MY DATABASE SYSTEM IS THE ONLY THING I CAN

TRUST

@ANDY_PAVLO

Thirty Years Ago...

- Interactive Transactions
- Small # of CPU Cores
- Small Memory Sizes

TPC-C BENCHMARK

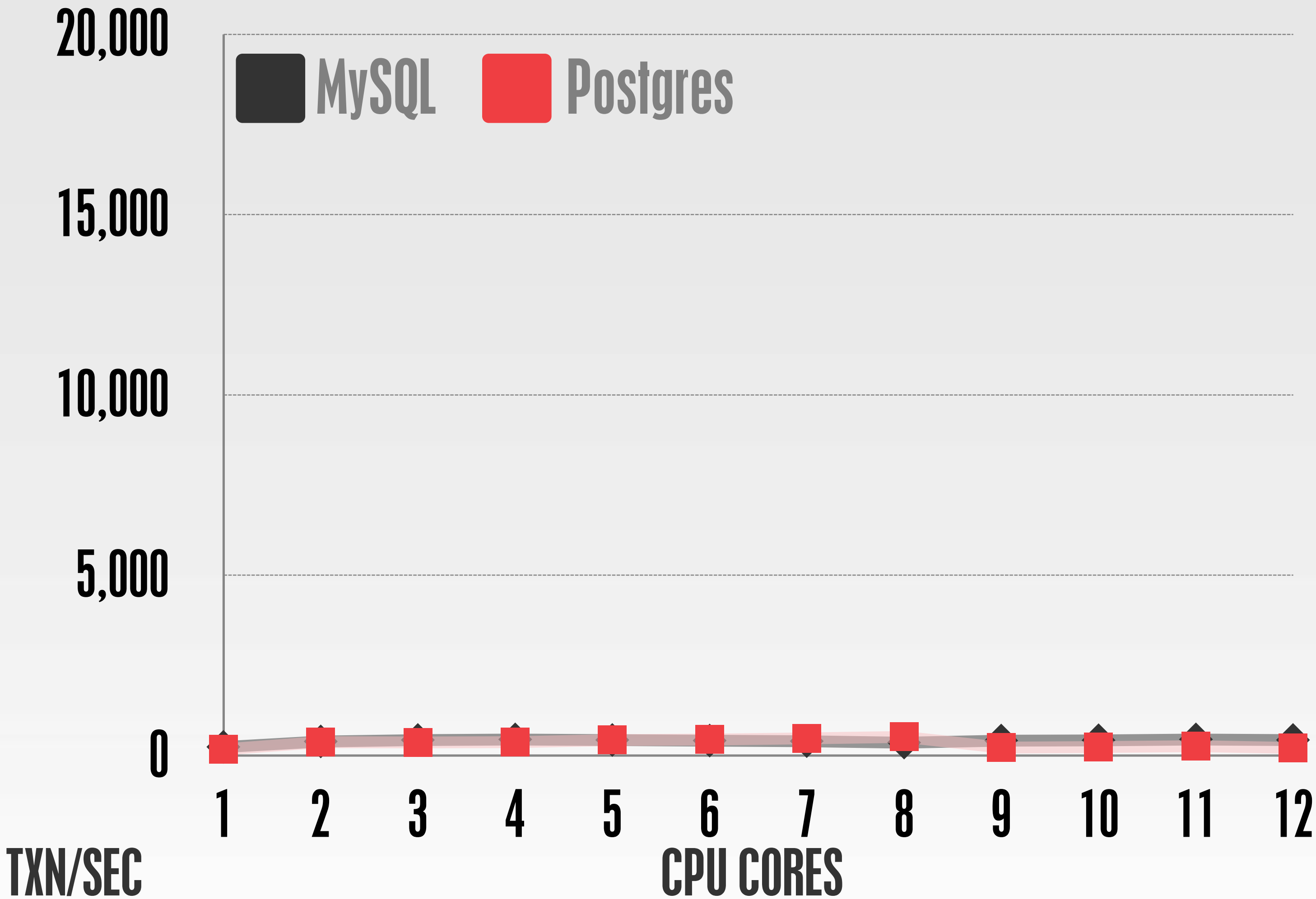
Warehouse Order Processing

NewOrder Transaction

1. Check item stock level.
2. Create new order information.
3. Update item stock levels.

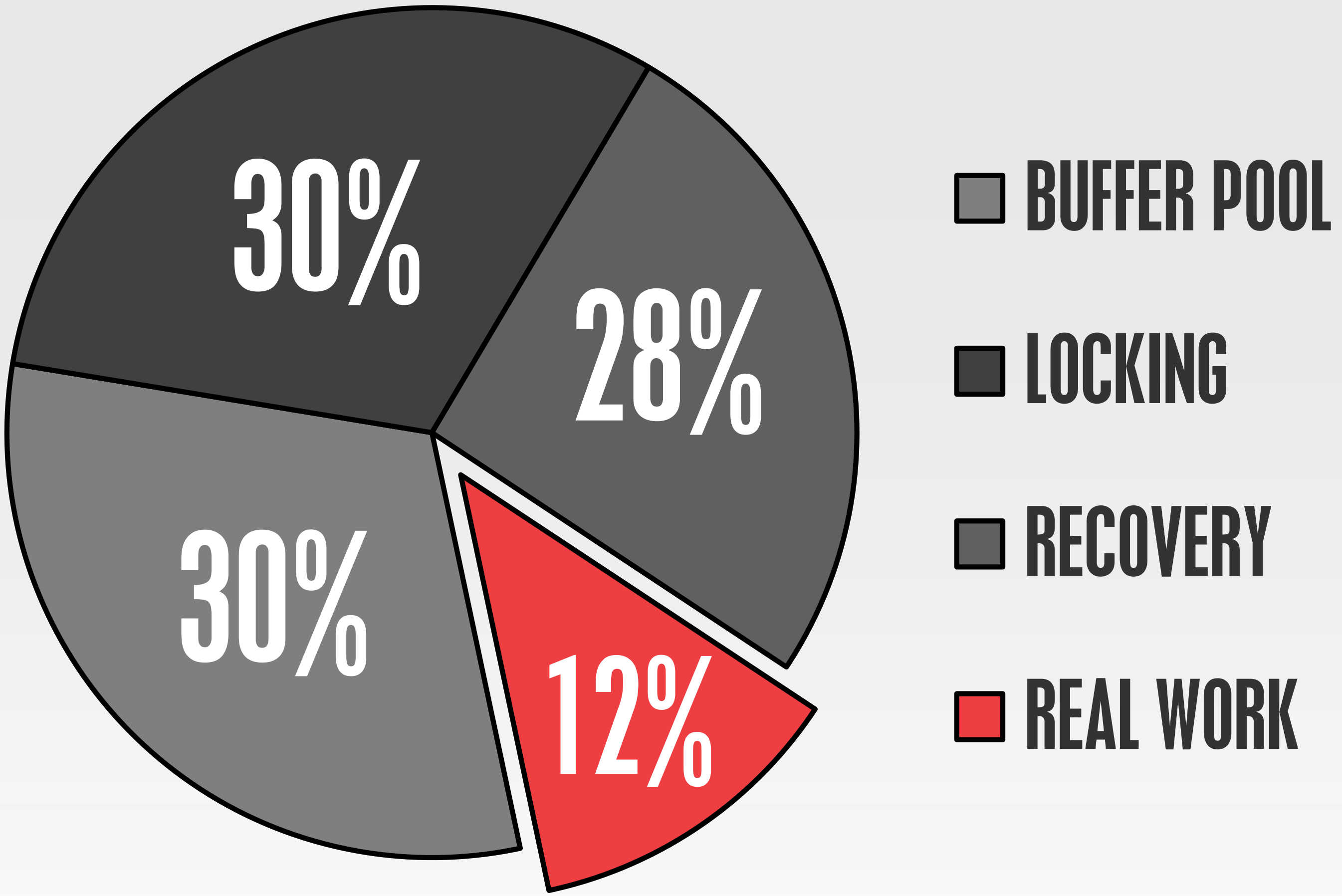
TPC-C BENCHMARK

Warehouse Order Processing



TRADITIONAL DBMS

Measured CPU Cycles



OLTP THROUGH THE LOOKING GLASS,
AND WHAT WE FOUND THERE
SIGMOD, pp. 981-992, 2008.

- Hardware Upgrade
- Replication
- Distributed Cache
- Sharding Middleware
- NoSQL

HOW TO SCALE UP WITHOUT GIVING UP TRANSACTIONS?

