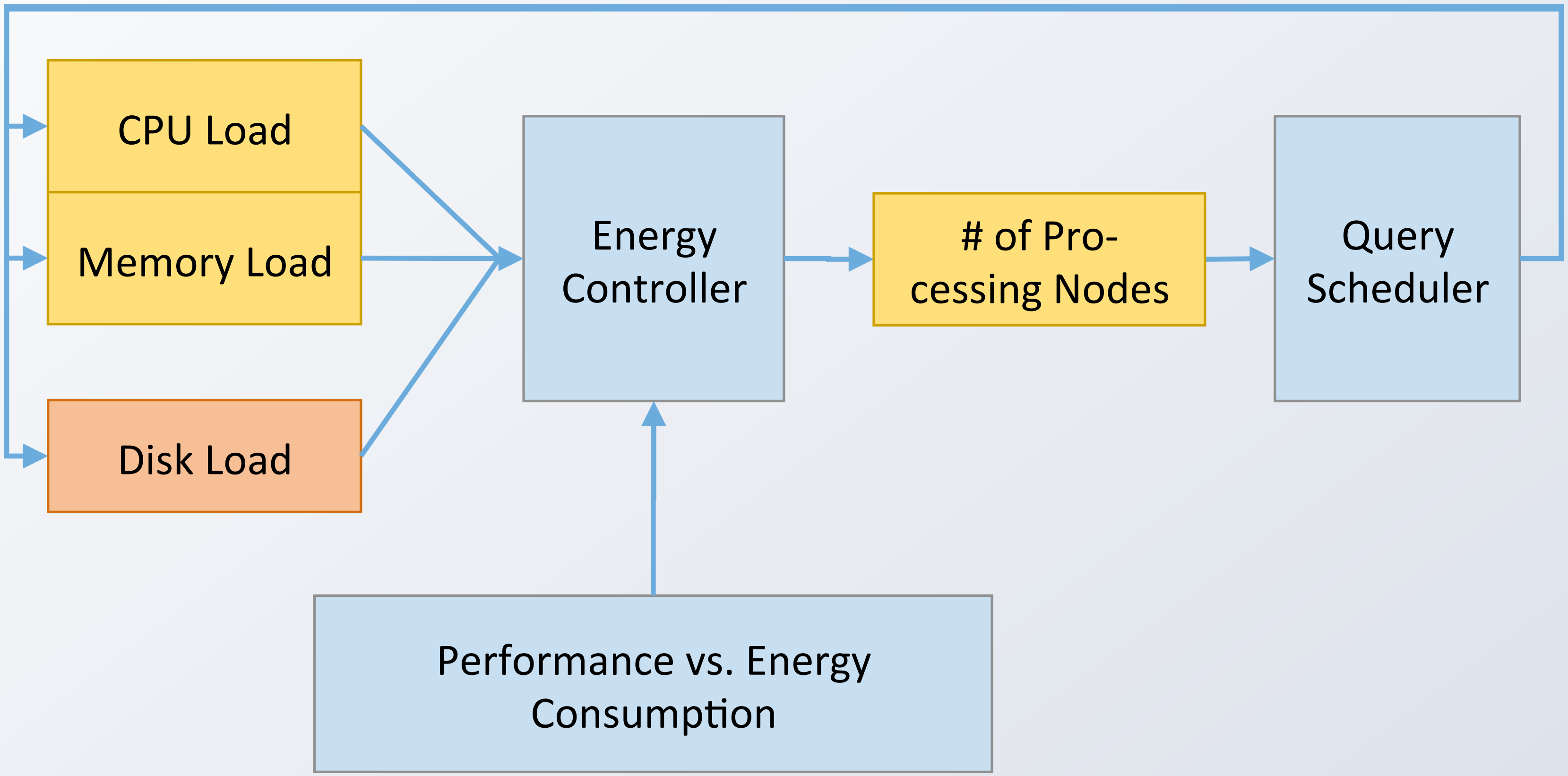
- Master Node responsible for
  - Coordination
  - Provisioning nodes
  - Front-end for client access
- Processing Nodes
  - evaluate queries
- Storage nodes
  - provide data pages
  - manage write-back
- 1 Gbit/s Ethernet



Query Execution



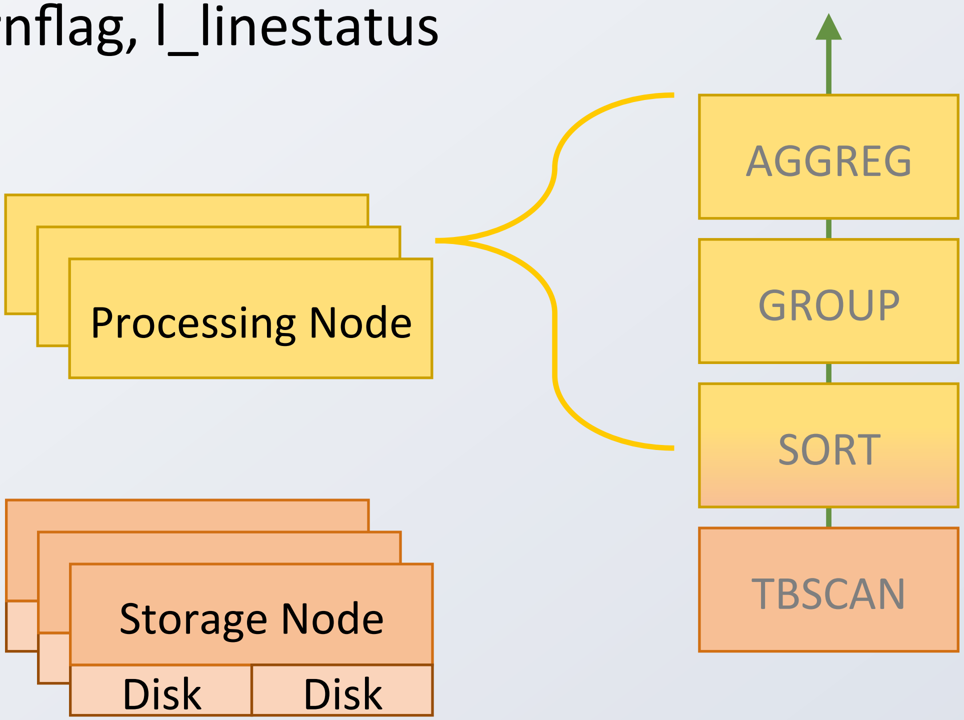
Energy Controller



Queries

- Q1 (of TPC-H)

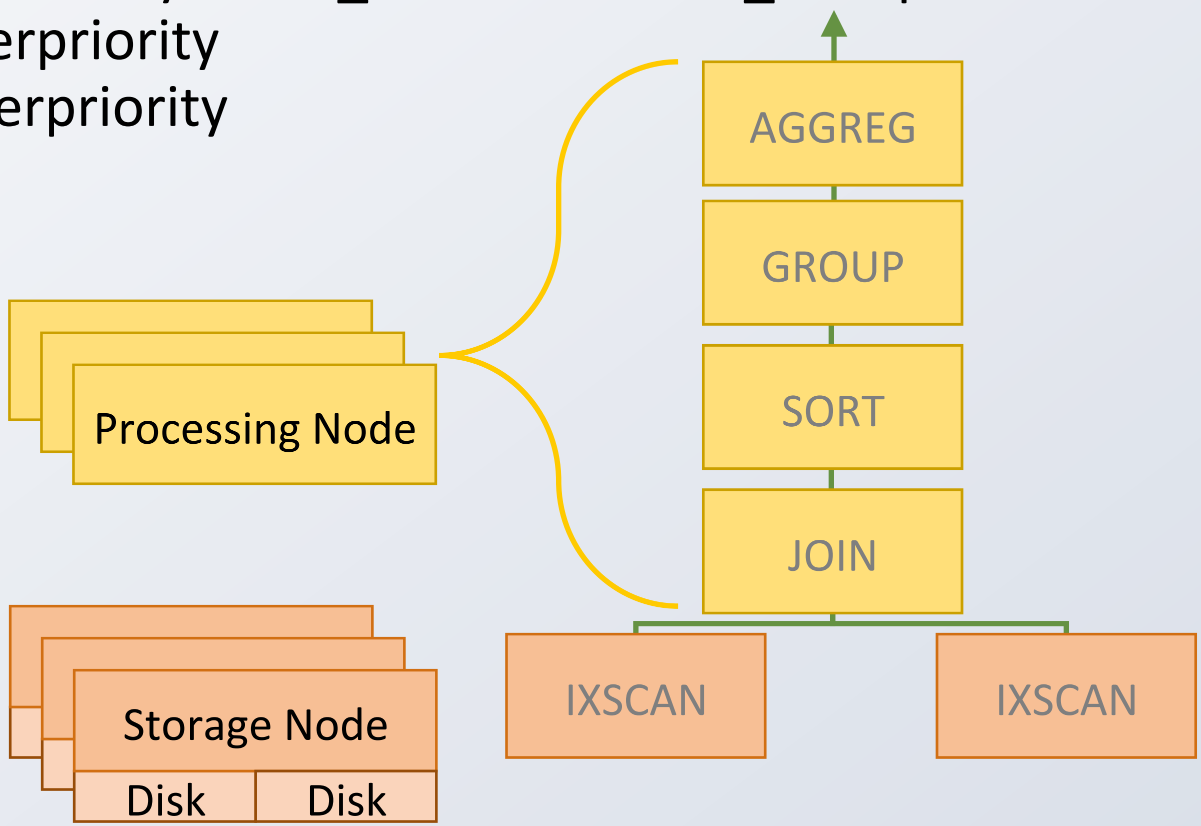
SELECT l_returnflag, l_linestatus, SUM(…), AVG(…), COUNT(*)
FROM lineitem
WHERE l_shipdate <= '1998-12-01' - interval '[DELTA]' day (3)
GROUP BY l_returnflag, l_linestatus
ORDER BY l_returnflag, l_linestatus



Queries (2)

