

TOWARDS ENERGY- EFFICIENT DATABASE CLUSTER DESIGN

Presented by:
Vinay Nagar
CS525 SPI3

Willis Lang University of Wisconsin wlang@cs.wisc.edu

Stavros Harizopoulos Nou Data stavros@noudata.com

Jignesh M. Patel University of Wisconsin
jignesh@cs.wisc.edu

Mehul A. Shah Nou Data mehul@noudata.com

Dimitris Tsirogiannis Microsoft Corp.
dimitsir@microsoft.com



MAIN IDEA

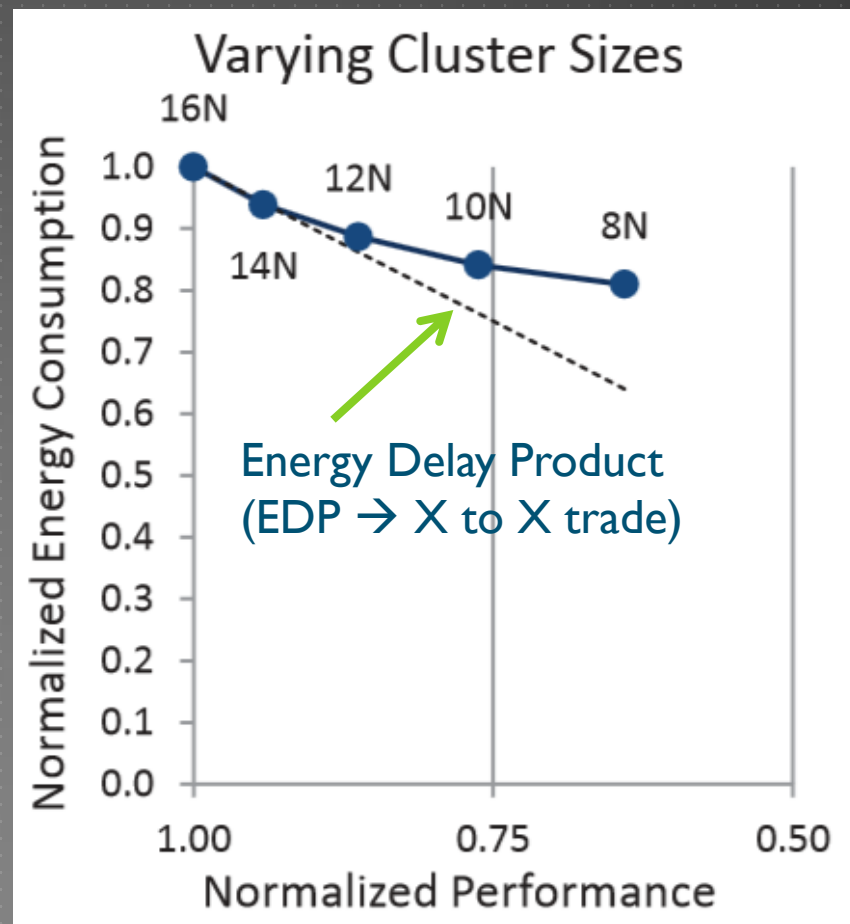
- ▶ **Study the trade-offs** between performance and energy consumption in a cluster to identify bottlenecks and **propose a model** that considers these bottlenecks and other query parameters to **predict the performance and energy consumption** of the cluster. With the findings provide **cluster design principles**.

INTRODUCTION

- ▶ Energy growing cost of operational cost
- ▶ **CHALLENGES** to increasing energy efficiency
 - ▶ Inherent scaling inefficiency
 - ▶ Choosing energy-efficient hardware
- ▶ Architectural design space of energy efficient database clusters

VARYING CLUSTER SIZES

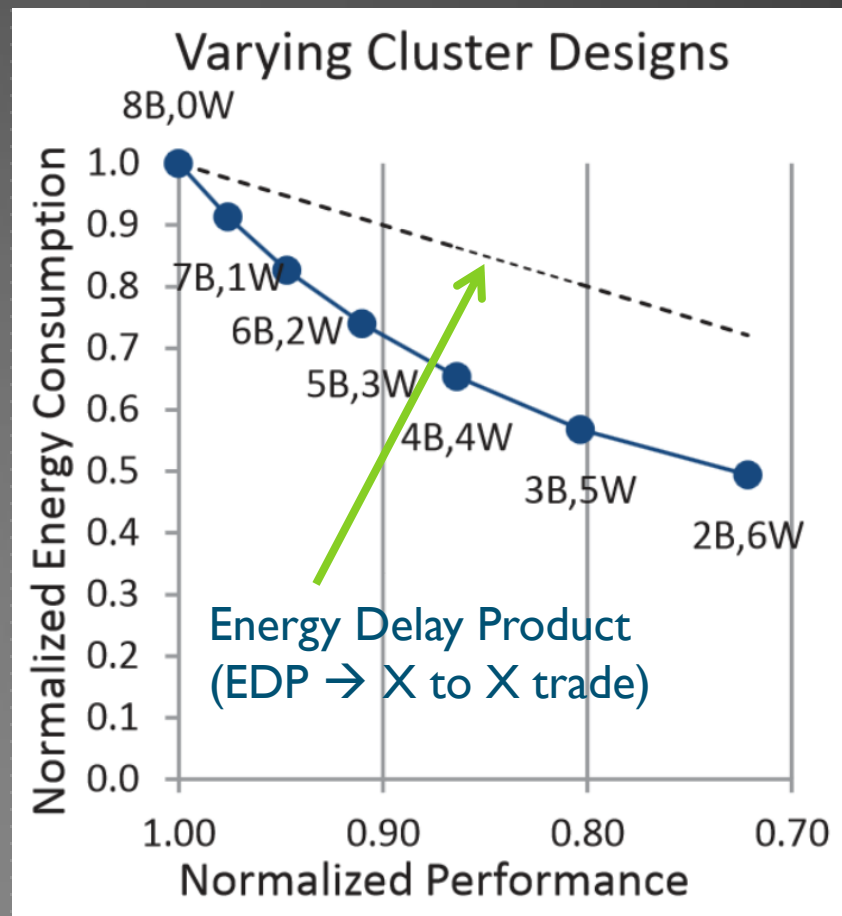
- ▶ Trade performance for reduced energy consumption
- ▶ Results
 - ▶ Sub-linear speedup
 - ▶ Resources  Performance 
- ▶ Data points above EDP curve
- ▶ Eg: 10 N
 - ▶ 24% penalty in performance
 - ▶ 16% decrease in energy