H-Store

Distributed Main Memory Transaction Processing System

H-STORE: A HIGH-PERFORMANCE, DISTRIBUTED MAIN MEMORY TRANSACTION PROCESSING SYSTEM

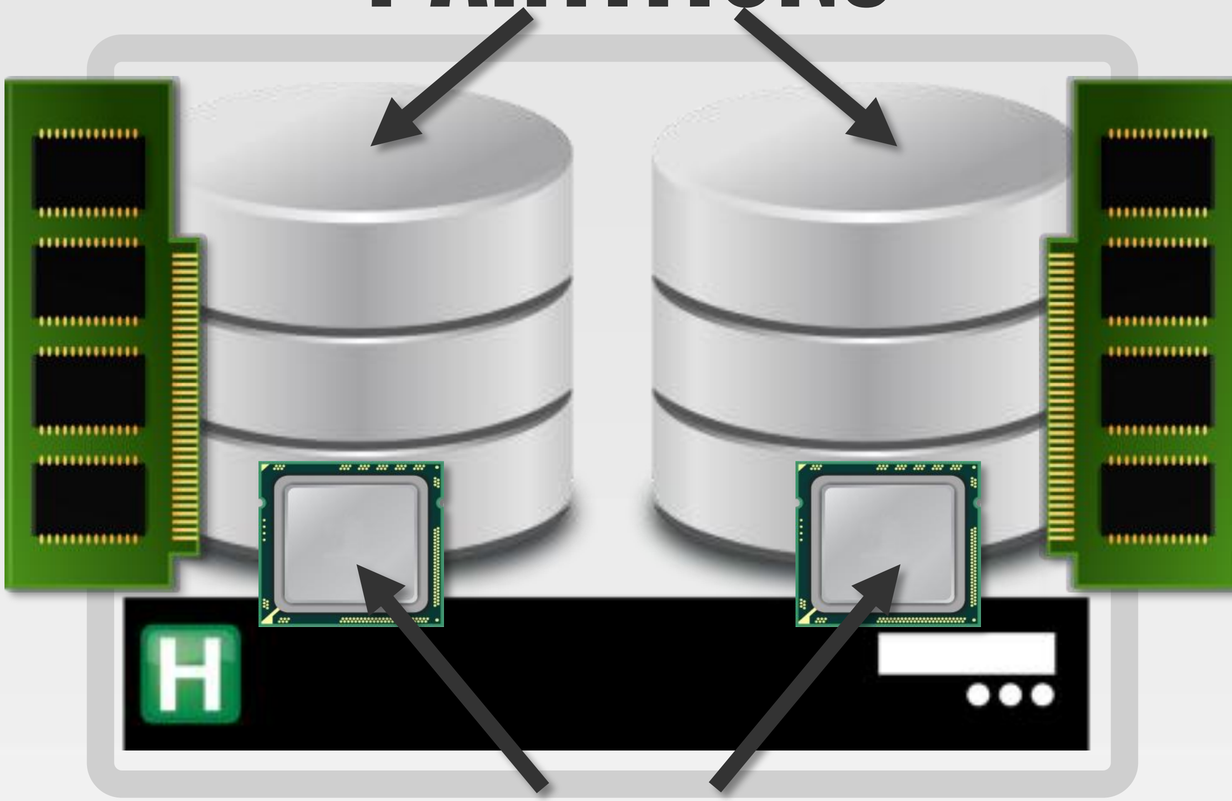
Proc. VLDB Endow., vol. 1, iss. 2, pp. 1496-1499, 2008.

DISK ORIENTED
MAIN MEMORY STORAGE

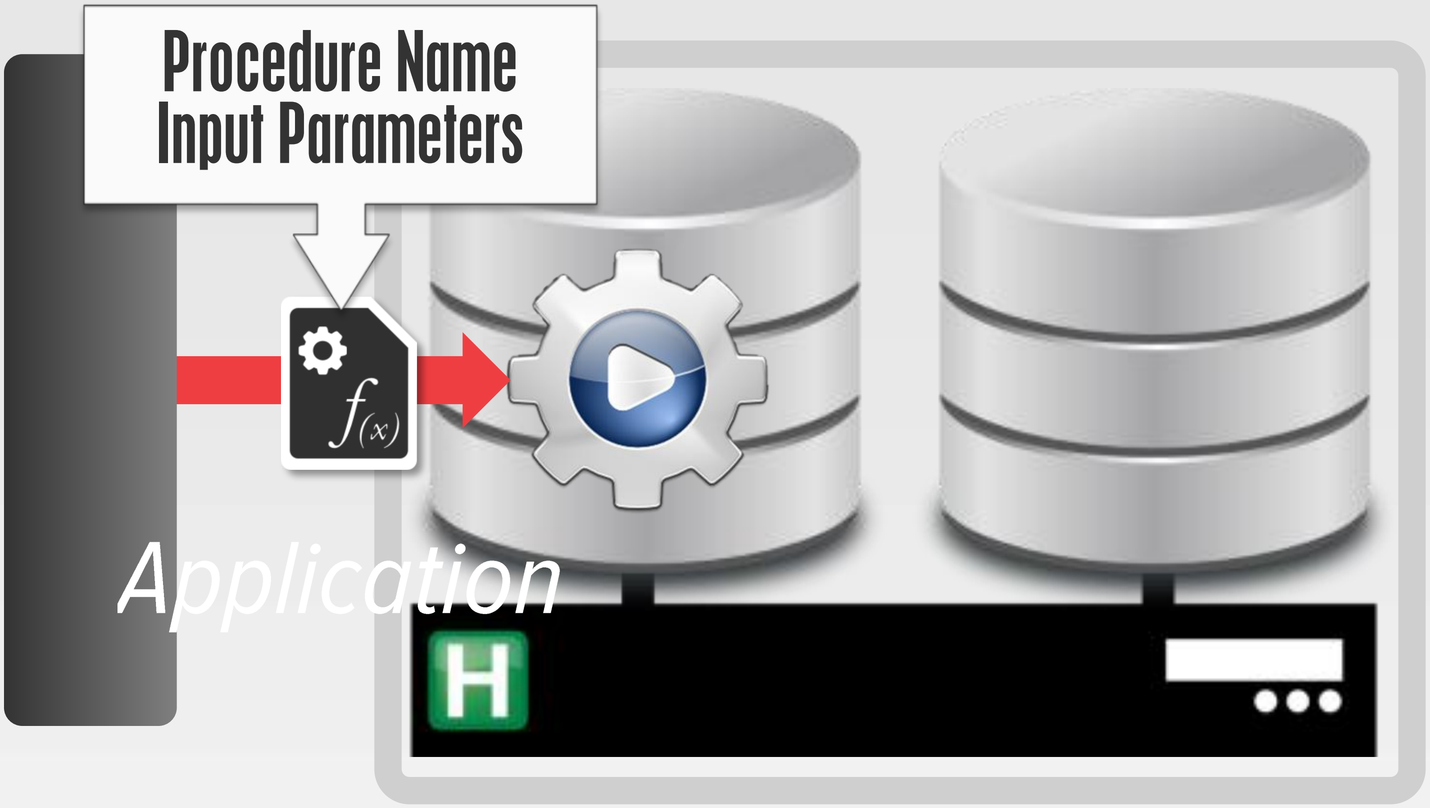
CONCURRENT EXECUTION
SERIAL EXECUTION

HEAVYWEIGHT RECOVERY
COMPACT LOGGING

PARTITIONS



SINGLE-THREADED EXECUTION ENGINES


Procedure Name
Input Parameters
Application

STORED PROCEDURE

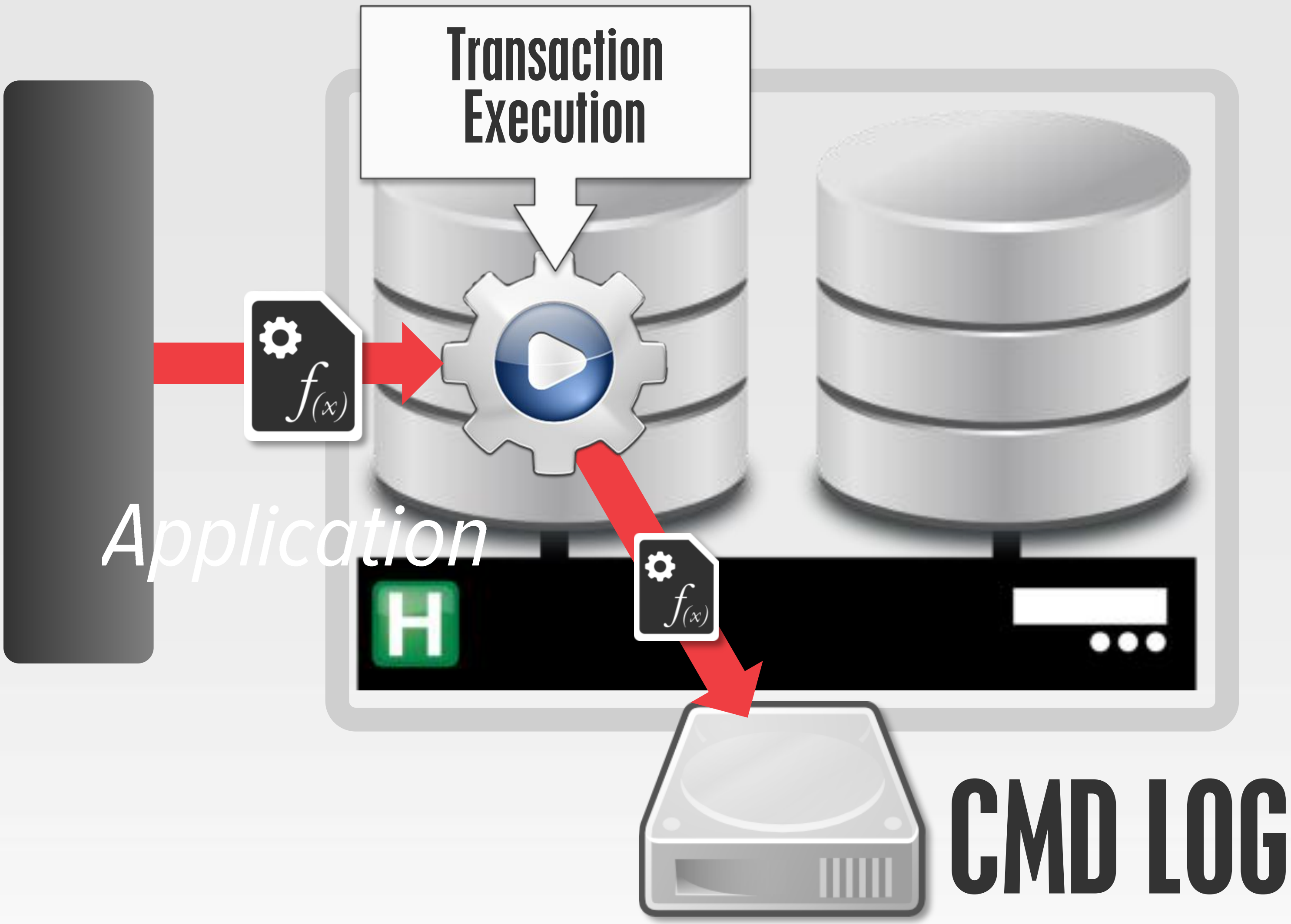
VoteCount:

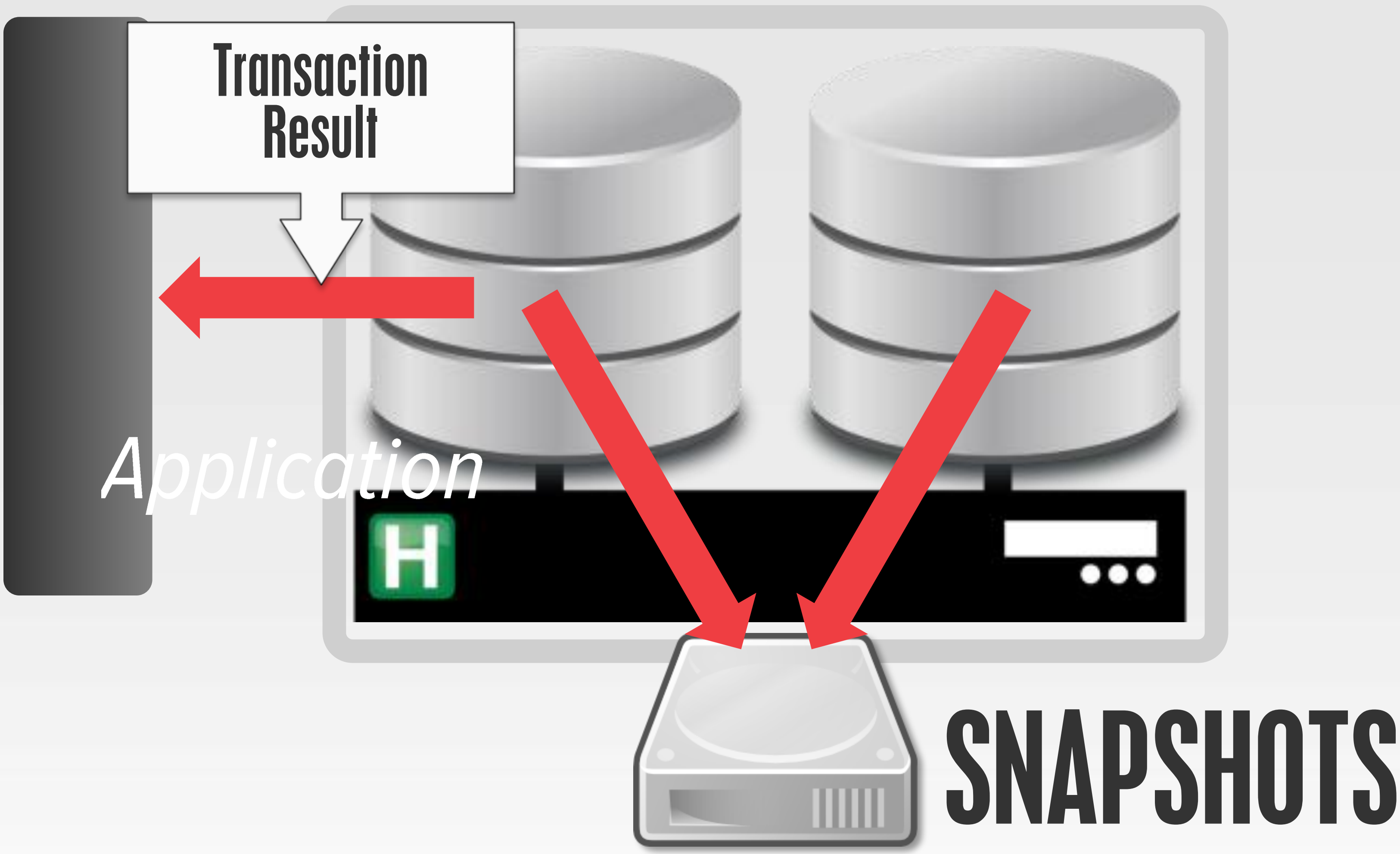
```
SELECT COUNT(*)
  FROM votes
 WHERE phone_num = ?;
```

InsertVote:

```
INSERT INTO votes
 VALUES (?, ?, ?);
```

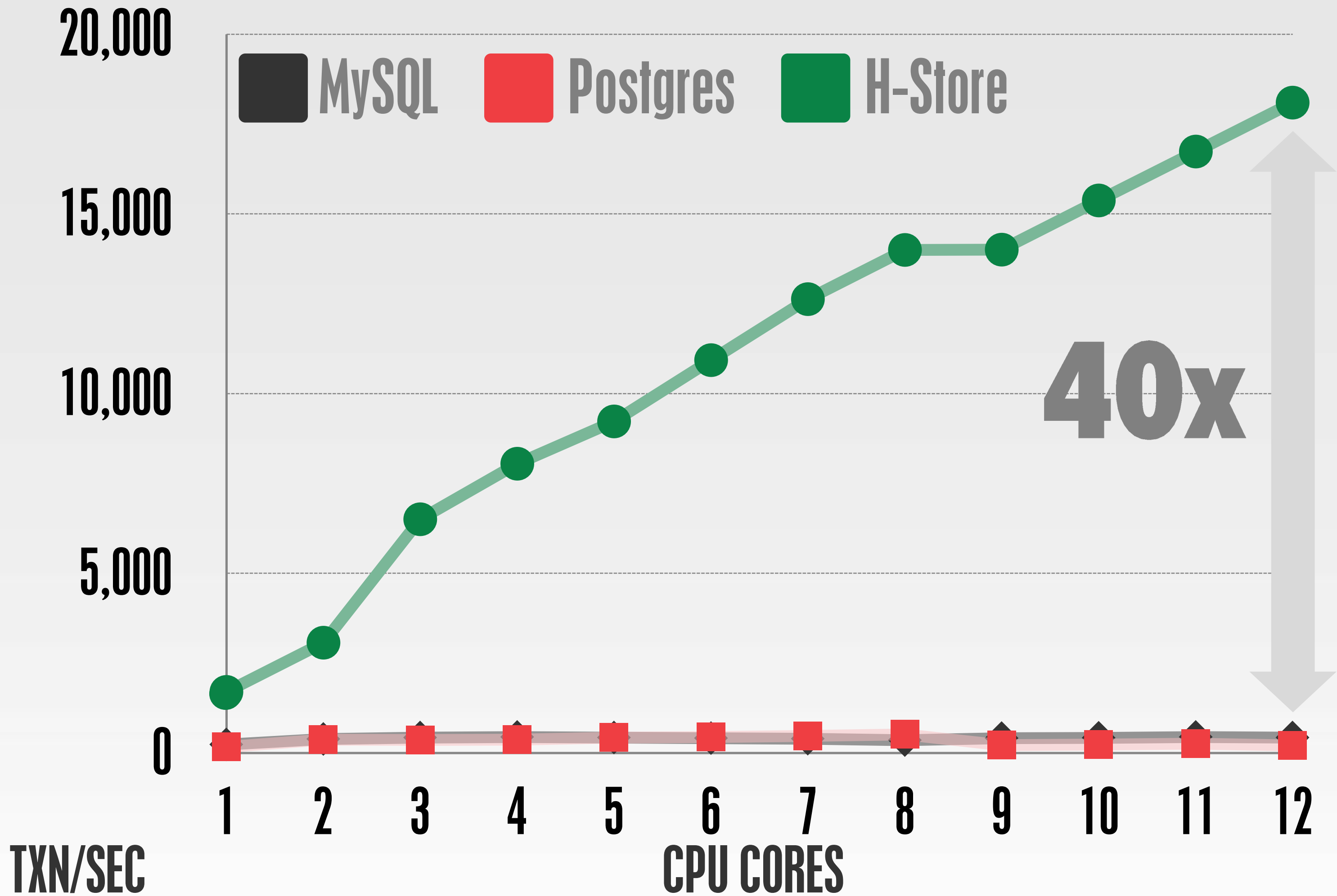
```
run(phoneNum, contestantId, currentTime) {
   result = execute(VoteCount, phoneNum);
   if (result > MAX_VOTES) {
      return (ERROR);
   }
   execute(InsertVote, phoneNum,
                       contestantId,
                       currentTime);
   return (SUCCESS);
}
```