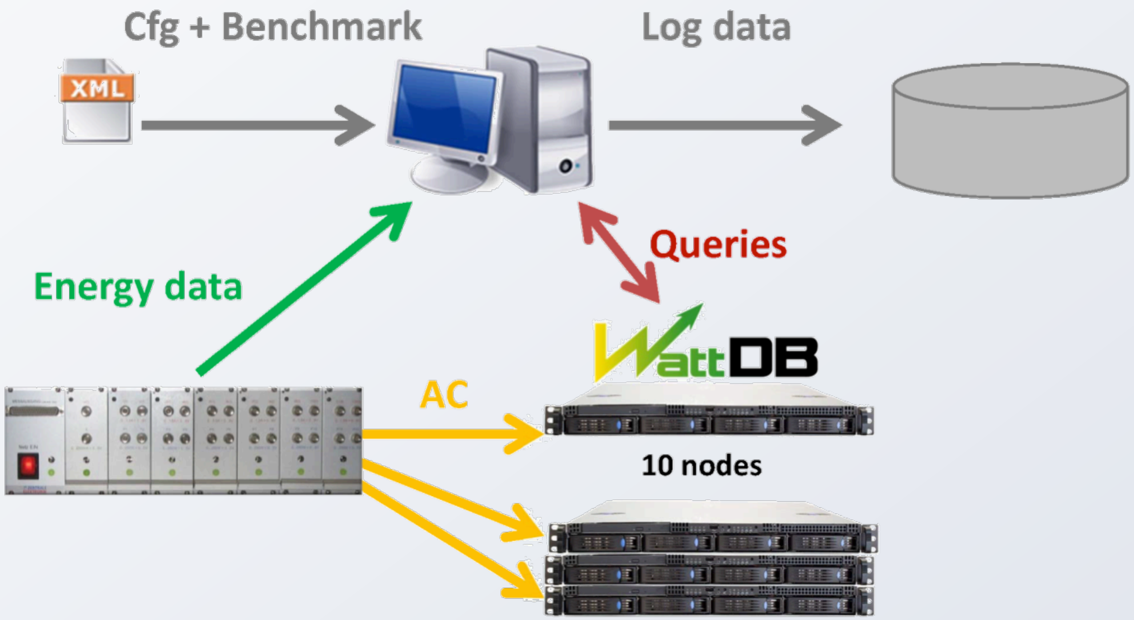
- Q4 (of TPC-H)

SELECT o_orderpriority, COUNT(*)
FROM orders, lineitem
WHERE o_orderdate IN (date '[DATE]' + interval '3' month) AND
l_orderkey = o_orderkey AND l_commitdate < l_receiptdate
ORDER BY o_orderpriority
GROUP BY o_orderpriority



SSDs are a disruptive I/O technology
Performance behavior
Energy consumption
Energy-proport. computing
Benchmarking/ measurements
Architectural concepts of WattDB
WattDB – how to achieve the goal?
Experiments – processing layer
Experiments – symmetric cluster
Comparison: big server vs cluster

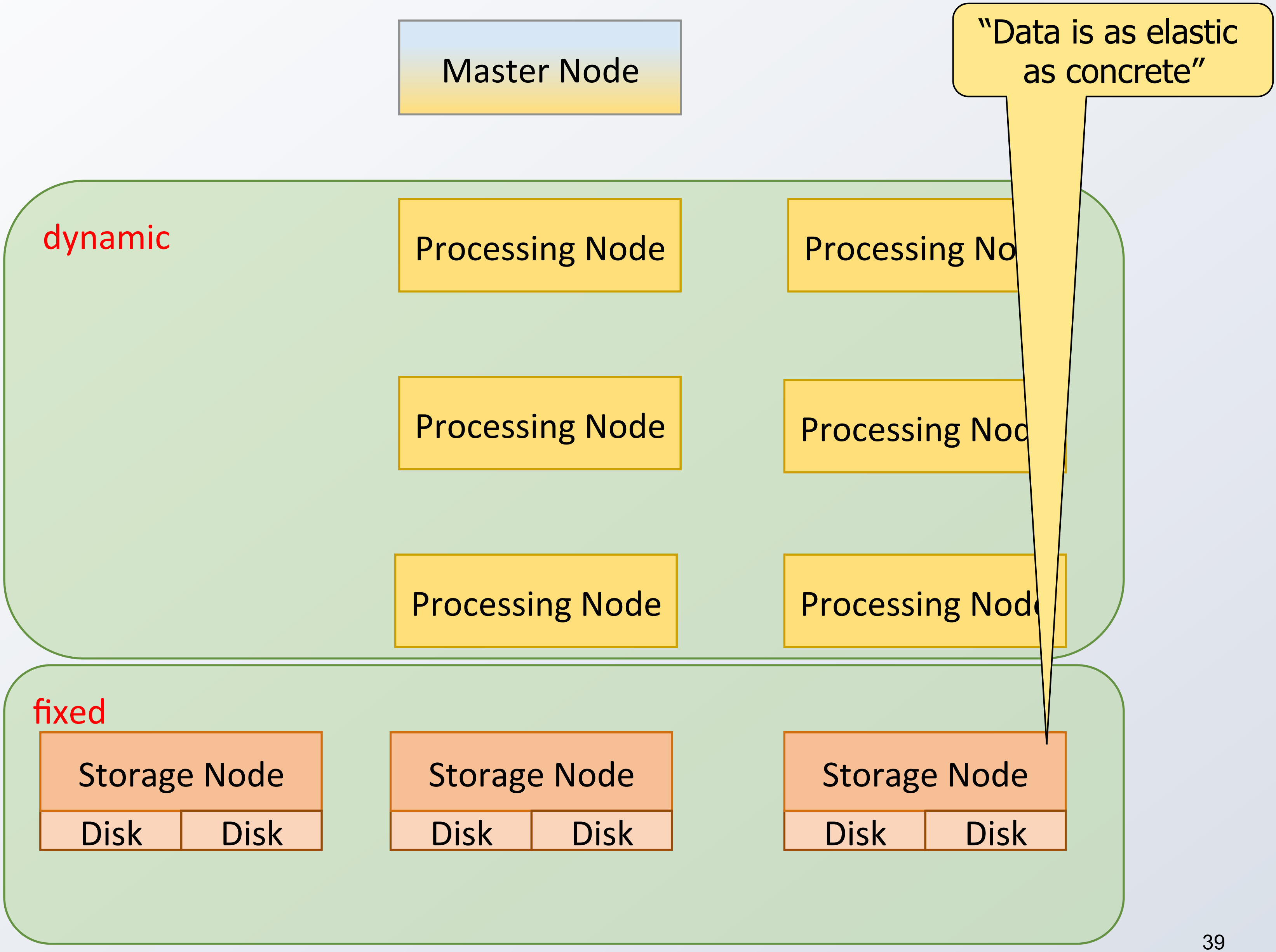
Experimental Set-Up



- Experiments on fixed-size cluster
  - Fixed number of DB clients
  - Fixed number of nodes
  - Measure query throughput & energy consumption
- Experiments on dynamic cluster
  - Varying number of DB clients
  - Dynamic adaptation of the cluster size (processing nodes)
  - Measure query throughput & energy consumption

DBIS Datenbanken und Informationssysteme

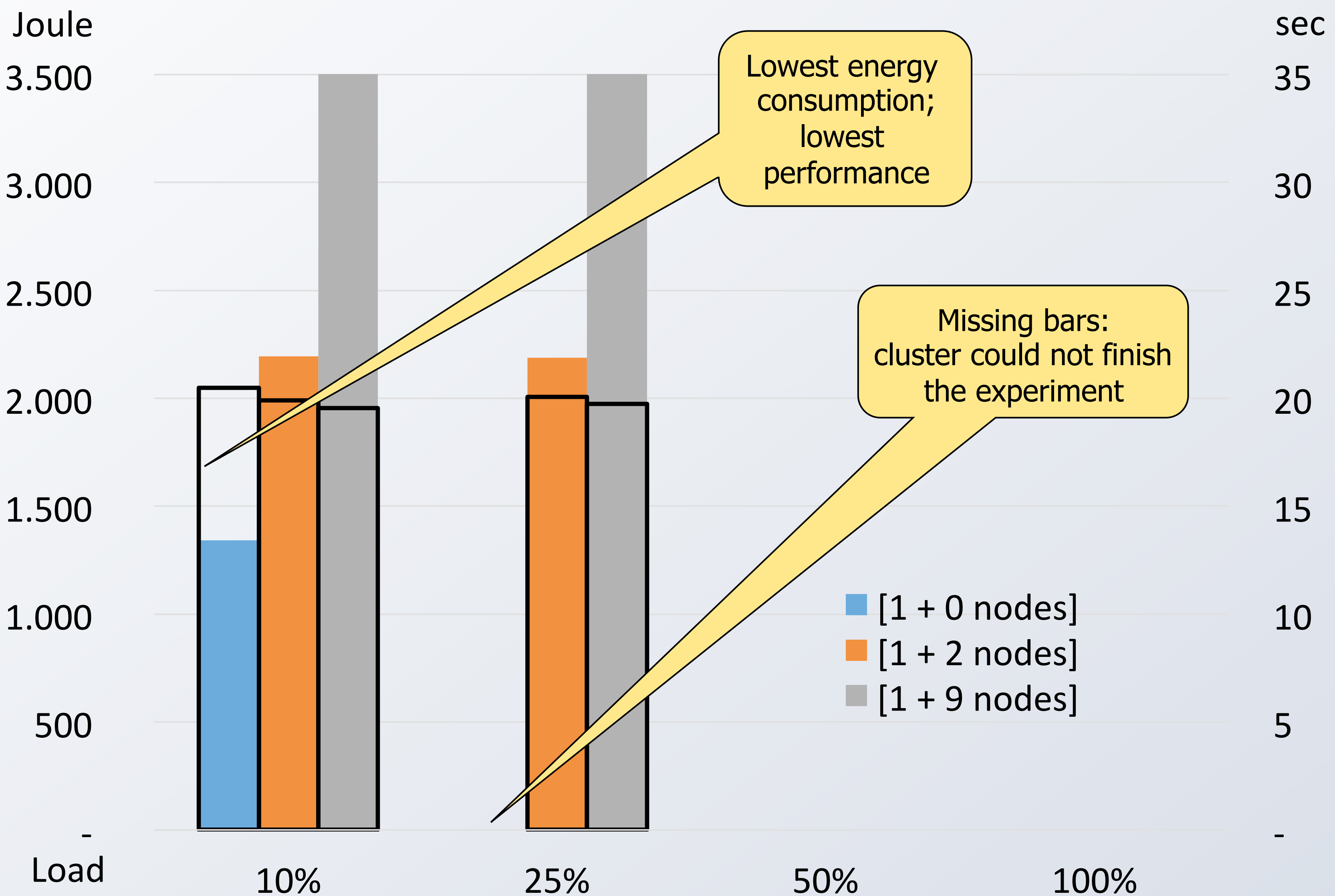
Experimental Results



SSDs are a disruptive I/O technology
Performance behavior
Energy consumption
Energy-proport. computing
Benchmarking/ measurements
Architectural concepts of WattDB
WattDB – how to achieve the goal?
Experiments – processing layer
Experiments – symmetric cluster
Comparison: big server vs cluster