Vertica – running TPC-H Q12

VARYING CLUSTER DESIGNS

- ▶ Performing parallel hash join
P-store
- ▶ Heterogeneous cluster design
- ▶ Wimpy scan and filter data and
send to Beefy
- ▶ Results
 - ▶ Data points below EDP curve
 - ▶ Greater energy savings for less
performance penalty

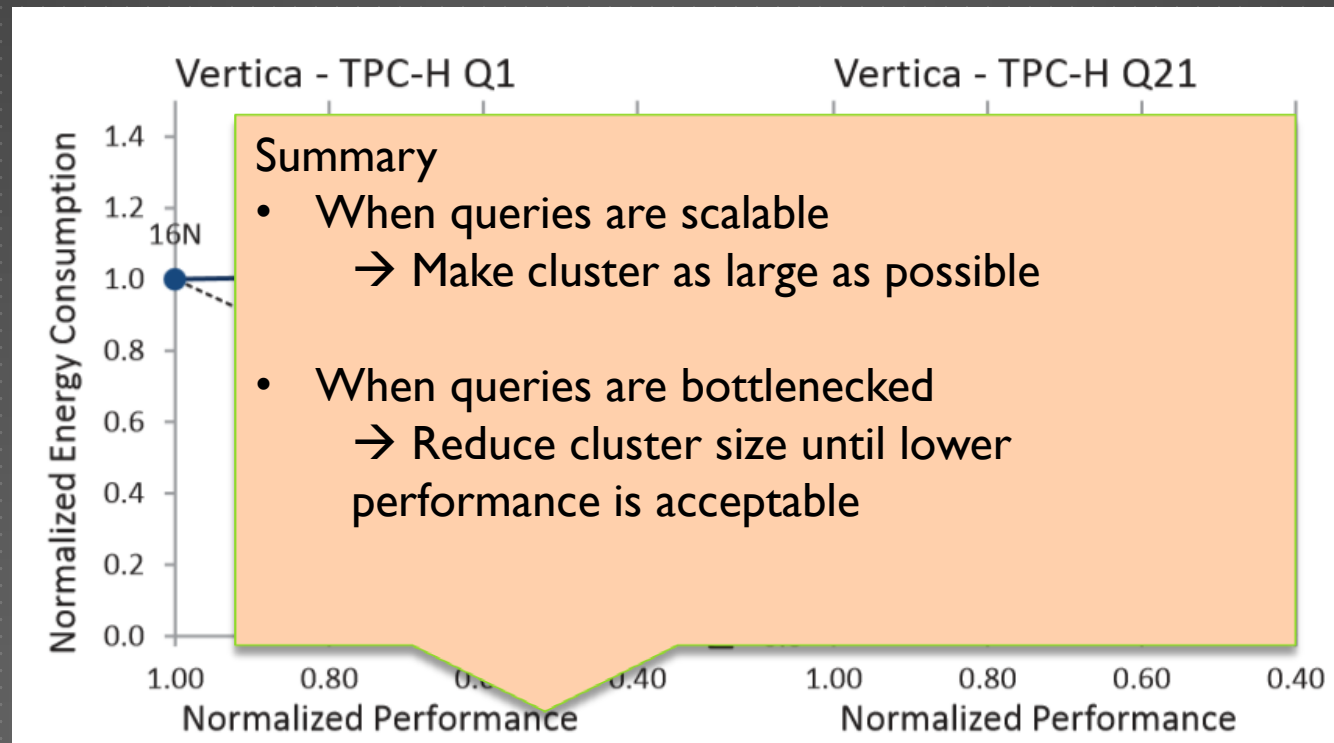


8 node cluster of Beefy (B) and Wimpy (W)

CONTRIBUTIONS

- ▶ Explore trade-offs in performance versus energy efficiency
 - ▶ **Vertica, P-store**, HadoopDB
- ▶ Identify bottlenecks for performance and energy efficiency
- ▶ Build a model which predicts performance and energy efficiency for various cluster configurations
- ▶ Illustrate interesting cluster design points
- ▶ Provide guiding principles for energy-efficient data processing
- ▶ Seed future research in this area

PERFORMANCE VERSUS ENERGY EFFICIENCY



- Simple aggregation, no joins
- Performance scales linearly
- Constant energy consumption
- Hence, add as many nodes as possible

- 94.5% of query on local machines
- Performance scales linearly
- Constant energy consumption
- Hence, add as many nodes as possible
- Refer (I)

BOTTLENECKS

- ▶ Hardware (network and disk)
 - ▶ Repartitioning → internode communication
 - ▶ Node waits for data from network
- ▶ Algorithmic (broadcast)
 - ▶ Broadcast takes same time regardless of number of nodes
 - ▶ Eg: 16 N → each node receives $15m/16$, 32 N → each node receives $31m/32$
- ▶ Data skew
 - ▶ Part of future work