Transaction
Execution
Application
CMD LOG


Transaction Result
Application
SNAPSHOTS

TPC-C BENCHMARK

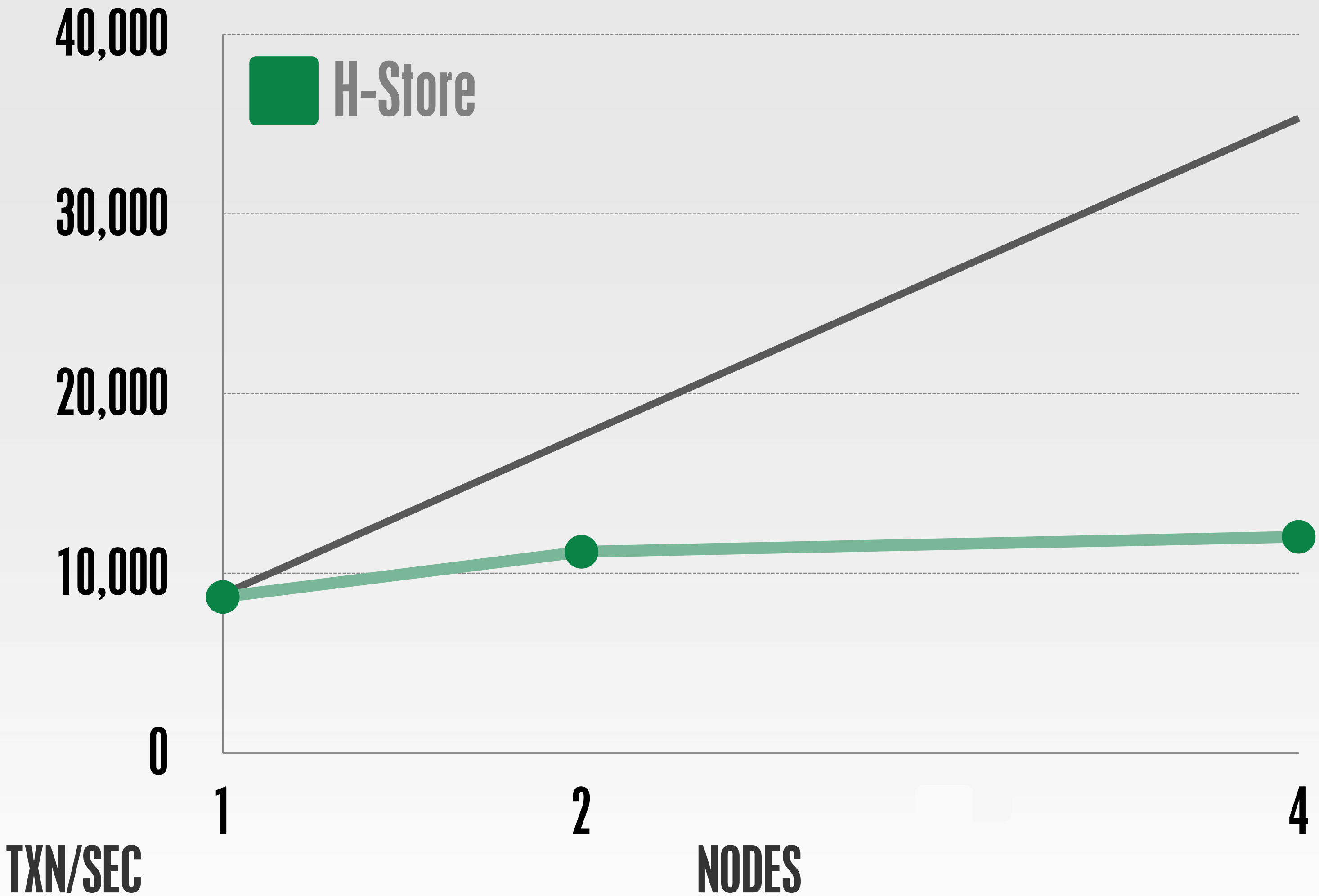
Warehouse Order Processing



DISTRIBUTED TRANSACTIONS

TPC-C BENCHMARK

8 Cores per Node
10% Distributed Transactions



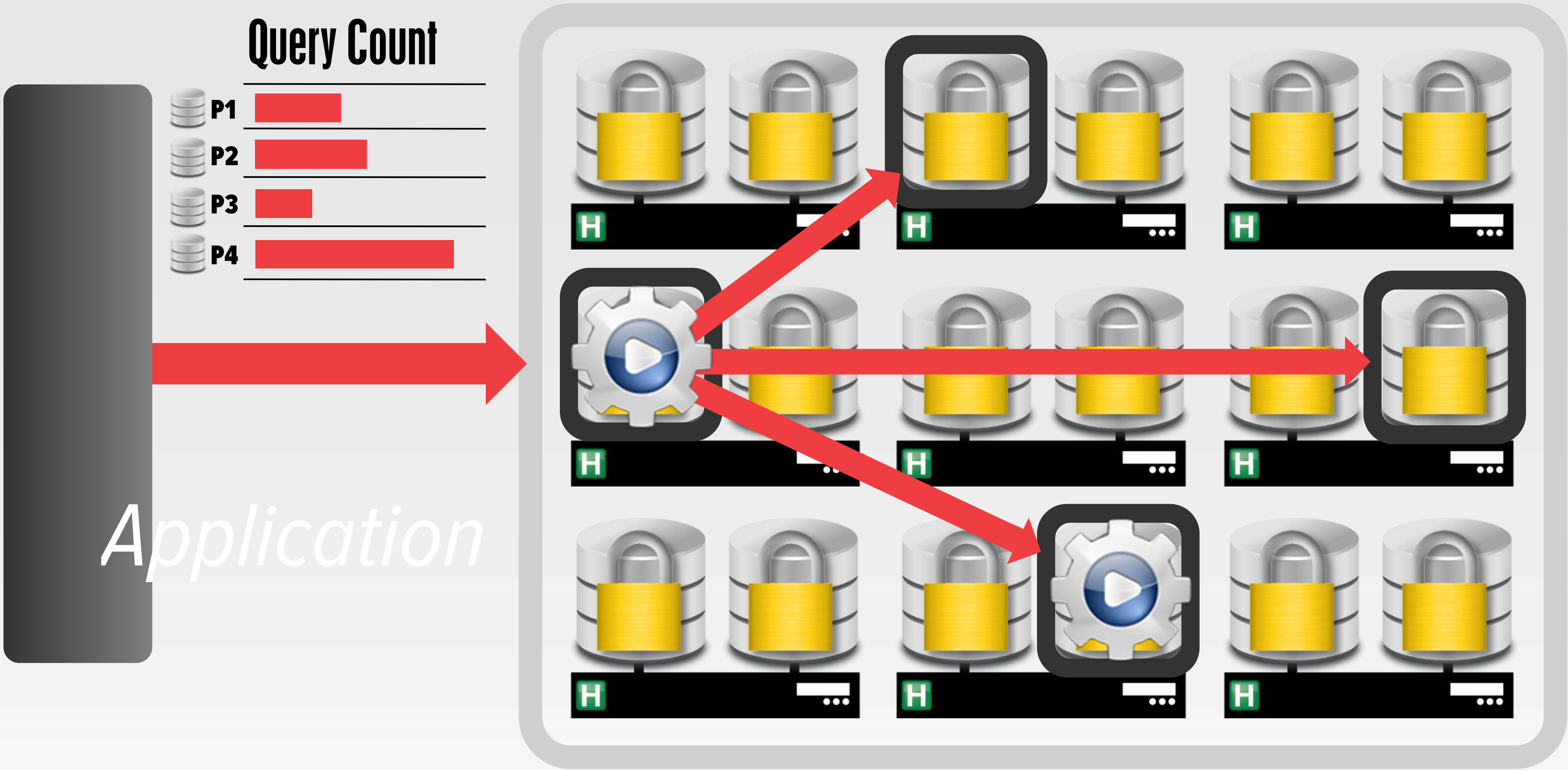
H-Store

DISTRIBUTED TRANSACTIONS

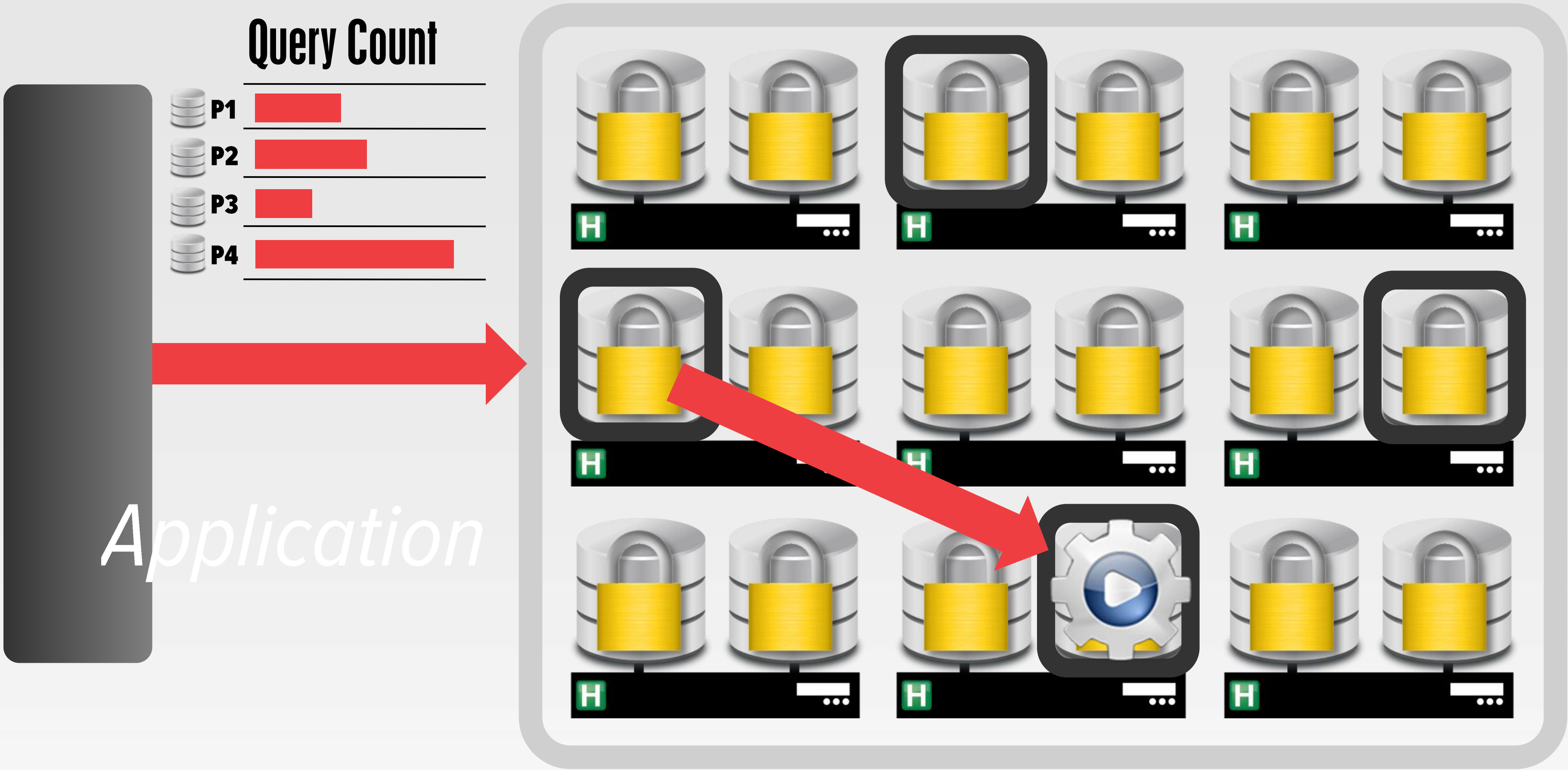