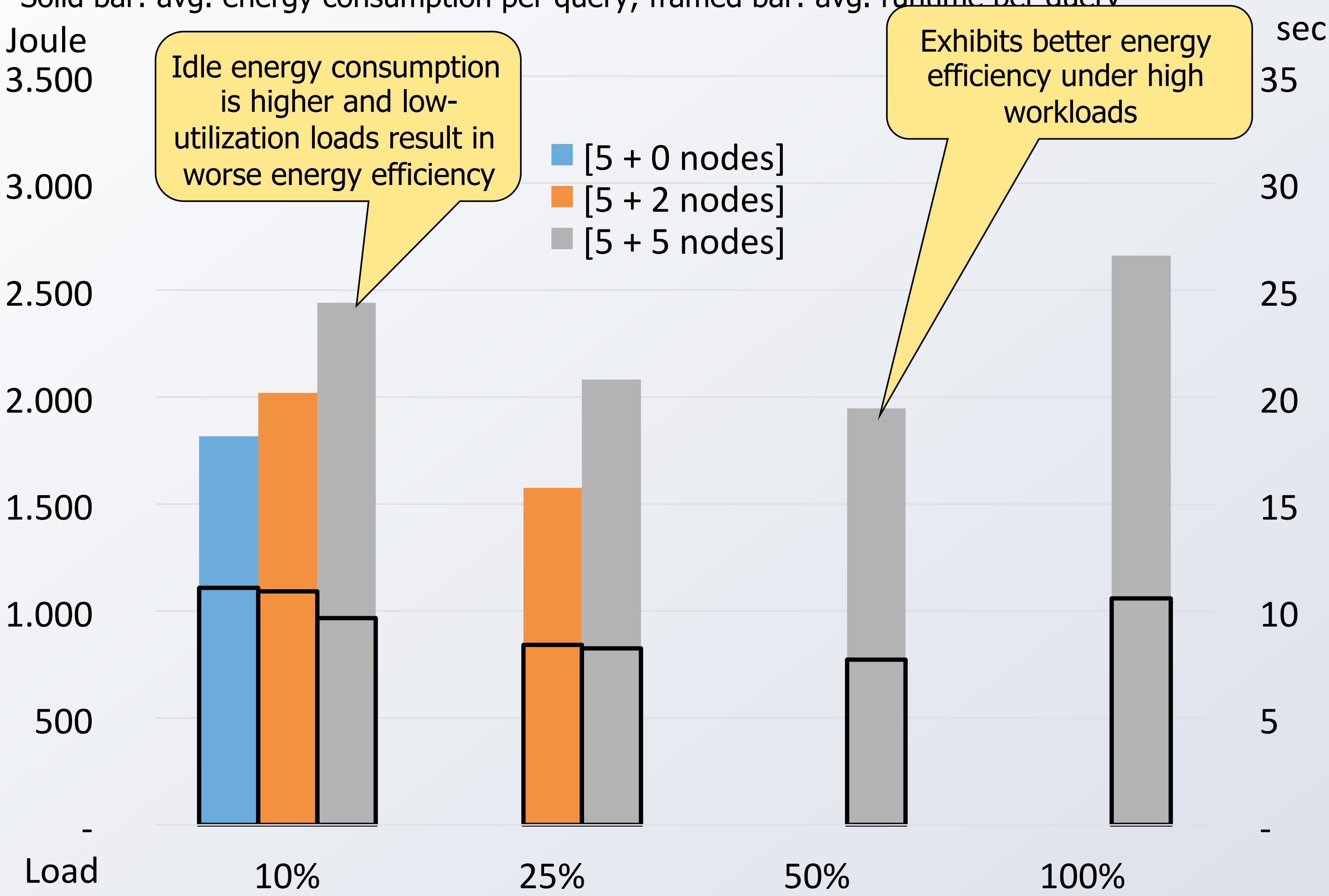
Q1 – 1 Storage Node

Solid bar: avg. energy consumption per query; framed bar: avg. runtime per query



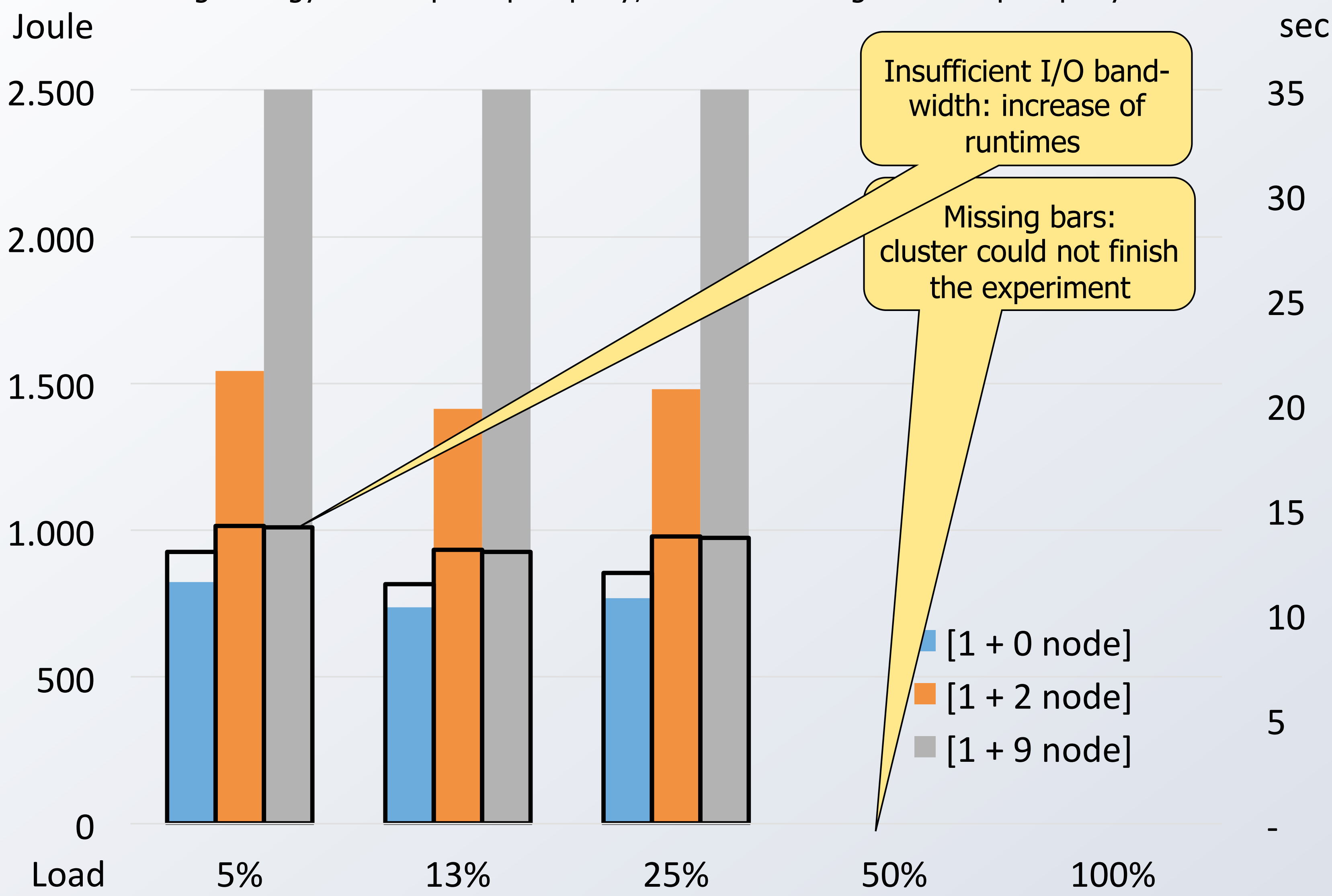
Q1 – 5 Storage Nodes

Solid bar: avg. energy consumption per query; framed bar: avg. runtime per query



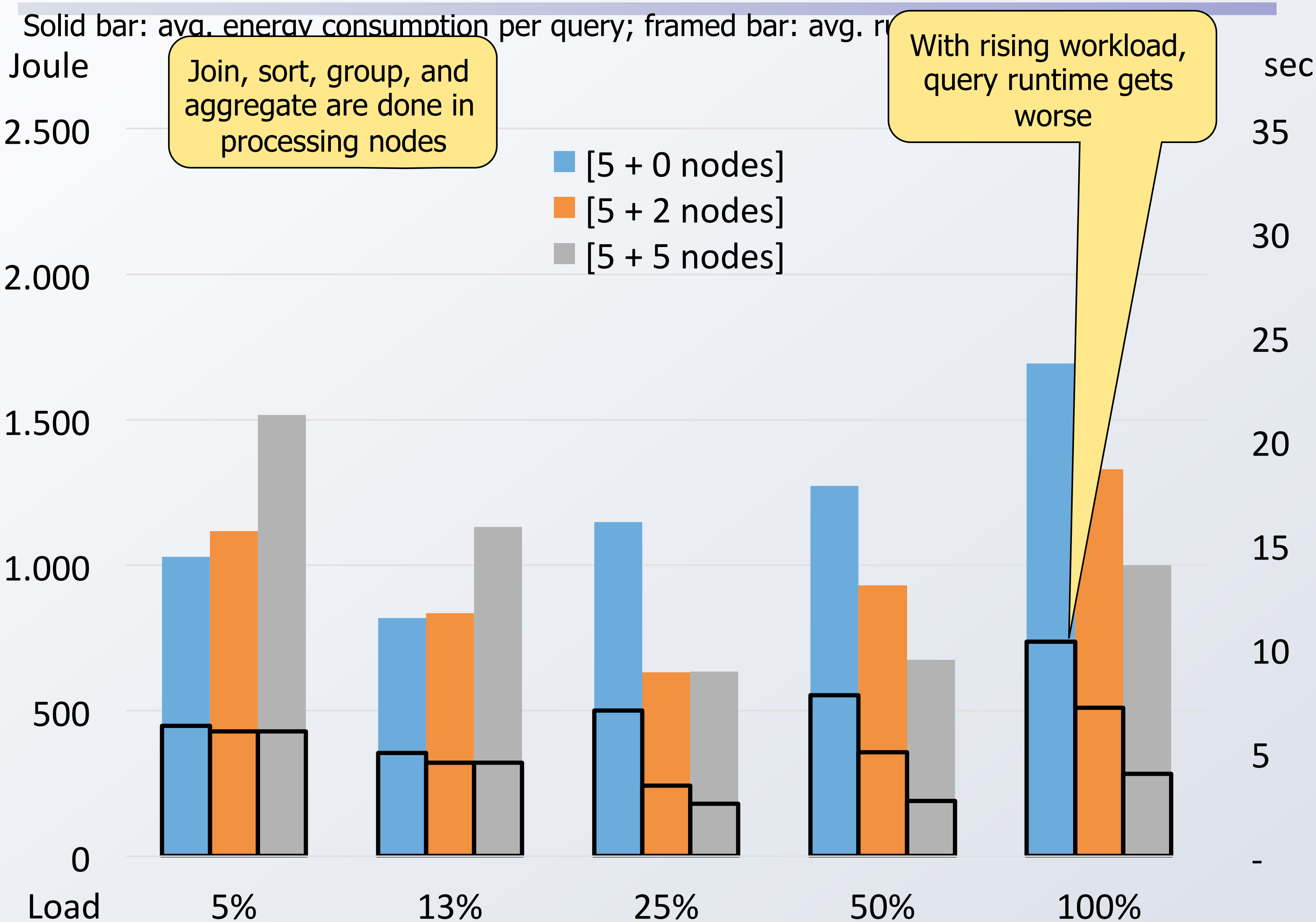
Q4 – 1 Storage Node