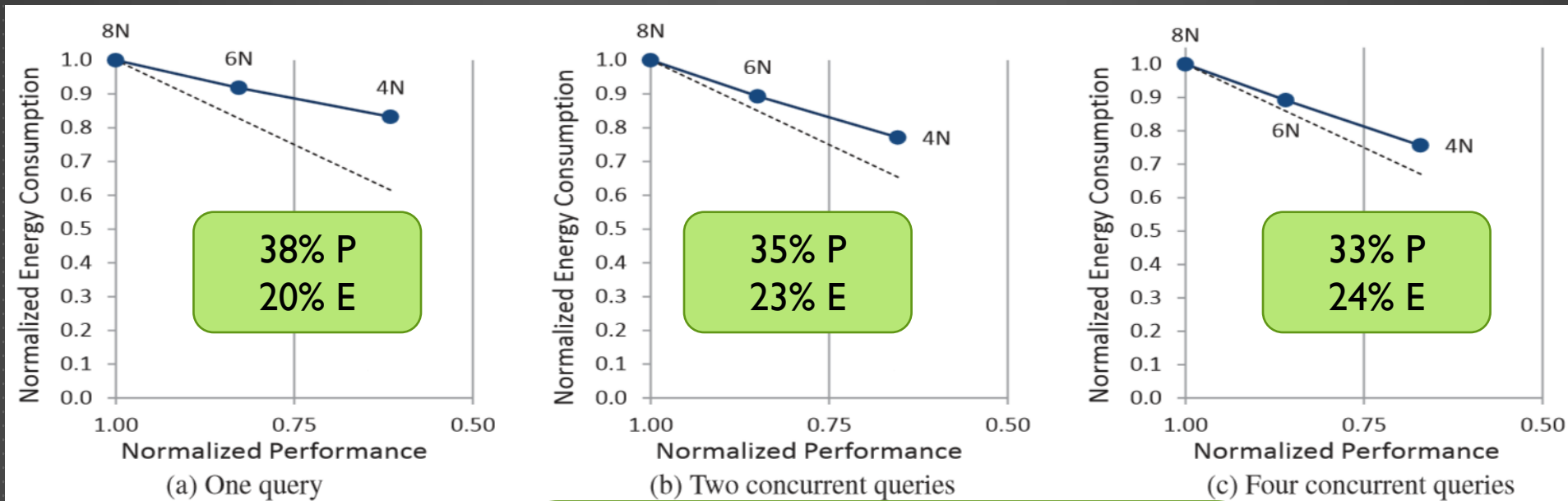
BUILDING A MODEL

- ▶ P-store
 - ▶ Custom built parallel engine → scan, project, select, hash join, network exchange
- ▶ Explore performance bottlenecks affecting energy efficiency
 - ▶ Hash join query
 - ▶ Cluster V configuration

DBMS	Vertica	RAM	48GB
# nodes	16	Disks	8x300GB
TPC-H size	1TB (scale 1000)	Network	1Gb/s
CPU	Intel X5550 2 sockets	SysPower	130.03C ^{0.2369}
			C = CPU utilization

EXPERIMENT #I

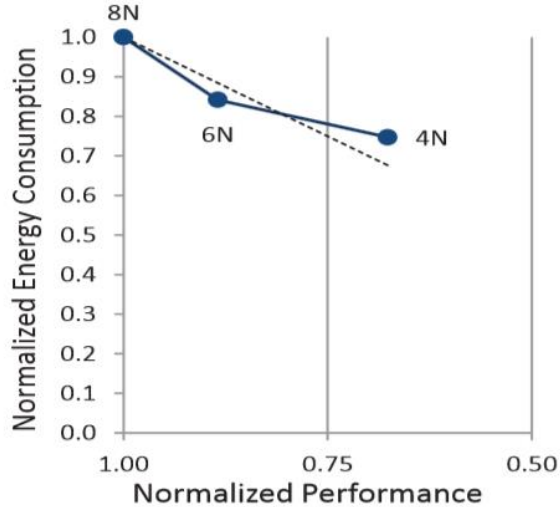
TPC-H Q3 hash join



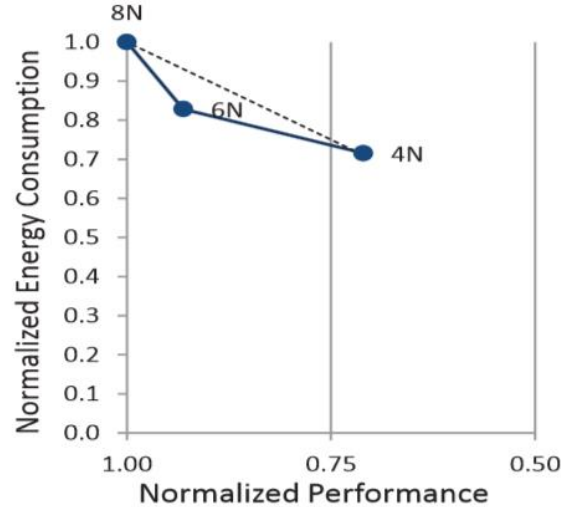
- Poor performance scalability
- Energy savings increases as concurrency increases (data points closer to EDP)
- Reason → CPU utilization does not scale because of network bottleneck