DISTRIBUTED TRANSACTIONS

H-Store

DISTRIBUTED TRANSACTIONS



DISTRIBUTED TRANSACTIONS

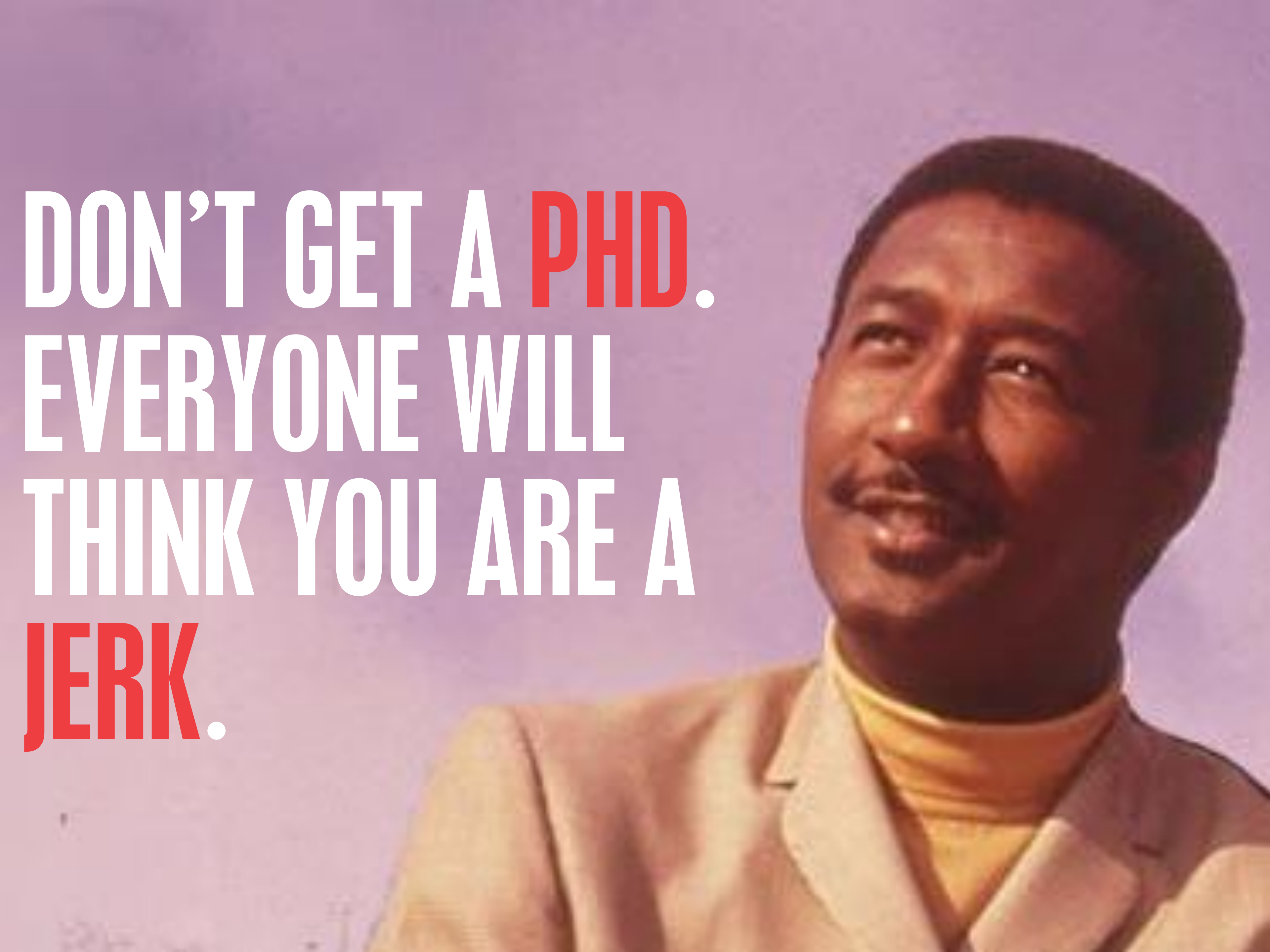


H-Store

KNOW WHAT TRANSACTIONS WILL DO BEFORE THEY START

BUT PEOPLE ALWAYS GIVE ME BAD ADVICE


DON'T GET
INVOLVED WITH
COMPUTERS.
YOU'LL NEVER
MAKE ANY
MONEY.


DON'T GET A PHD.
EVERYONE WILL
THINK YOU ARE A
JERK.