Solid bar: avg. energy consumption per query; framed bar: avg. runtime per query



Q4 – 5 Storage Nodes

Solid bar: avg. energy consumption per query; framed bar: avg. r

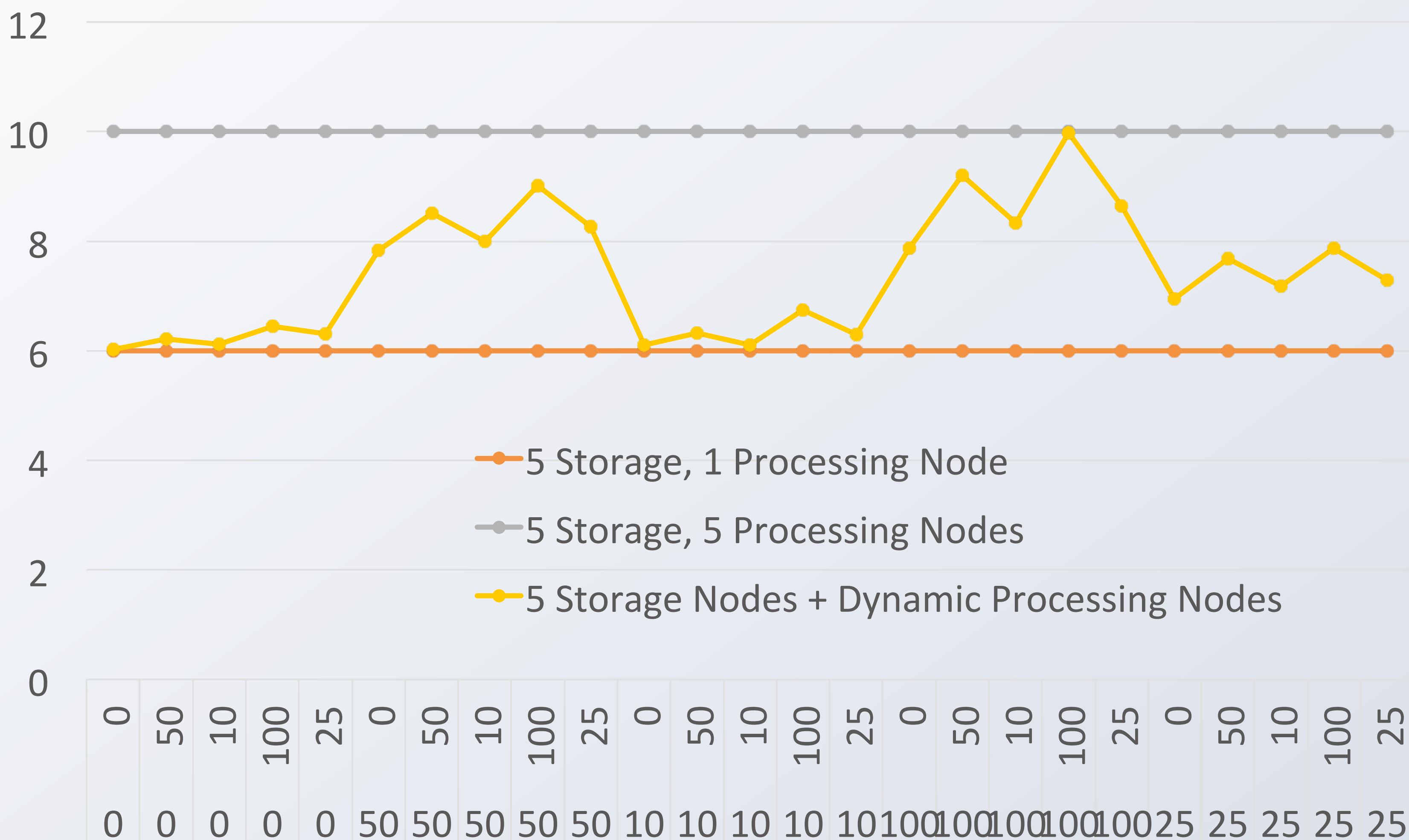


Dynamic Benchmark – Load

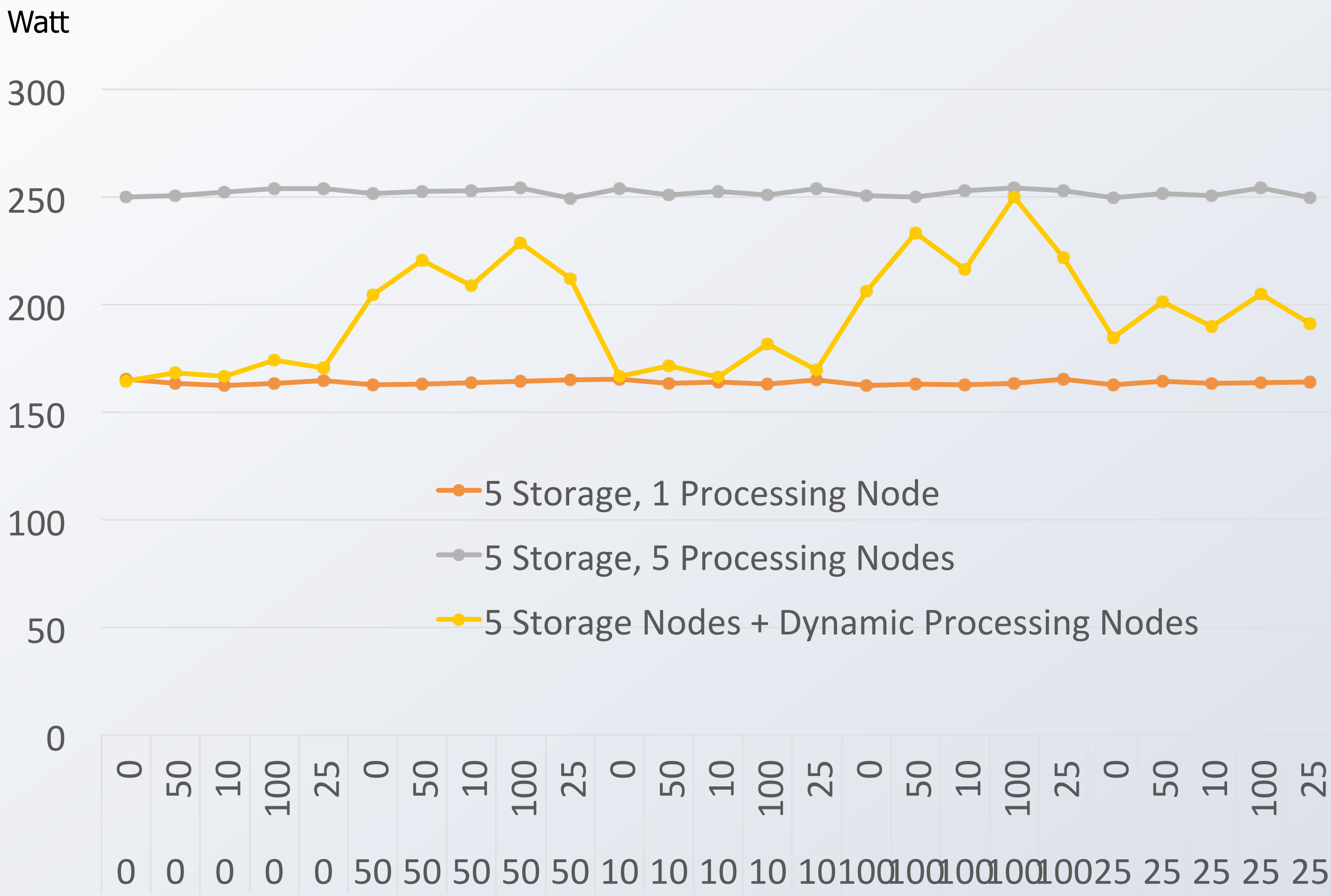
Q1 and Q4 are concurrently scheduled in a dynamically varying workload