EXPERIMENT #2

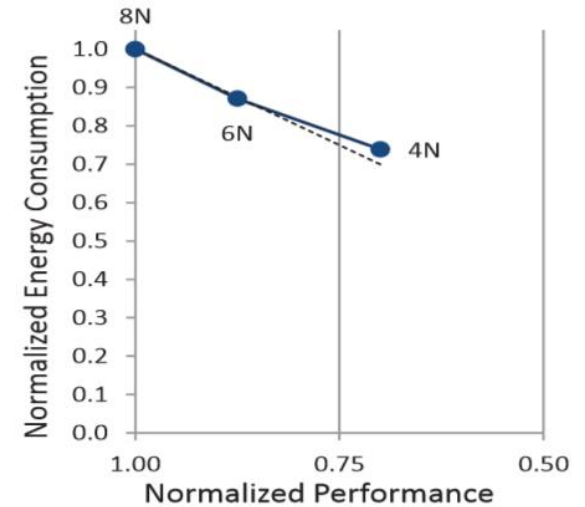
TPC-H Q3 broadcast join



(a) One query



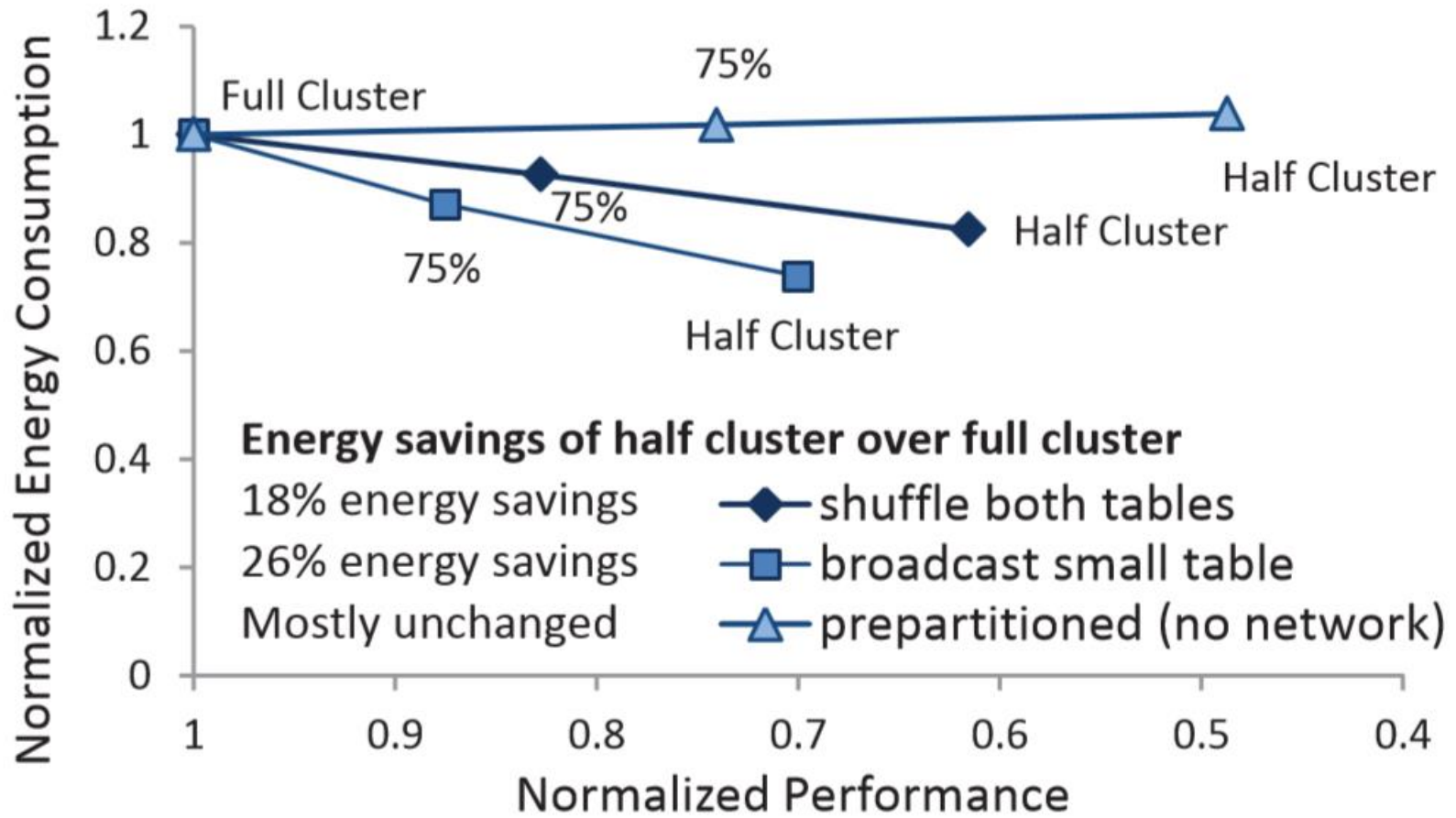
(b) Two concurrent queries



(c) Four concurrent queries

- Energy savings increases as concurrency increases (data points closer to EDP)
- Suffers non-linear scalability
- Reason → Broadcast does not scale

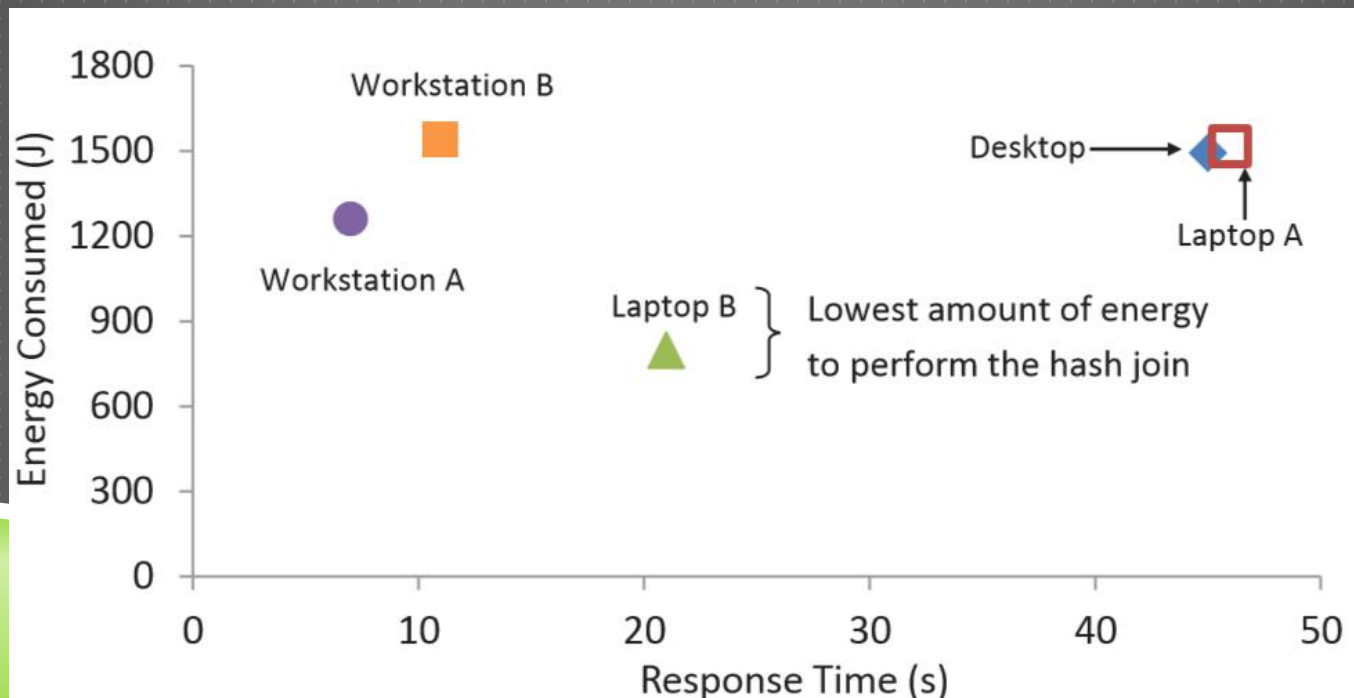
SUMMARY OF NETWORK AND ALGORITHMIC BOTTLENECKS



ENERGY EFFICIENCY OF INDIVIDUAL NODES

System	CPU (cores/threads)	RAM	Idle Power (W)
Workstation A	i7 920 (4/8)	12GB	93W
Workstation B	Xeon (4/4)	24GB	69W
Desktop	Atom (2/4)	4GB	28W
Laptop A	Core 2 Duo (2/2)	4GB	12W (screen off)
Laptop B	i7 620m (2/4)	8GB	11W (screen off)