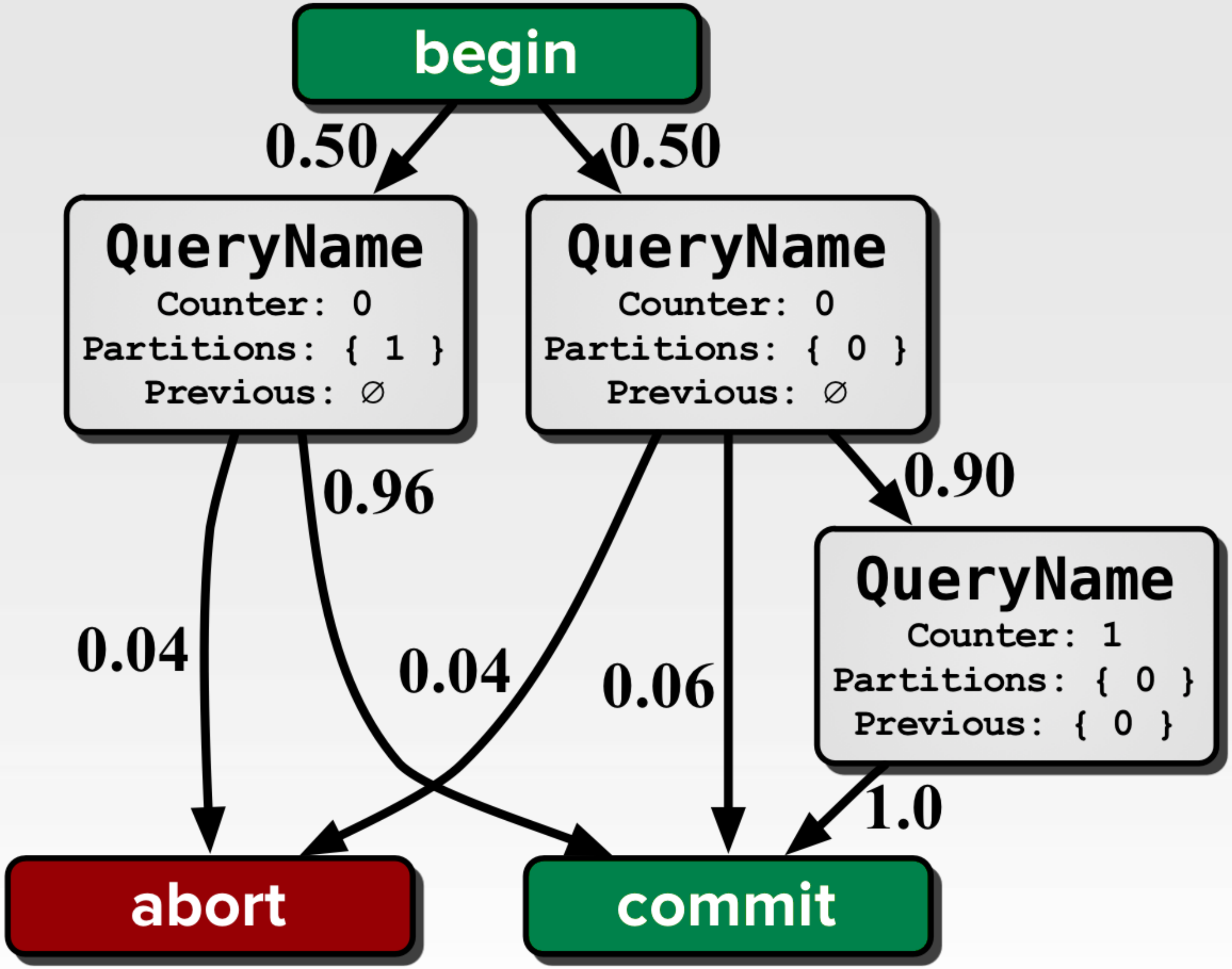
THE DATABASE SYSTEM ALWAYS HAS MORE

INFORMATION

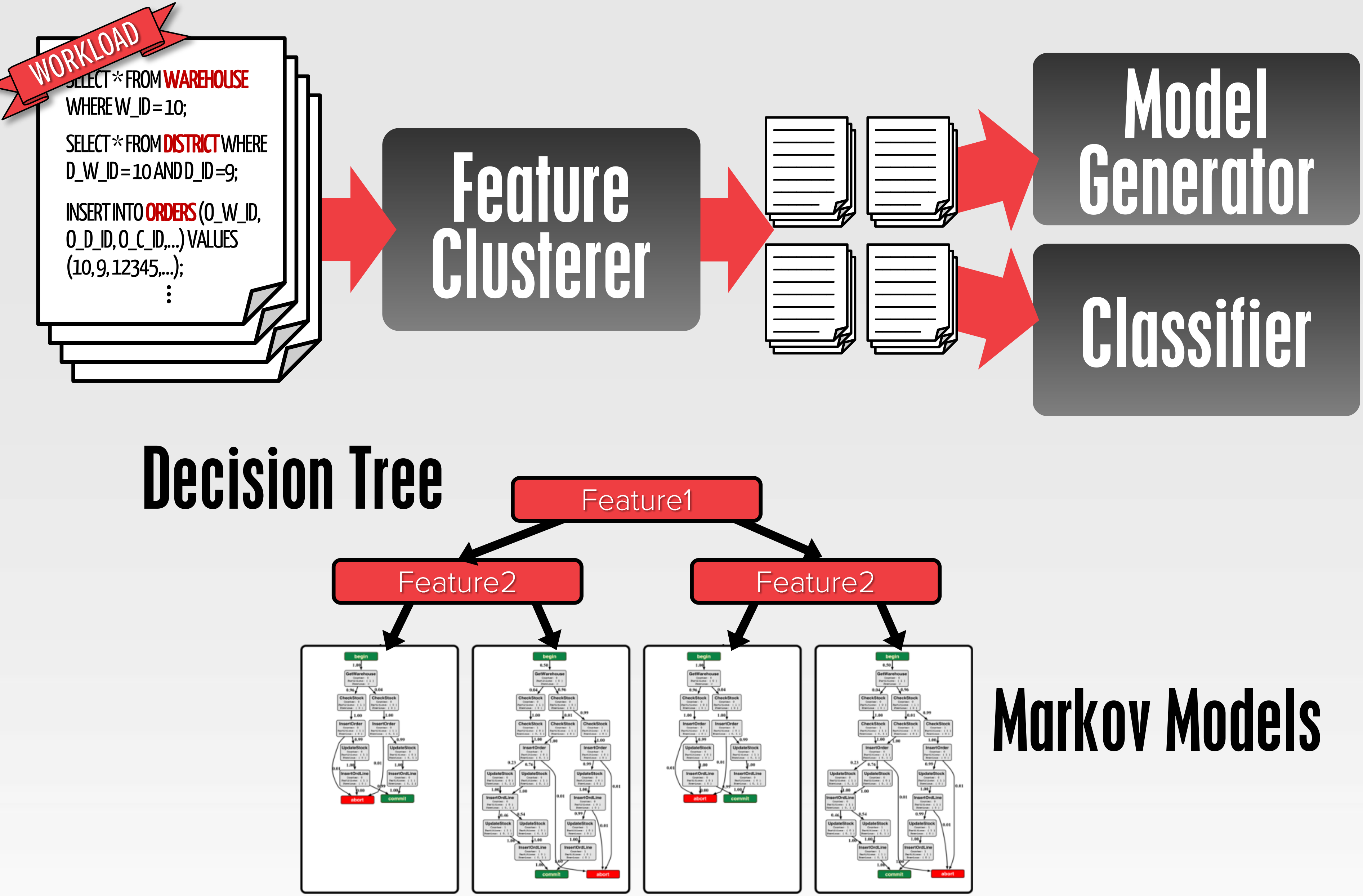
DO USE MACHINE LEARNING TO PREDICT TRANSACTION BEHAVIOR.

ON PREDICTIVE MODELING FOR OPTIMIZING TRANSACTION EXECUTION IN PARALLEL OLTP SYSTEMS
Proc. VLDB Endow., Vol 5, Iss. 2, pp. 85-96, 2011