Dynamic Benchmark – Nodes

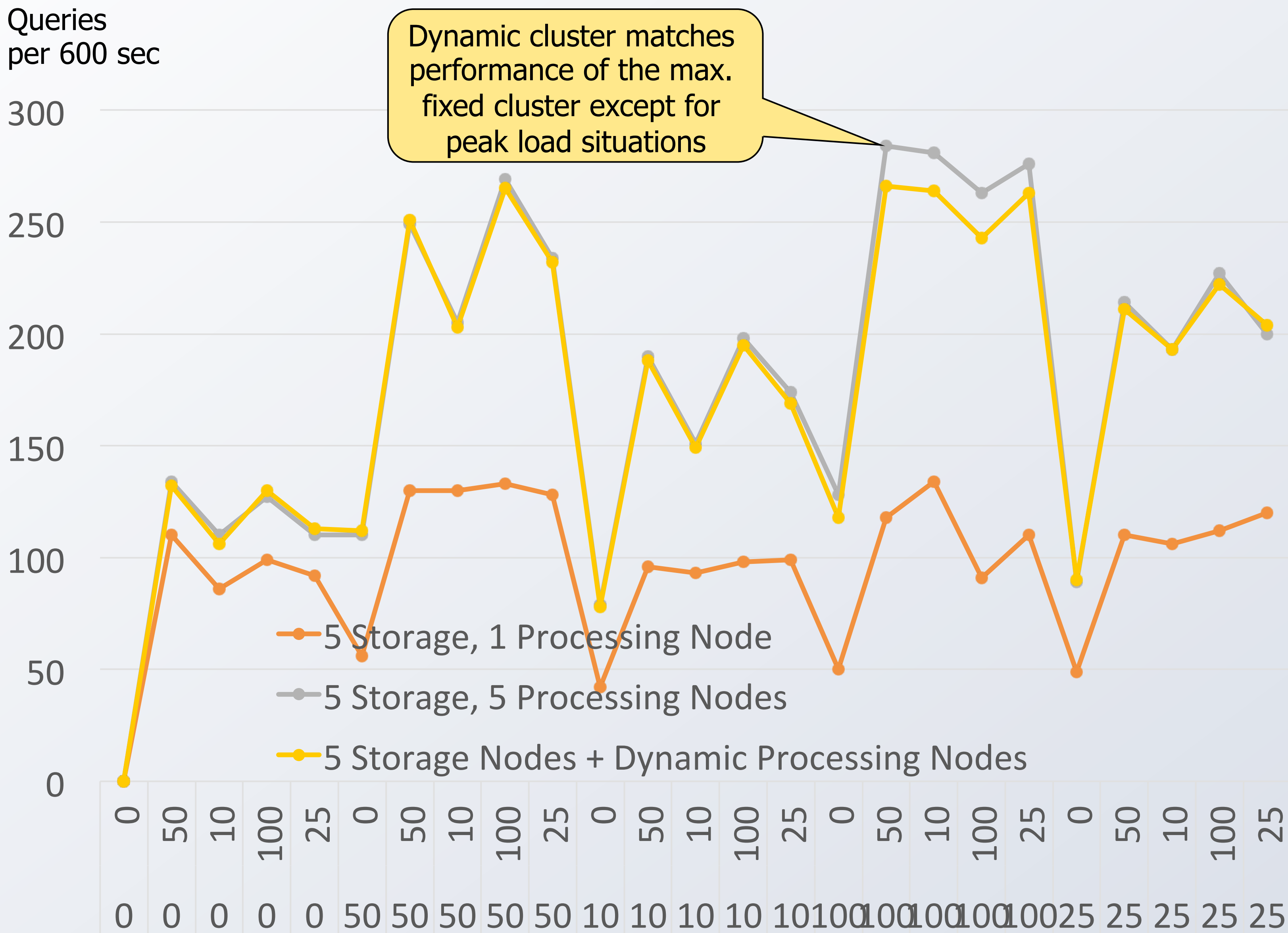
Fixed minimal and maximal clusters together with a dynamic cluster, fixed storage nodes



Dynamic Benchmark – Power Consumption

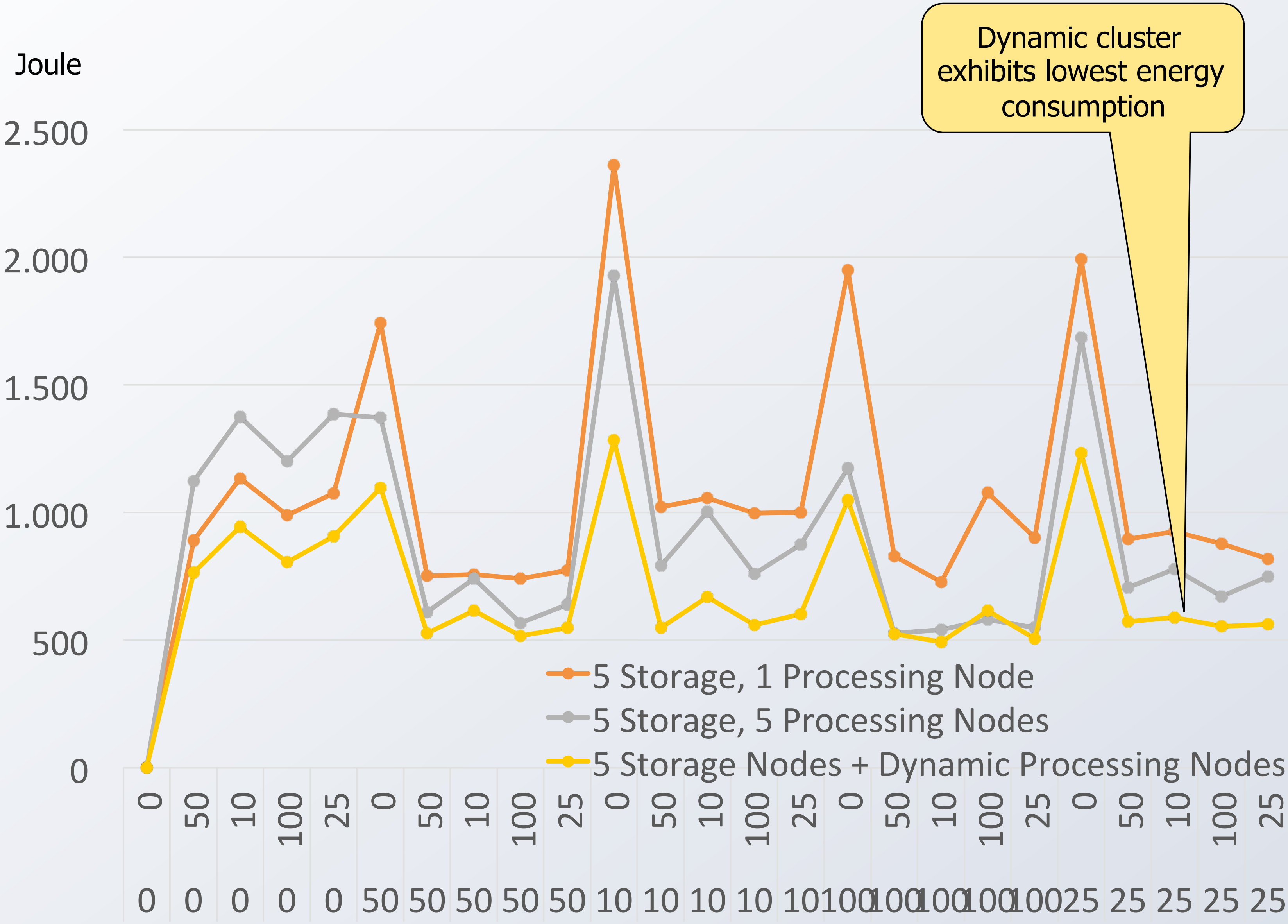


Dynamic Benchmark – Performance



Summer SOC´2014

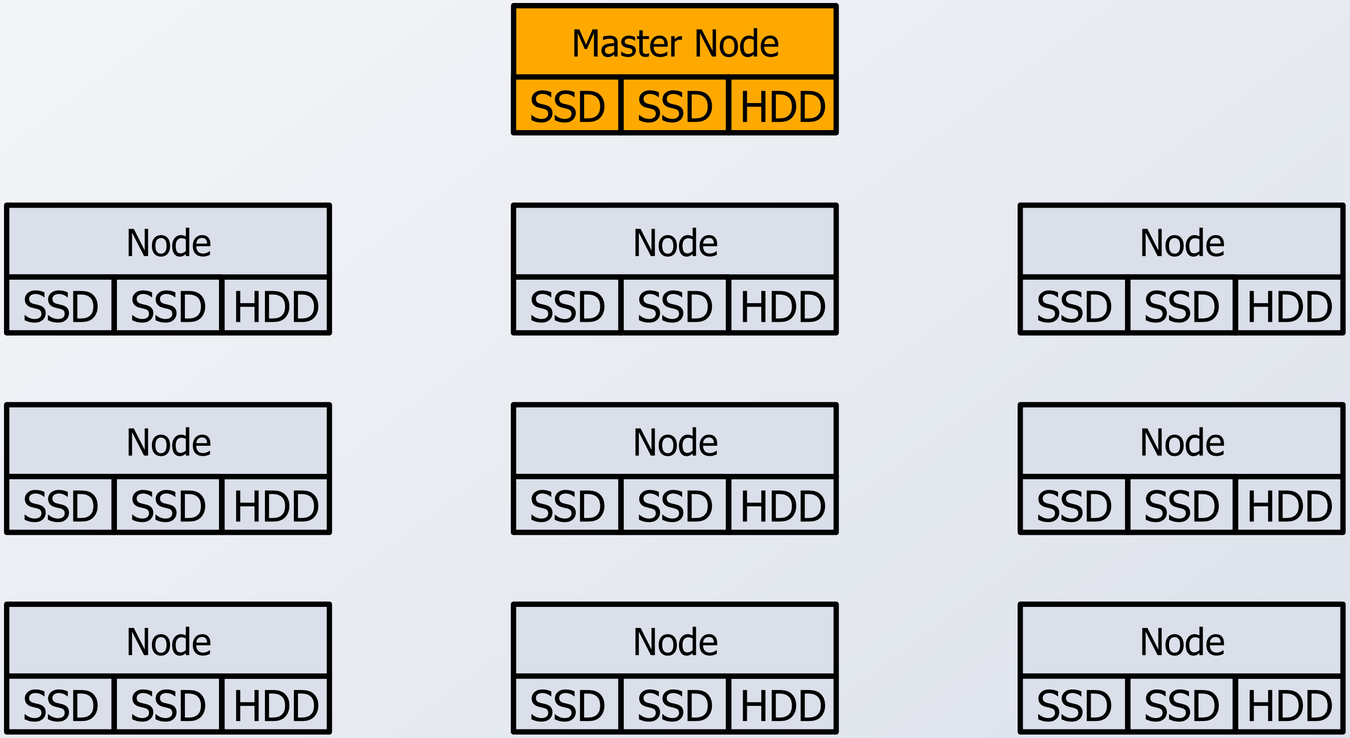
Dynamic Benchmark – Energy Consumption



DBIS Datenbanken und Informationssysteme

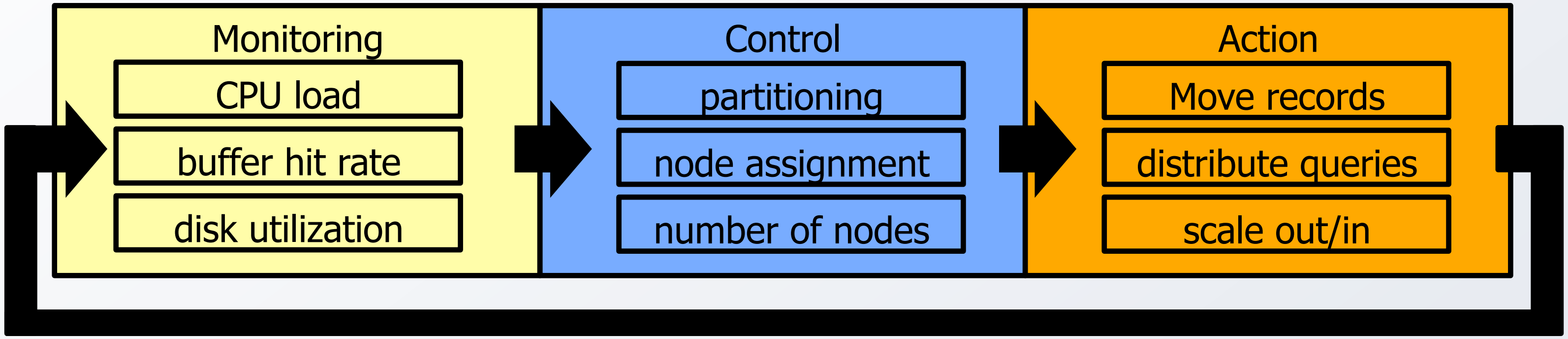
Symmetric Cluster Configuration in WattDB