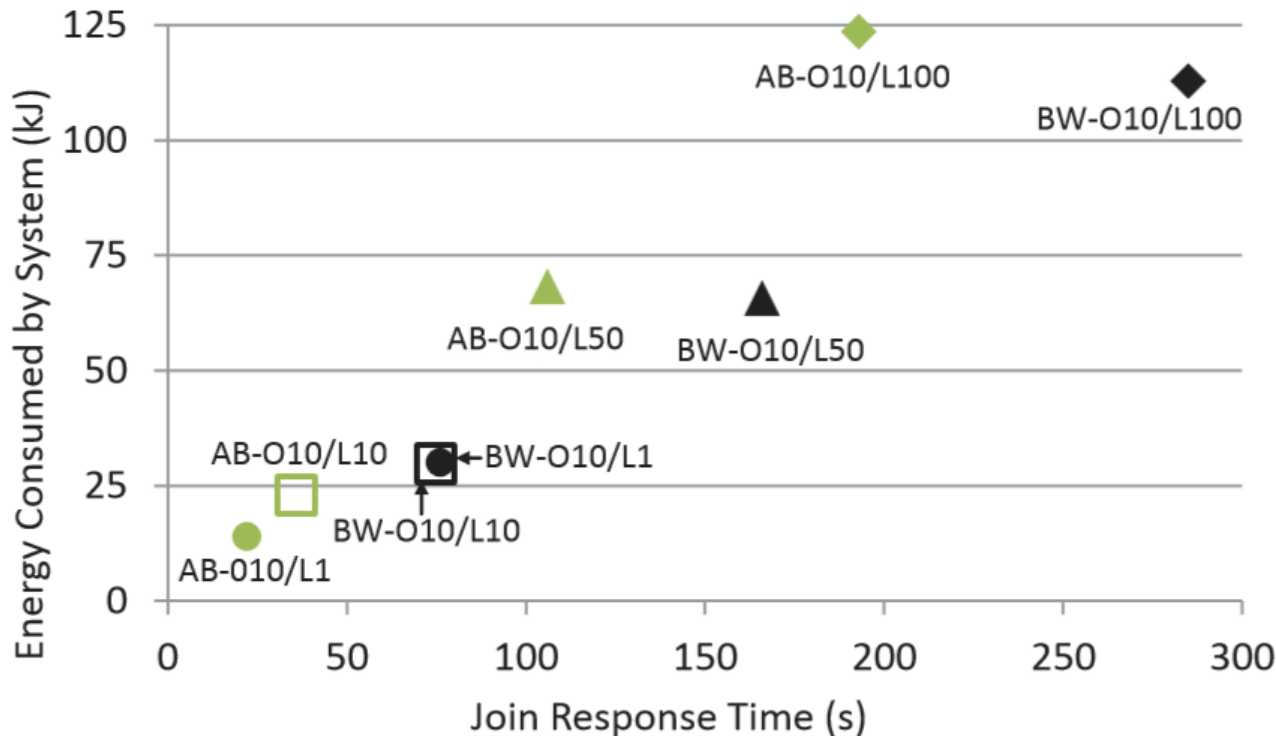
- In-memory workload
- Hash join



CLUSTER DESIGN POINTS

- Beefy → 4 HP ProLiant servers with quad core Nehalem Xeon processors
- Each node has 32 GB of memory
- Average power 154 W

- 2 Beefy/2 Wimpy → 2 Beefy nodes + 2 Laptop Bs with i7 processor
- Wimpy node has 8 GB of memory
- Average laptop power 37W



- 1%, 10%, 50%, 100% selectivity on LINEITEM
- **10%** selectivity on ORDERS
- Beefy nodes build hash tables, Wimpy scan and filter
- Heterogeneous Execution

MODELING P-STORE AND BOTTLENECKS

- Understand the nature of query parameters and scalability bottleneck
- Predict the performance and energy consumption of various ways to execute a hash join
- Parameters

- Hash join
 1. Build phase
 2. Probe phase

T_{bld}	Build phase time (s)	T_{prb}	Probe phase time (s)
E_{bld}	Build phase energy (J)	E_{prb}	Probe phase energy (J)
N_B	# Beefy nodes	N_W	# Wimpy nodes
M_B	Beefy memory size (MB)	M_W	Wimpy memory size (MB)
I	Disk bandwidth (MB/s)	L	Network bandwidth (MB/s)
Bld	Hash join build table size (MB)	Prb	Hash join probe table size (MB)
S_{bld}	Build table predicate selectivity	S_{prb}	Probe table predicate selectivity
R_{Wbld}	Rate at which a Wimpy node builds its hash table (MB/s)		
R_{Bbld}	Rate at which a Beefy node builds its hash table (MB/s)		
U_{Wbld}	Wimpy node CPU bandwidth during the build phase		
U_{Bbld}	Beefy node CPU bandwidth during the build phase		
R_{Wprb}	Rate at which the Wimpy node probes its hash table (MB/s)		
R_{Bprb}	Rate at which the Beefy node probes its hash table (MB/s)		
U_{Wprb}	Wimpy node CPU bandwidth during the probe phase		
U_{Bprb}	Beefy node CPU bandwidth during the probe phase		
$C_B = 5037$	Maximum CPU bandwidth of a Beefy node (MB/s)		
$C_W = 1129$	Maximum CPU bandwidth of a Wimpy node (MB/s)		
$G_B = 0.25$	Beefy CPU utilization constants for P-store		
$G_W = 0.13$	Wimpy CPU utilization constants for P-store		
$f_B(c) = 130.03 \times (100c)^{0.2369}$ (c=CPU util.)		Beefy node power model	
$f_W(c) = 10.994 \times (100c)^{0.2875}$ (c=CPU util.)		Wimpy node power model	
$H = M_W \geq (Bld * Bld_{sel}) / (N_B + N_W)$		Wimpy can build the hash table	