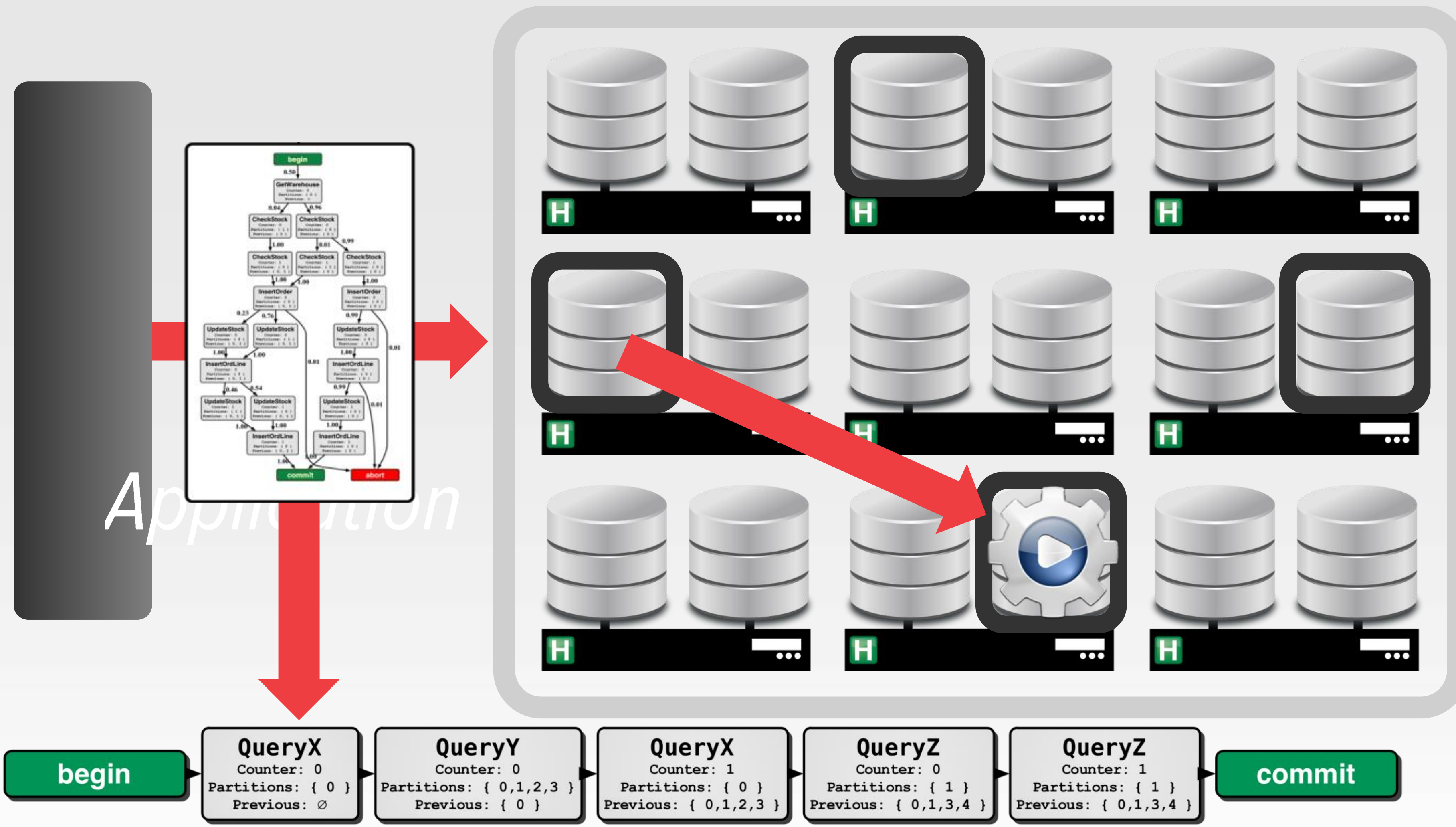
PREDICTIVE MODELS



Houdini

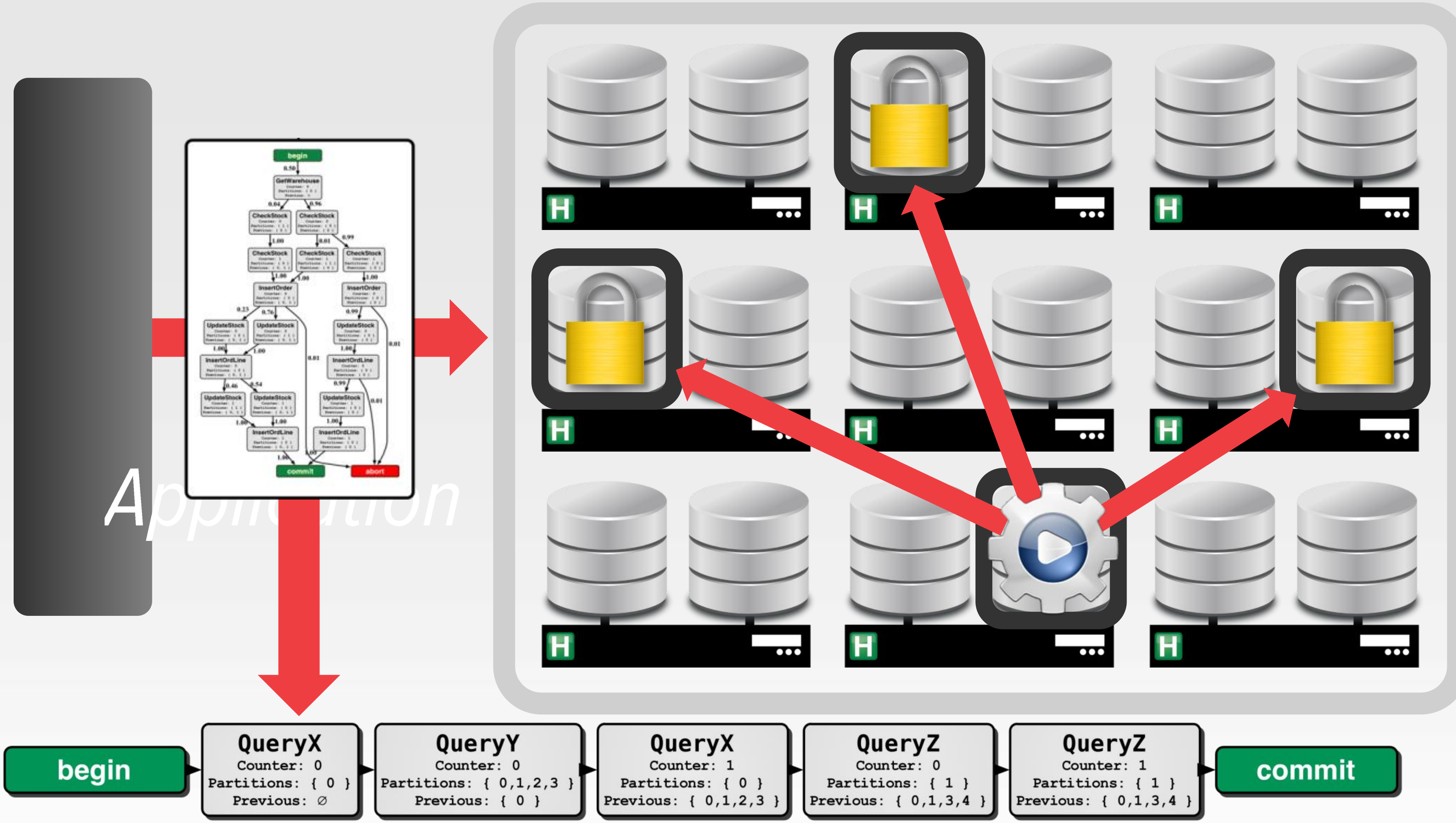

WORKLOAD
SELECT * FROM WAREHOUSE WHERE W_ID = 10;
SELECT * FROM DISTRICT WHERE D_W_ID = 10 AND D_ID = 9;
INSERT INTO ORDERS (O_W_ID, O_D_ID, O_C_ID,...) VALUES (10, 9, 12345,...);
⋮
Feature Clusterer
Model Generator
Classifier
Decision Tree
Feature1
Feature2
Feature2
Markov Models