- All nodes have storage and processing capabilities
  - All nodes can directly communicate with each other
  - Each node provides local operations (scan, selection, projection)
  - Complex operations may be distributed (sort, join, aggregation)
  - Dynamic clusters imply movement/redistribution of data

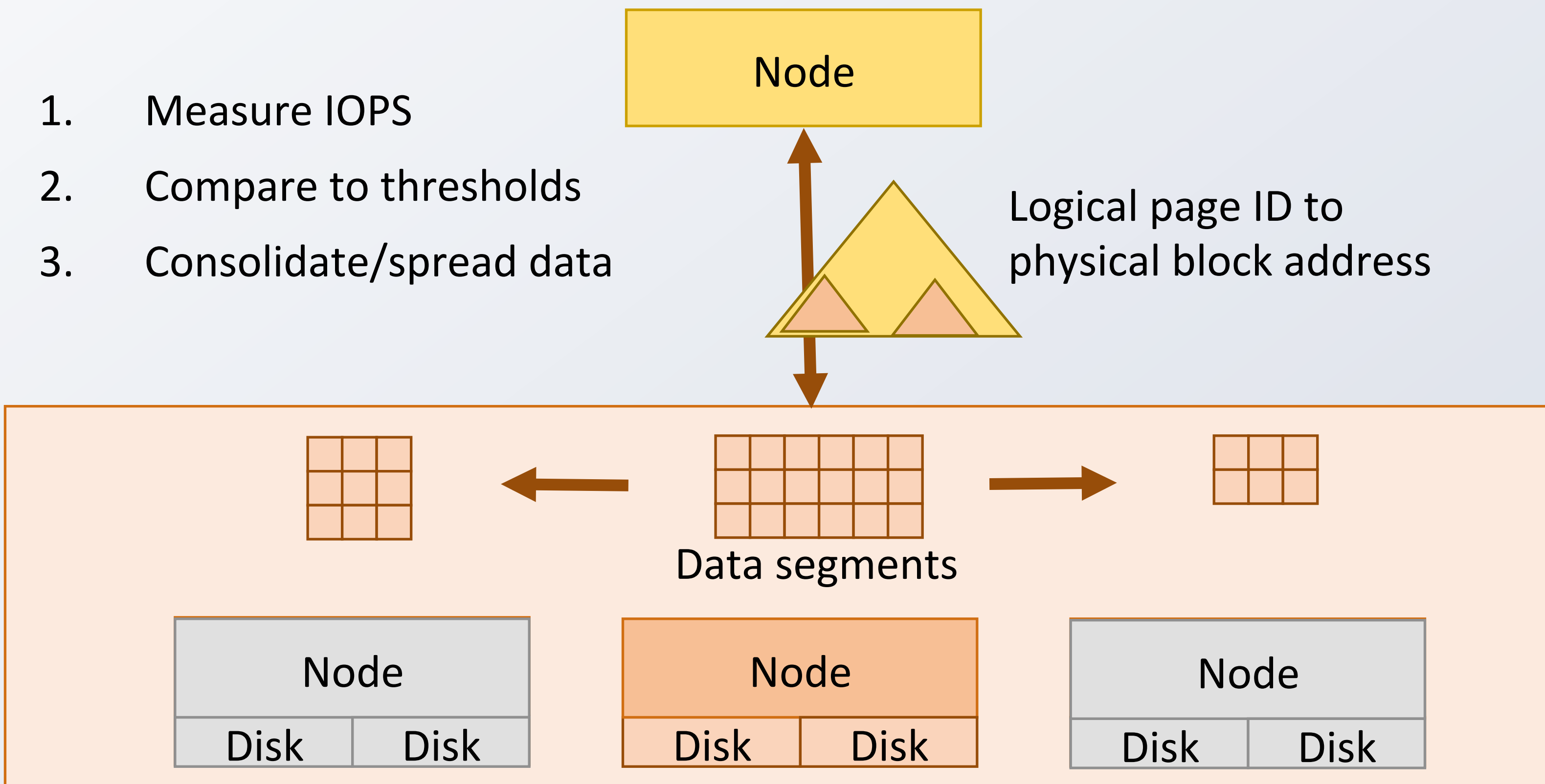


SSDs are a disruptive I/O technology
Performance behavior
Energy consumption
Energy-proport. computing
Benchmarking/ measurements
Architectural concepts of WattDB
WattDB – how to achieve the goal?
Experiments – processing layer
Experiments – symmetric cluster
Comparison: big server vs cluster

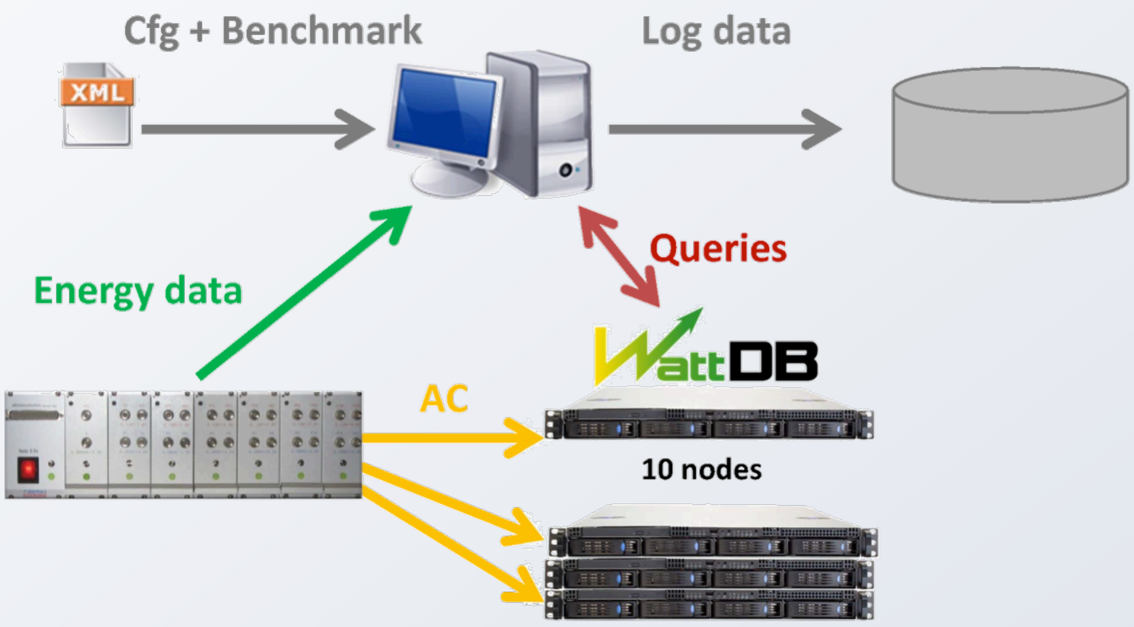
Elastic Storage and Processing



1. Measure IOPS
2. Compare to thresholds
3. Consolidate/spread data



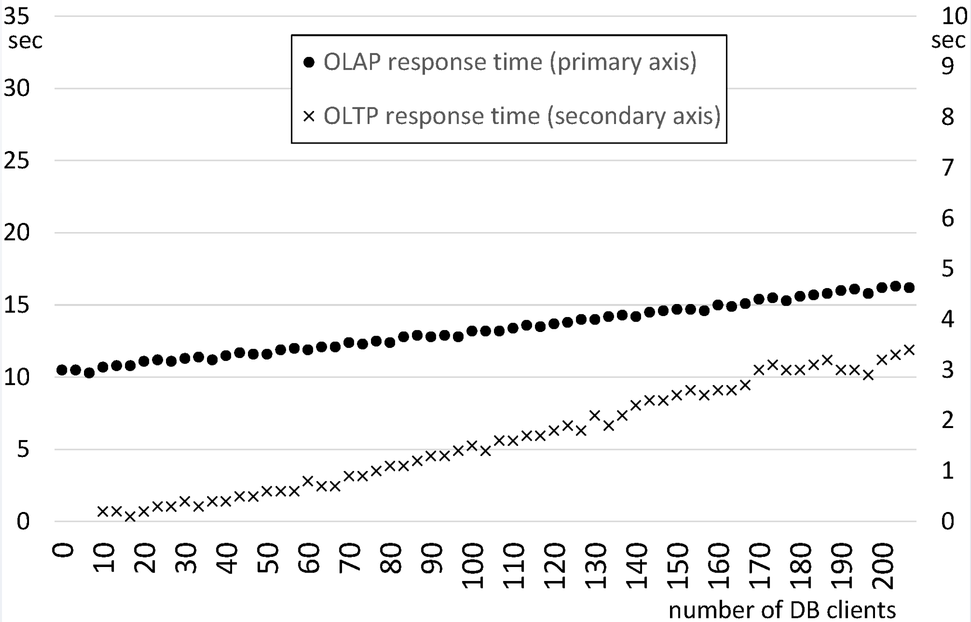
Experimental Set-Up



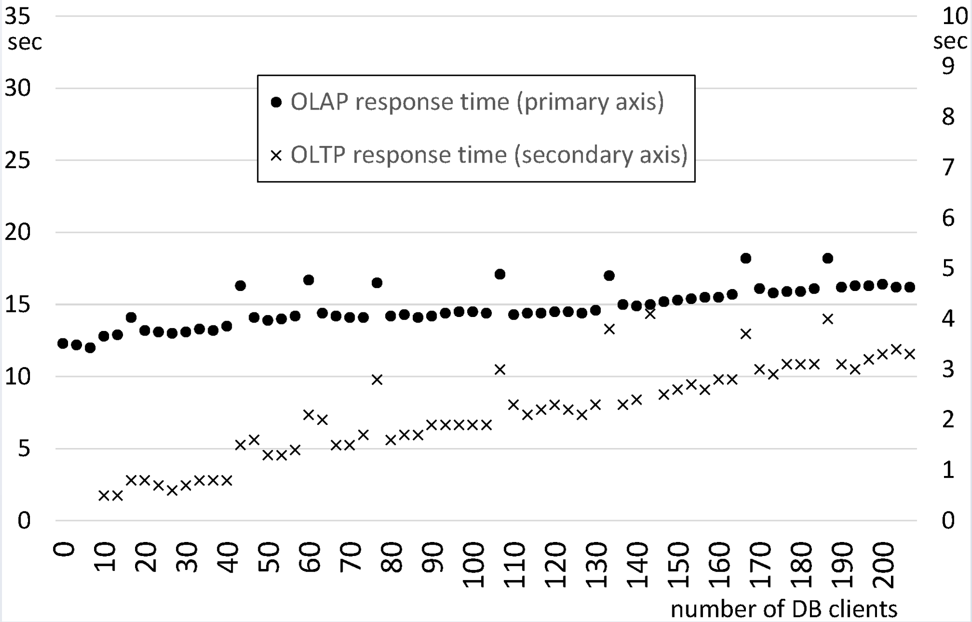
- Varying workloads
  - 1 OLAP query
    (TPC-H, complex read-only query accesses large segments of the data)
  - n DB clients are continuously running OLTP transactions
    (simple read/write queries)
  - 100-GB TPC-H dataset
- Each benchmark
  - consists of 63 workloads of 2 minutes (~2 hours)
  - Dynamic adaptation of the cluster size