HOMOGENEOUS EXECUTION

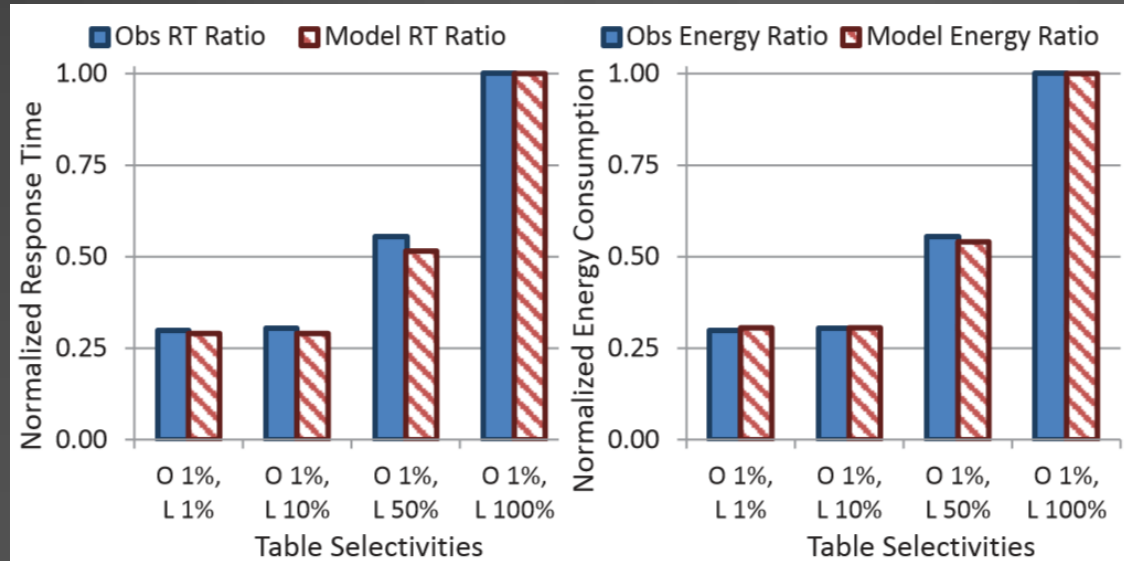
$$T_{prb} = \frac{Prb \times S_{prb}}{(N_B R_{Bprb}) + (N_W R_{Wprb})}$$

$$E_{prb} = T_{prb} \times (N_B f_B (G_B + \frac{U_{Bprb}}{C_B}) + N_W f_W (G_W + \frac{U_{Wprb}}{C_W}))$$

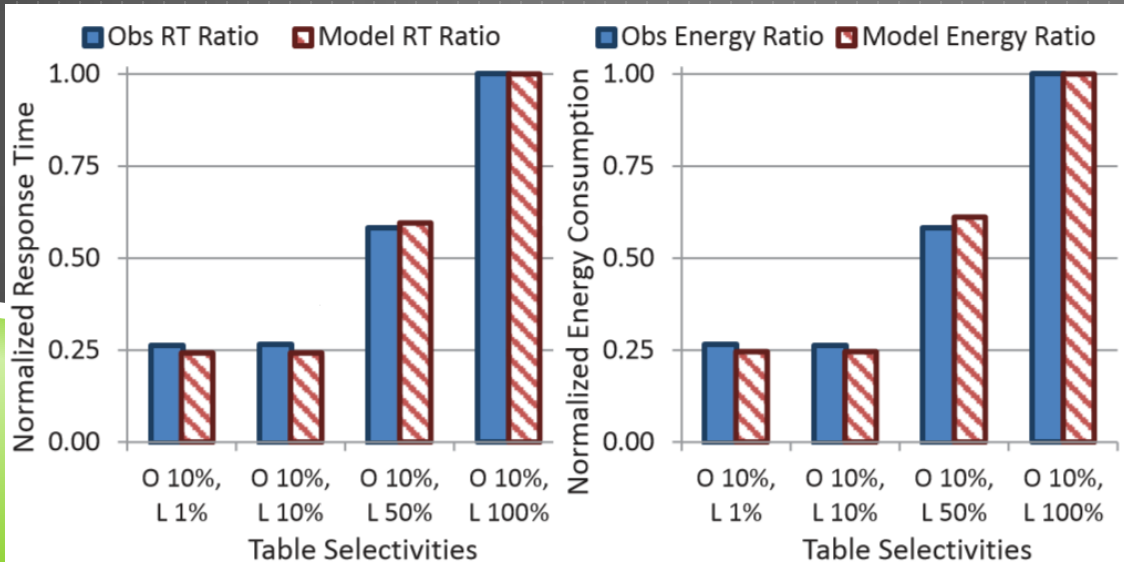
Probe phase
response time and
cluster energy
consumption

T_{bld}	Build phase time (s)	T_{prb}	Probe phase time (s)
E_{bld}	Build phase energy (J)	E_{prb}	Probe phase energy (J)
N_B	# Beefy nodes	N_W	# Wimpy nodes
M_B	Beefy memory size (MB)	M_W	Wimpy memory size (MB)
I	Disk bandwidth (MB/s)	L	Network bandwidth (MB/s)
Bld	Hash join build table size (MB)	Prb	Hash join probe table size (MB)
S_{bld}	Build table predicate selectivity	S_{prb}	Probe table predicate selectivity
R_{Wbld}	Rate at which a Wimpy node builds its hash table (MB/s)		
R_{Bbld}	Rate at which a Beefy node builds its hash table (MB/s)		
U_{Wbld}	Wimpy node CPU bandwidth during the build phase		
U_{Bbld}	Beefy node CPU bandwidth during the build phase		
R_{Wprb}	Rate at which the Wimpy node probes its hash table (MB/s)		
R_{Bprb}	Rate at which the Beefy node probes its hash table (MB/s)		
U_{Wprb}	Wimpy node CPU bandwidth during the probe phase		
U_{Bprb}	Beefy node CPU bandwidth during the probe phase		
$C_B = 5037$	Maximum CPU bandwidth of a Beefy node (MB/s)		
$C_W = 1129$	Maximum CPU bandwidth of a Wimpy node (MB/s)		
$G_B = 0.25$	Beefy CPU utilization constants for P-store		
$G_W = 0.13$	Wimpy CPU utilization constants for P-store		
$f_B(c) = 130.03 \times (100c)^{0.2369}$	(c=CPU util.) Beefy node power model		
$f_W(c) = 10.994 \times (100c)^{0.2875}$	(c=CPU util.) Wimpy node power model		
$H = M_W \geq (Bld * Bld_{sel}) / (N_B + N_W)$	Wimpy can build the hash table		