Performance Evaluation (1)

Static 10-node cluster, uniform distribution of data, ~10 GB per node

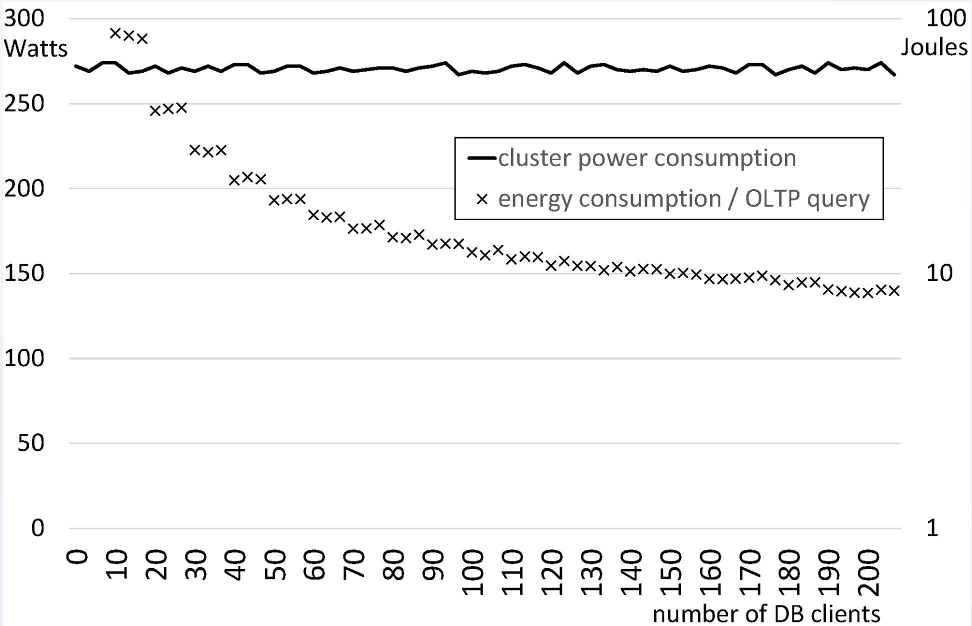


Growth of the cluster (scale-out), dynamic data partitioning/alloc., 100 GB on master node redistributed to ~10 GB per node

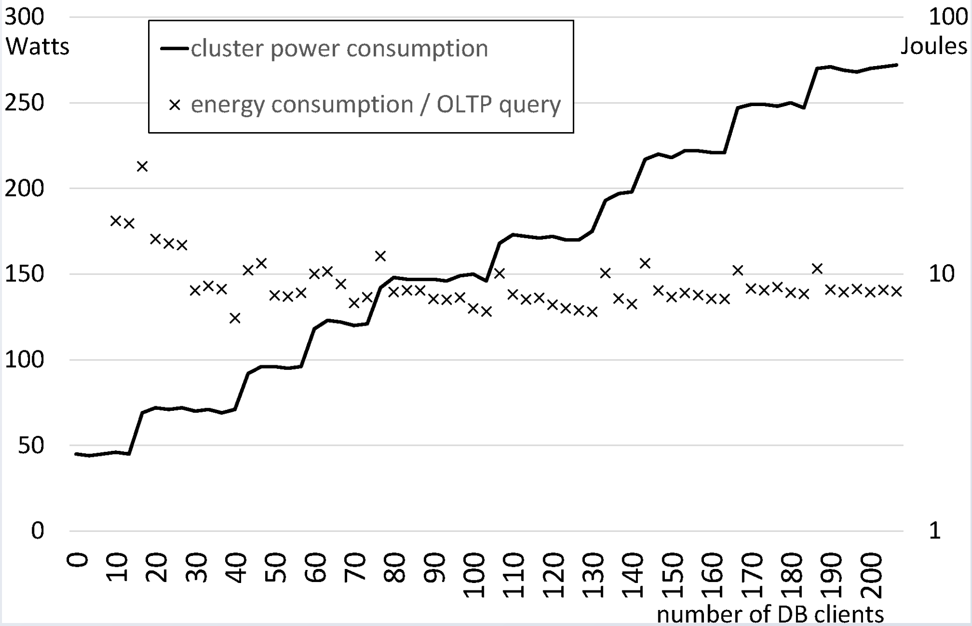


Power/Energy Consumption (1)

Static 10-node cluster, uniform distribution of data, ~10 GB per node

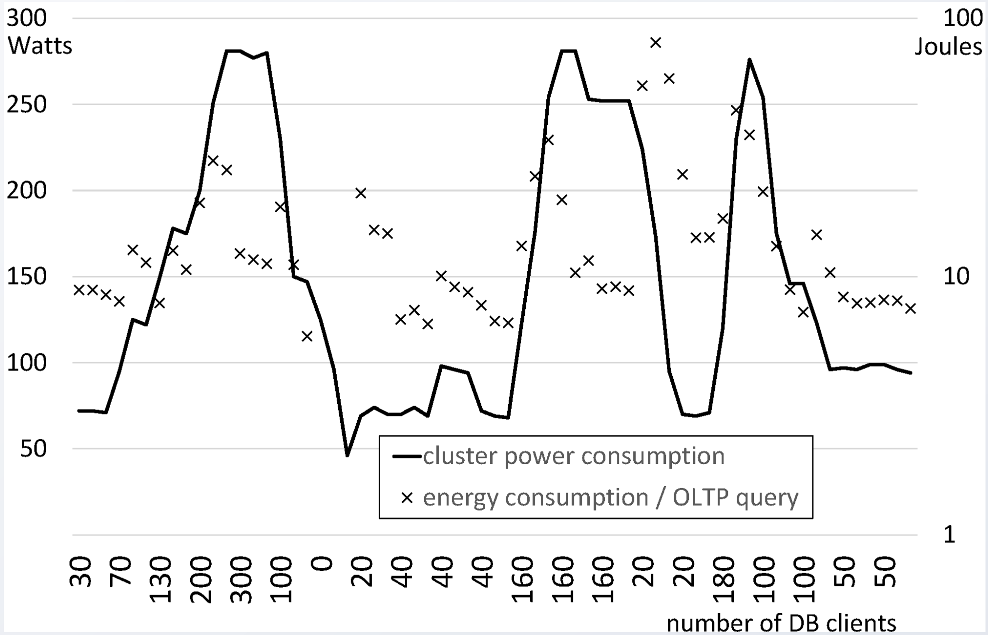


Growth of the cluster (scale-out), dynamic data partitioning/alloc., 100 GB on master node redistri-uted to ~10 GB per node

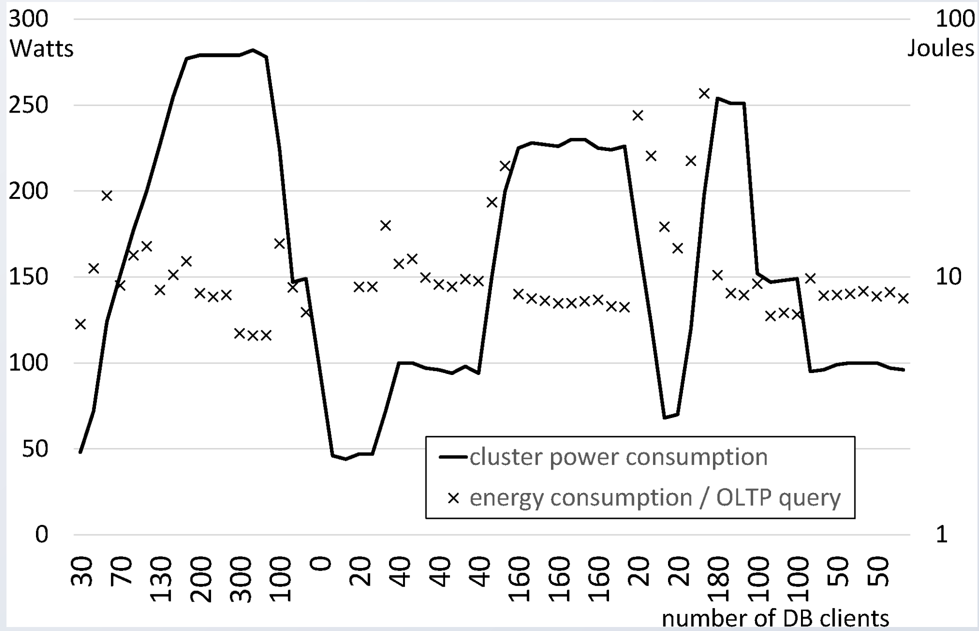


Power/Energy Consumption (2)

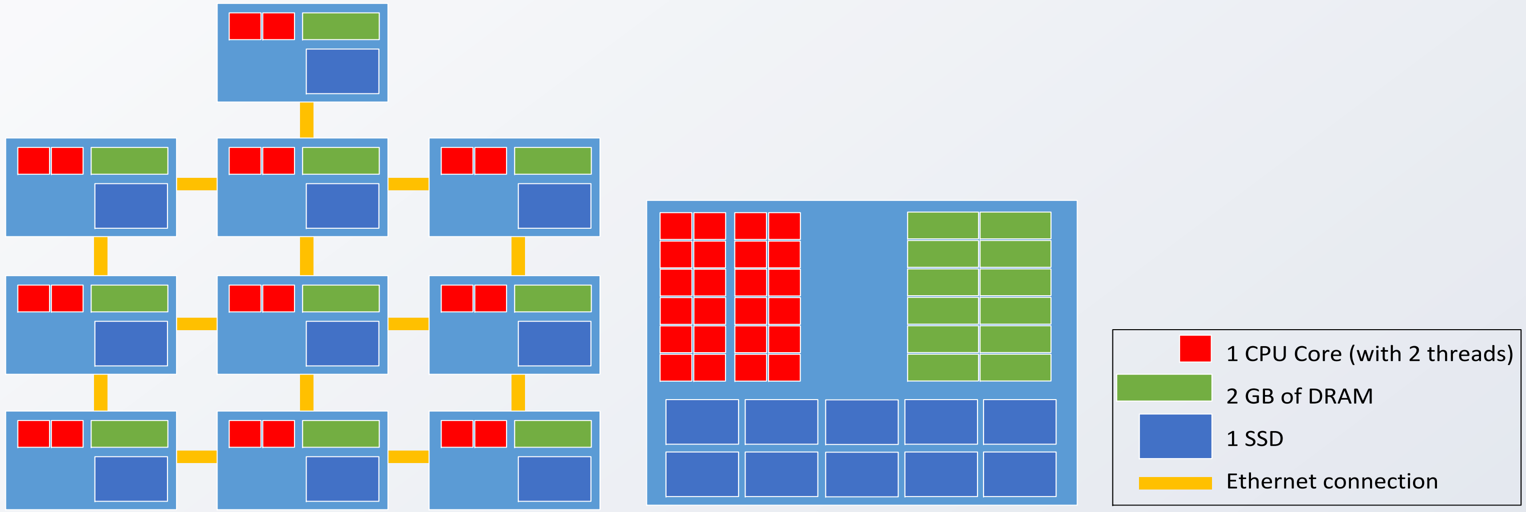
Adaptive behavior of WattDB, drastically growing and shrinking workload by varying the number of OLTP queries, dynamic splitting and merging of data partitions



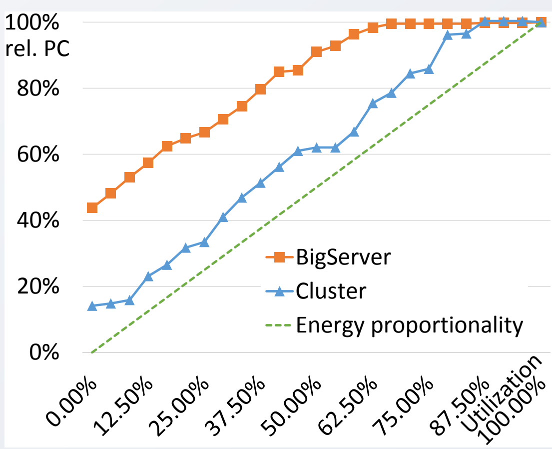
Adaptive behavior supported by simple forecasting (we used knowledge of the "future"), WattDB can pre-configure the cluster for upcoming workloads



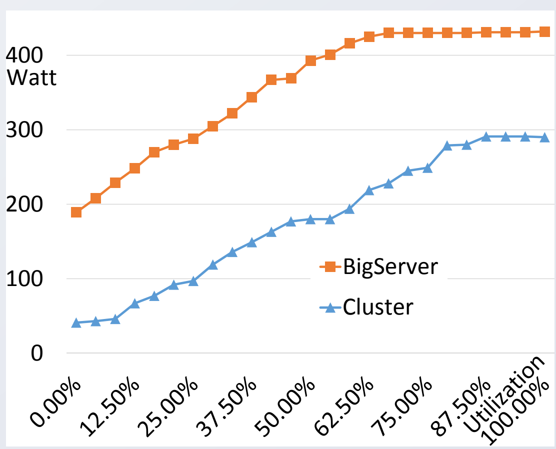
10-Node Cluster vs. Brawny Server



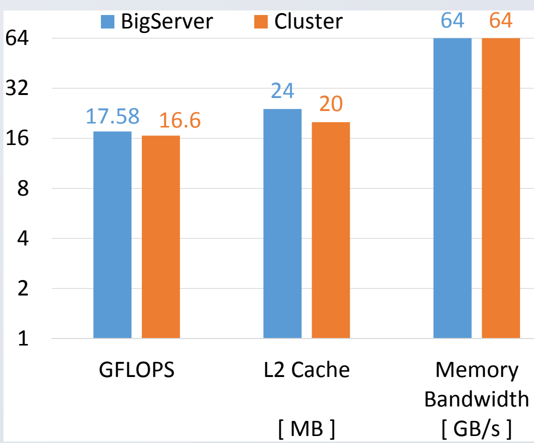
rel. power consumption



power consumption

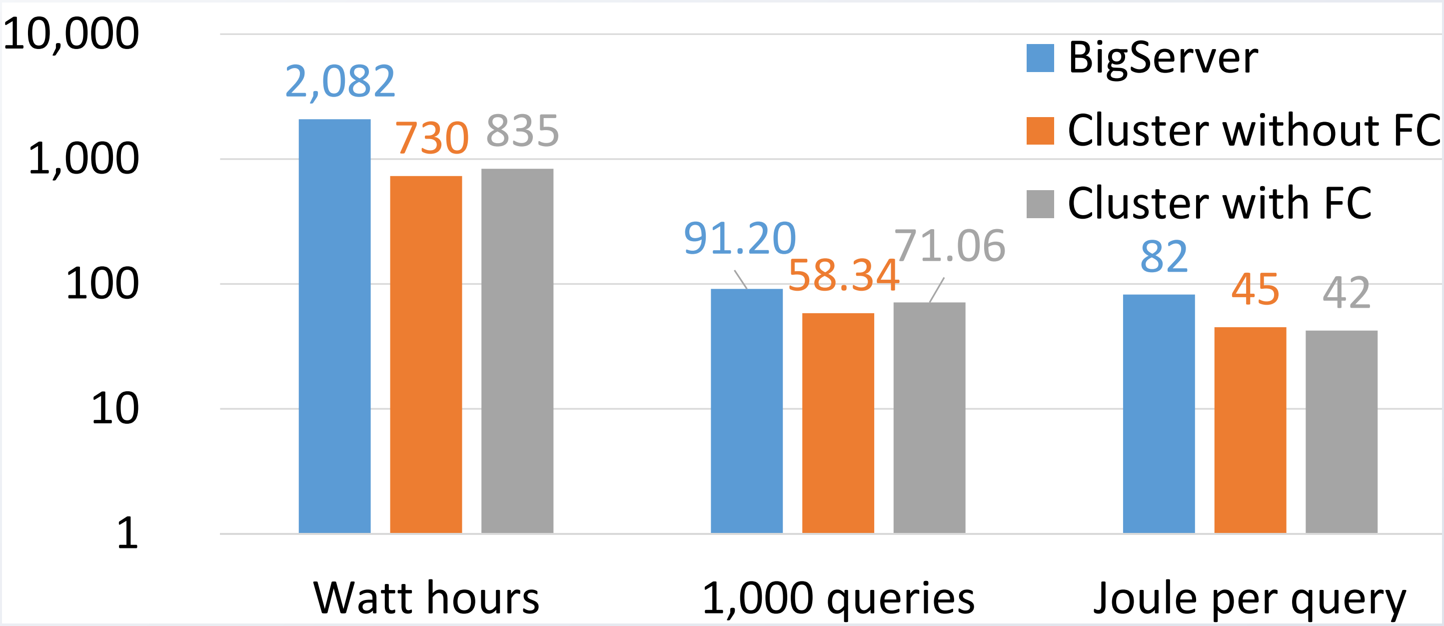


theoretical performance



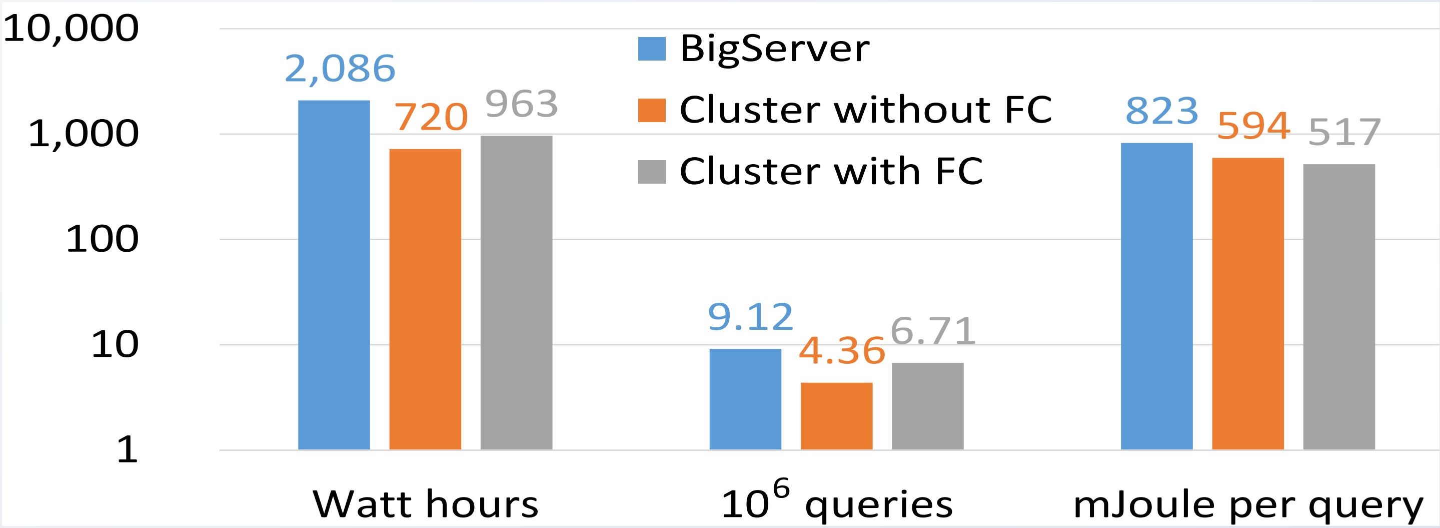
High-Level Comparison: Dynamic OLAP under the Energy-Centric Benchmark

- Total energy consumed
- Overall query throughput in units of 10^3
- Avg. energy consumption in Joule per query



High-Level Comparison: Dynamic OLTP under the Energy-Centric Benchmark

- Total energy consumed
- Overall query throughput in units of 10^6
- Avg. energy consumption in mJoule per query



Conclusions

- Every SSD behaves differently and shows unstable write behavior
- Use of SSDs: improve predictability and performance
- Key observation: The most energy-efficient configuration is typically the highest performing one within a single node intended for use in scale-out architectures (under 100% system utilization)
- WattDB achieves energy proportionality:
  cluster of wimpy nodes and not a single powerful DB server
- Architecture of WattDB combines the advantages of SN and SD
- Processing nodes can be reintegrated in a few seconds
  Storage repartitioning needs seconds to minutes
- Performance: cluster is no match for the big server
- Low and moderate workloads: cluster is competitive, server is still faster, but uses more than twice of the cluster's energy

DBIS Datenbanken und Informationssysteme