MODEL VALIDATION

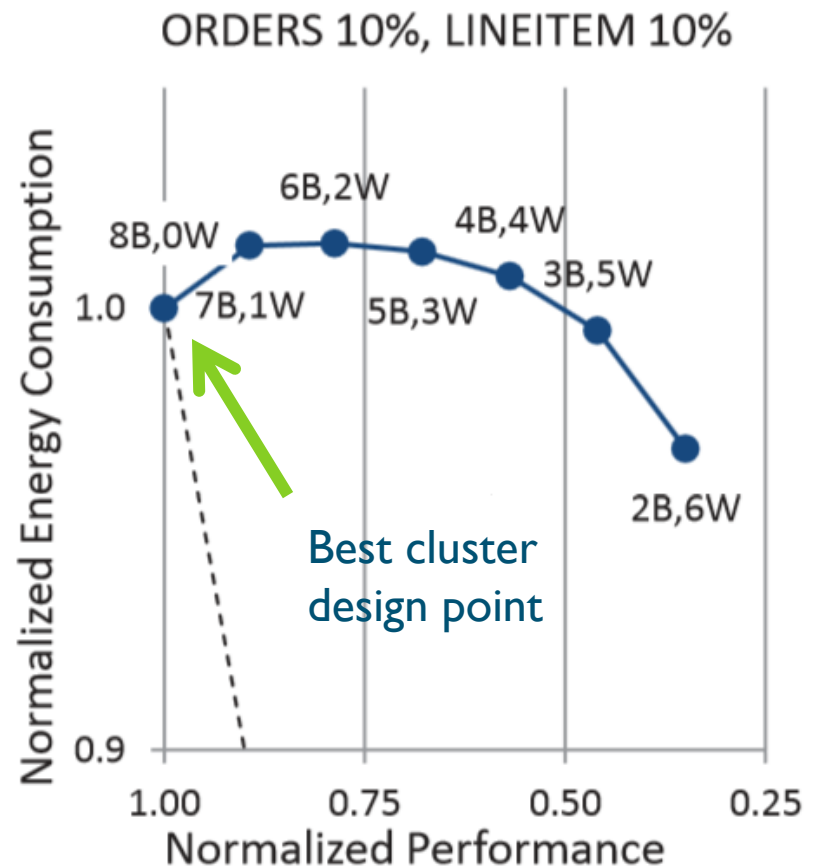
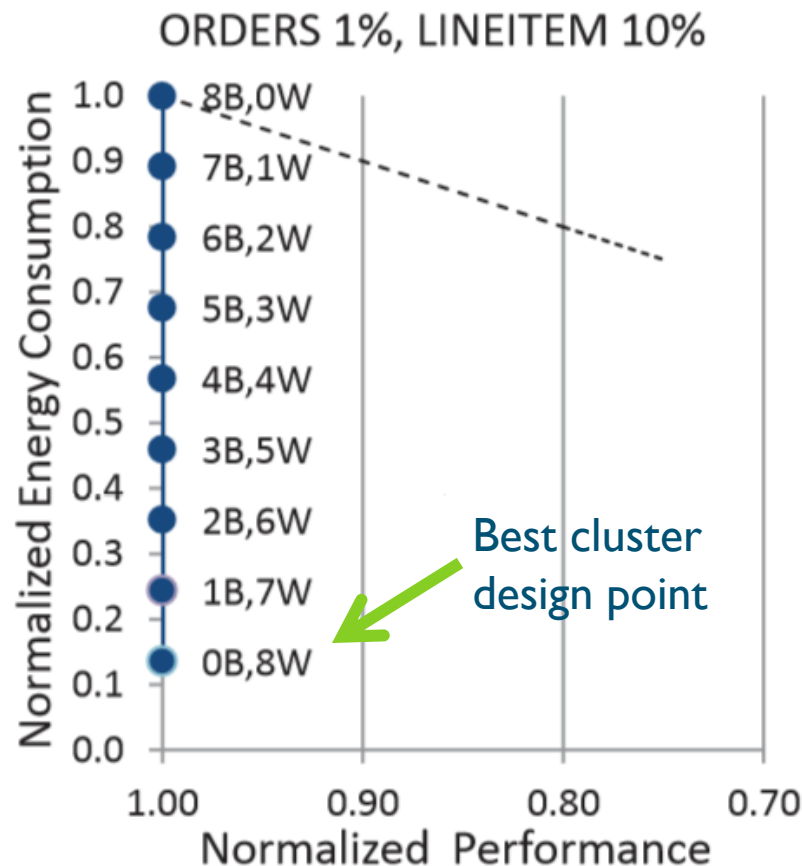


- Homogeneous Execution
- Error < 5%



- Heterogeneous Execution
- Error < 10%

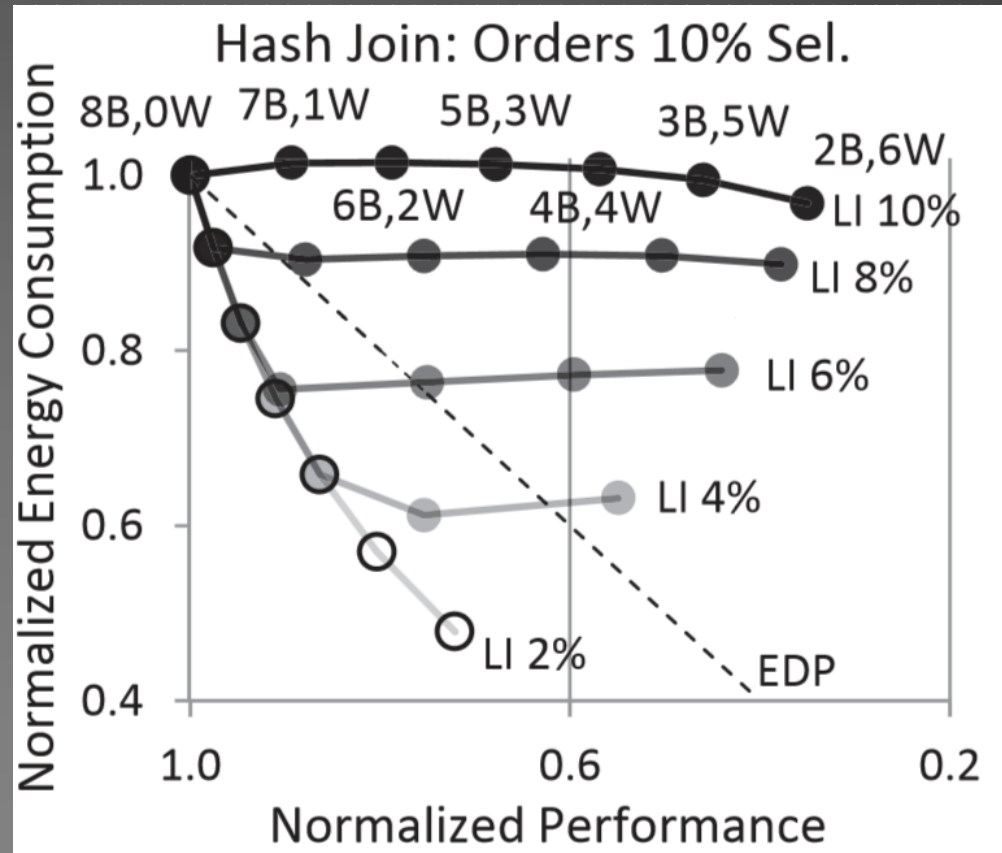
EXPLORING QUERY AND CLUSTER PARAMETERS



P-store hash join performance
and energy efficiency

EXPLORING QUERY AND CLUSTER PARAMETERS

P-store hash join performance and energy efficiency LINEITEM (2-10% selectivity)

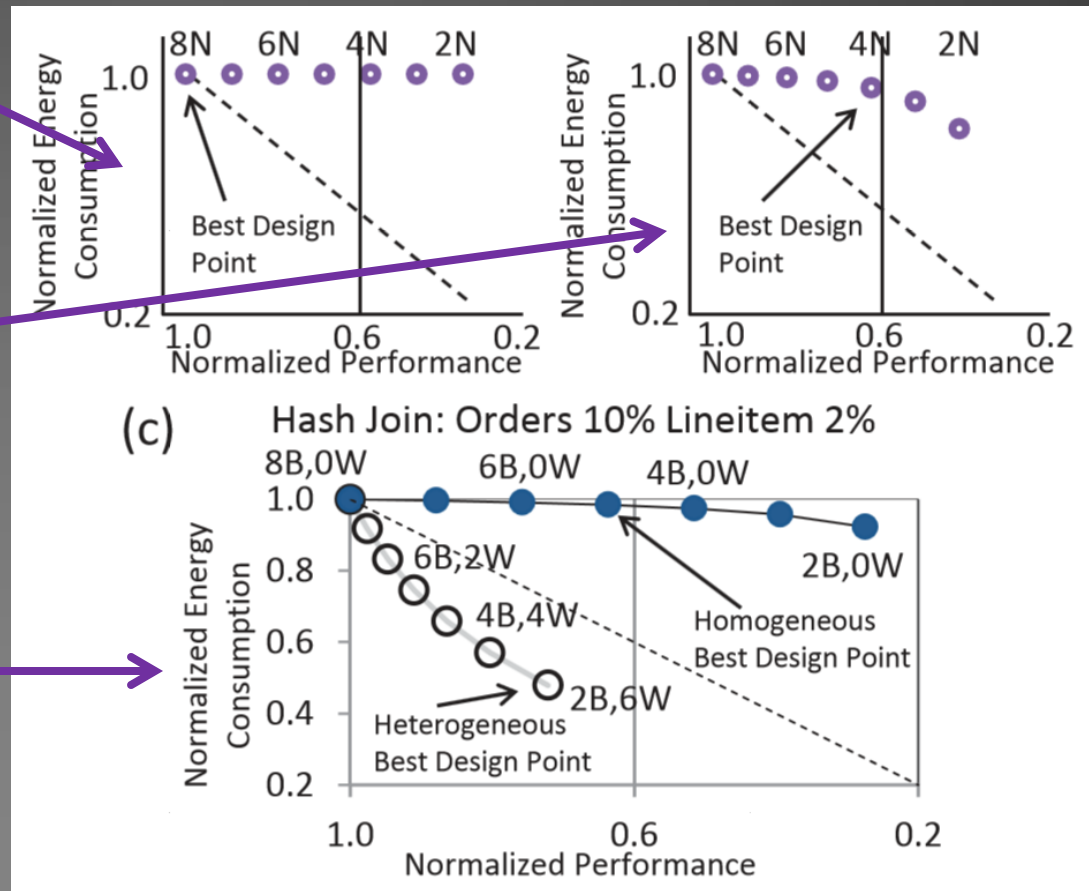


CLUSTER DESIGN PRINCIPLES (SUMMARY OF THE PAPER)

Query is highly scalable

Query is not scalable
→ Reduce performance
to meet required target
(SLAs)

Query is not scalable
→ Homogeneous/
Heterogeneous cluster



DISCUSSION

- ▶ Modeling a system. How easy/difficult?
- ▶ Only single queries used. Acknowledged to include more workloads.
- ▶ Max cluster size is 16N
- ▶ The break-even time of installing new clusters has not been discussed
 - ▶ Homogeneous \leftrightarrow Heterogenous
- ▶ Dynamic configuration of workloads/servers

Thank you!!