DISTRIBUTED TRANSACTIONS



Houdini

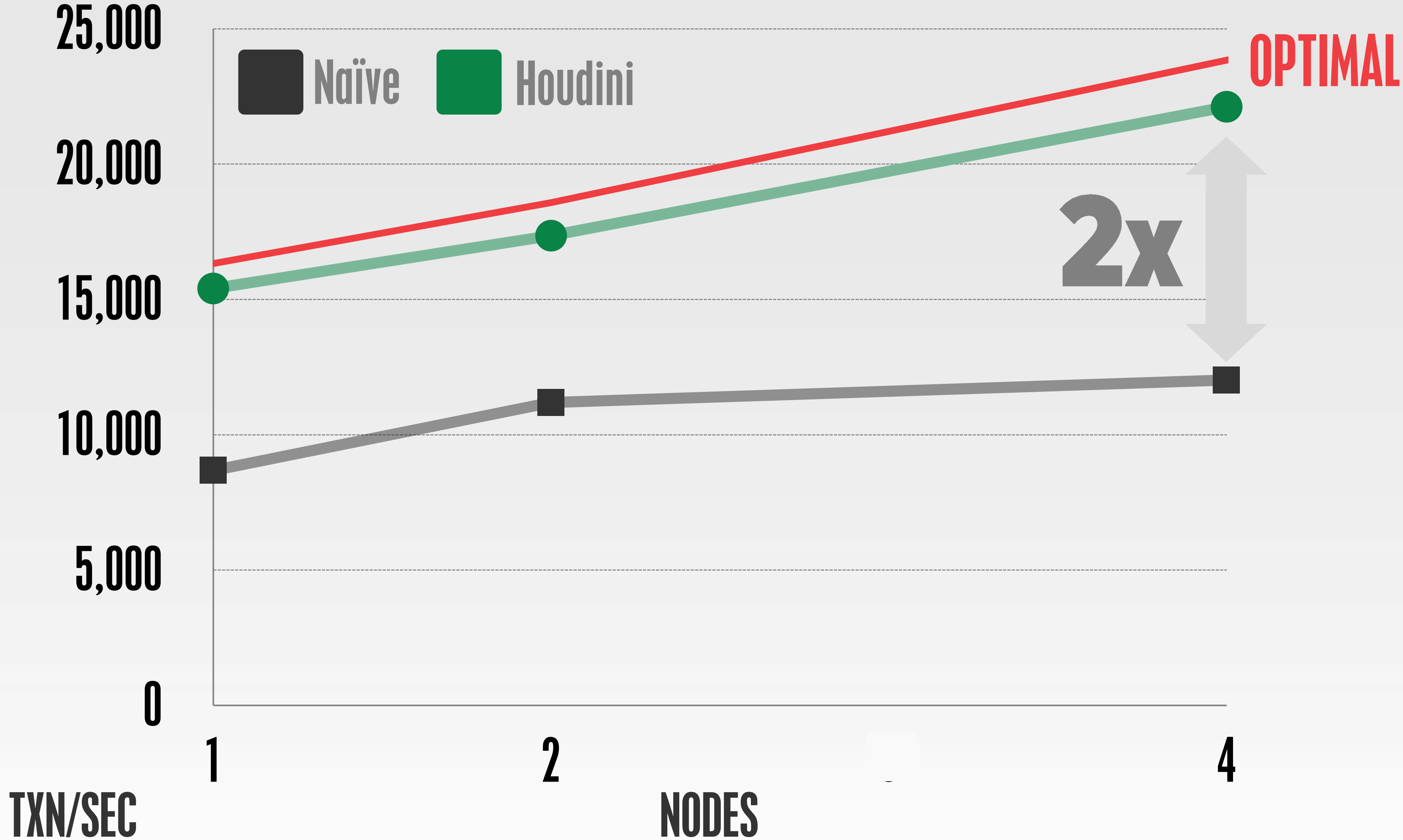
DISTRIBUTED TRANSACTIONS



Houdini

TPC-C BENCHMARK

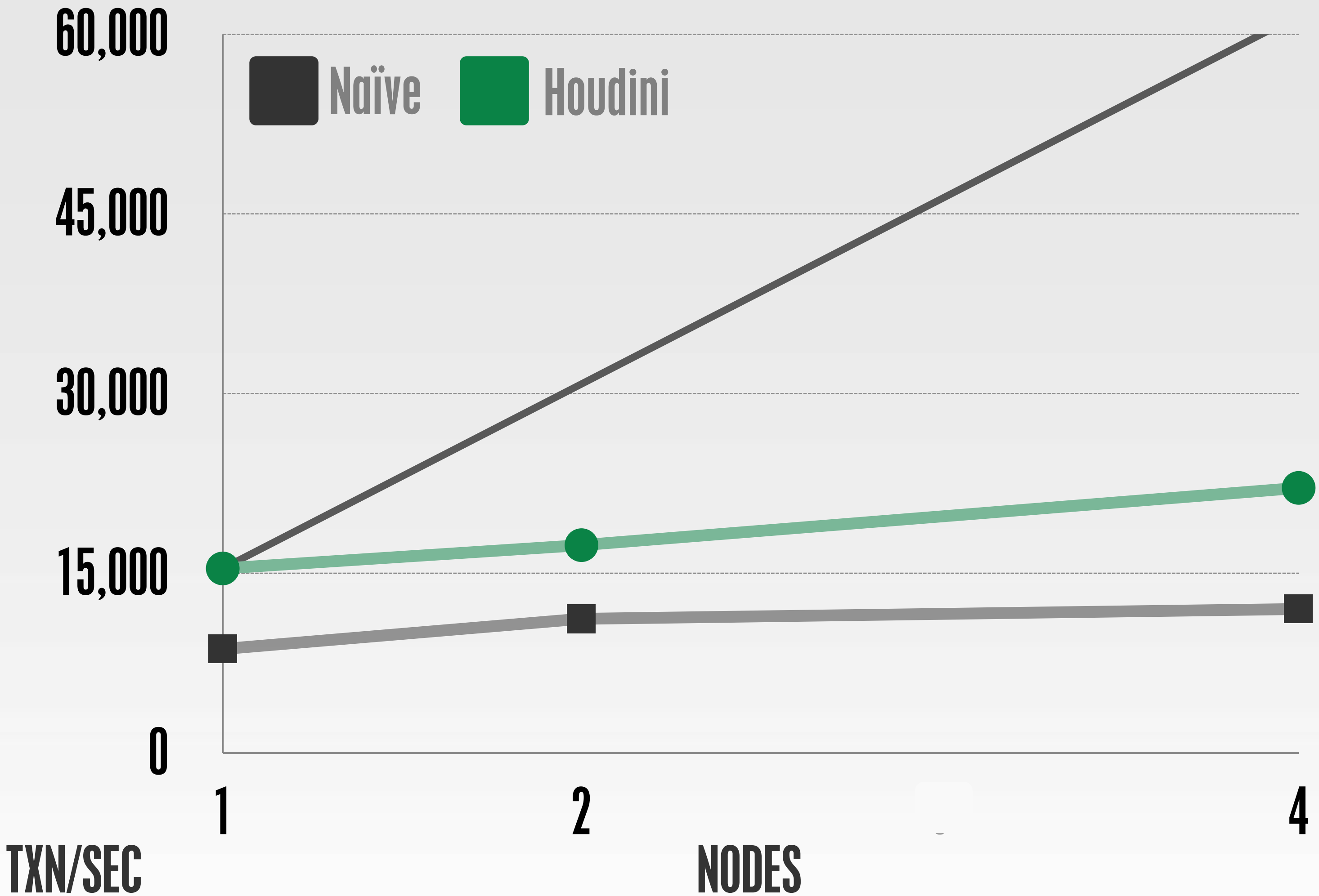
8 Cores per Node
10% Distributed Transactions



Houdini

TPC-C BENCHMARK

8 Cores per Node
10% Distributed Transactions



Houdini

DISTRIBUTED TRANSACTIONS

SP1 - Waiting for Query Result

DISTRIBUTED TRANSACTIONS

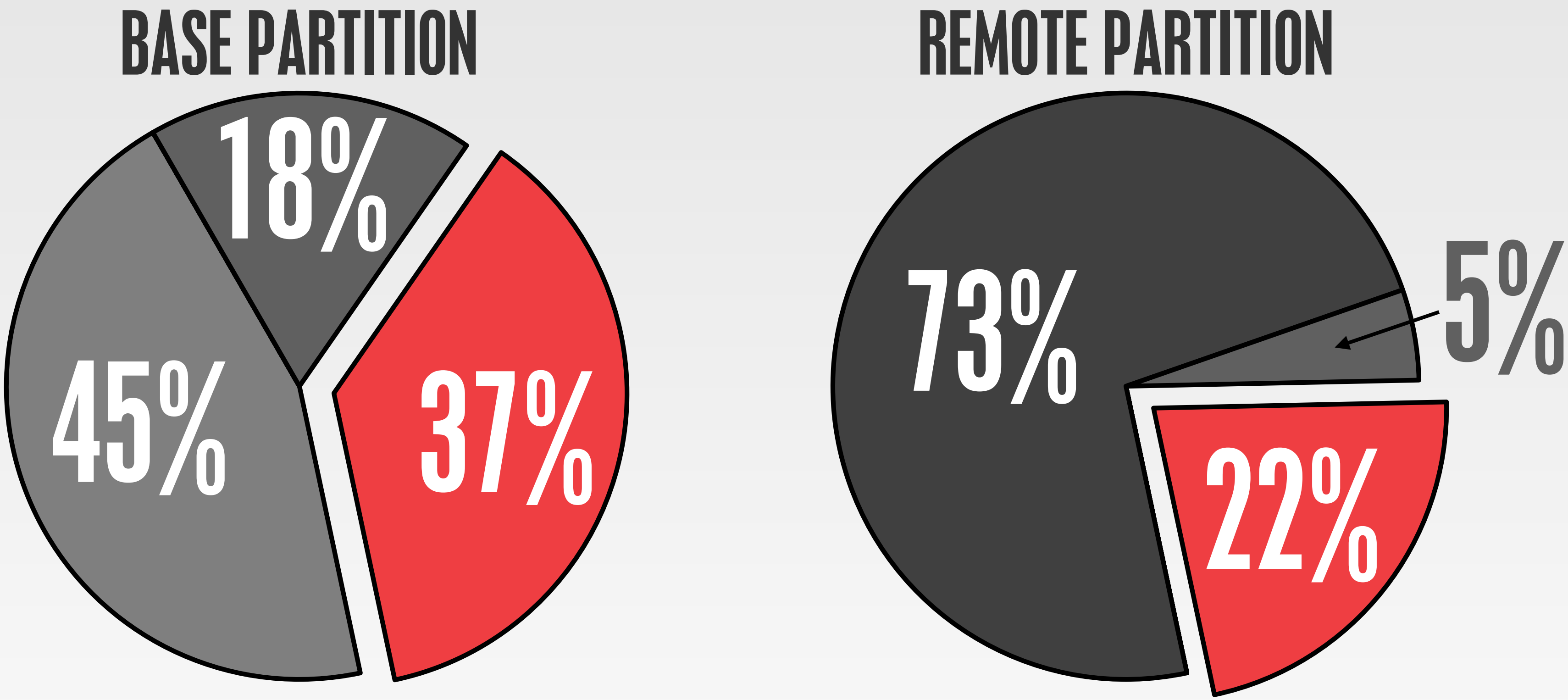
SP1 - Waiting for Query Result SP2 - Waiting for Query Request

DISTRIBUTED TRANSACTIONS

SP1 - Waiting for Query Result   SP2 - Waiting for Query Request

SP3 - Two-Phase Commit

TRANSACTION STALL POINTS



SP1 - Waiting for Query Result

SP2 - Waiting for Query Request

SP3 - Two-Phase Commit

Real Work

DO SOMETHING USEFUL WHEN STALLED


DON'T BE SURPRISED
IF YOU & KB DON'T
LAST THROUGH
GRAD SCHOOL.

DON’T BE STAN’S STUDENT IF YOU GO TO BROWN.

DO USE MACHINE LEARNING TO SCHEDULE SPECULATIVE TASKS.

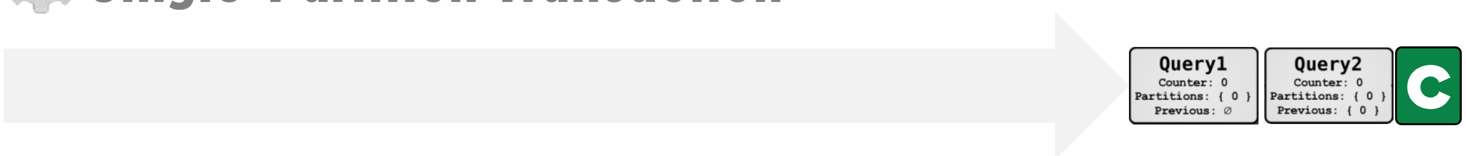
THE ART OF SPECULATIVE EXECUTION

In Progress (August 2013)

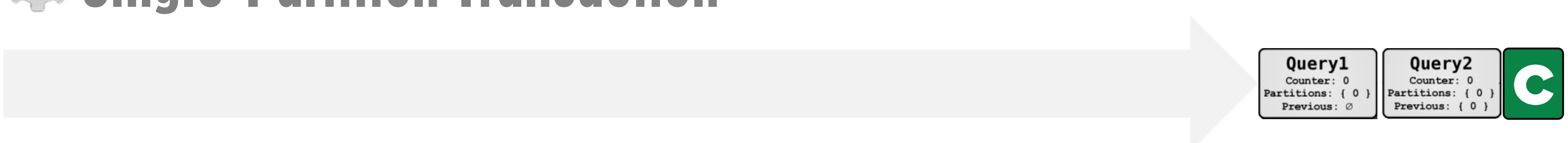
SERIALIZABLE SCHEDULE

Distributed Transaction

Single-Partition Transaction

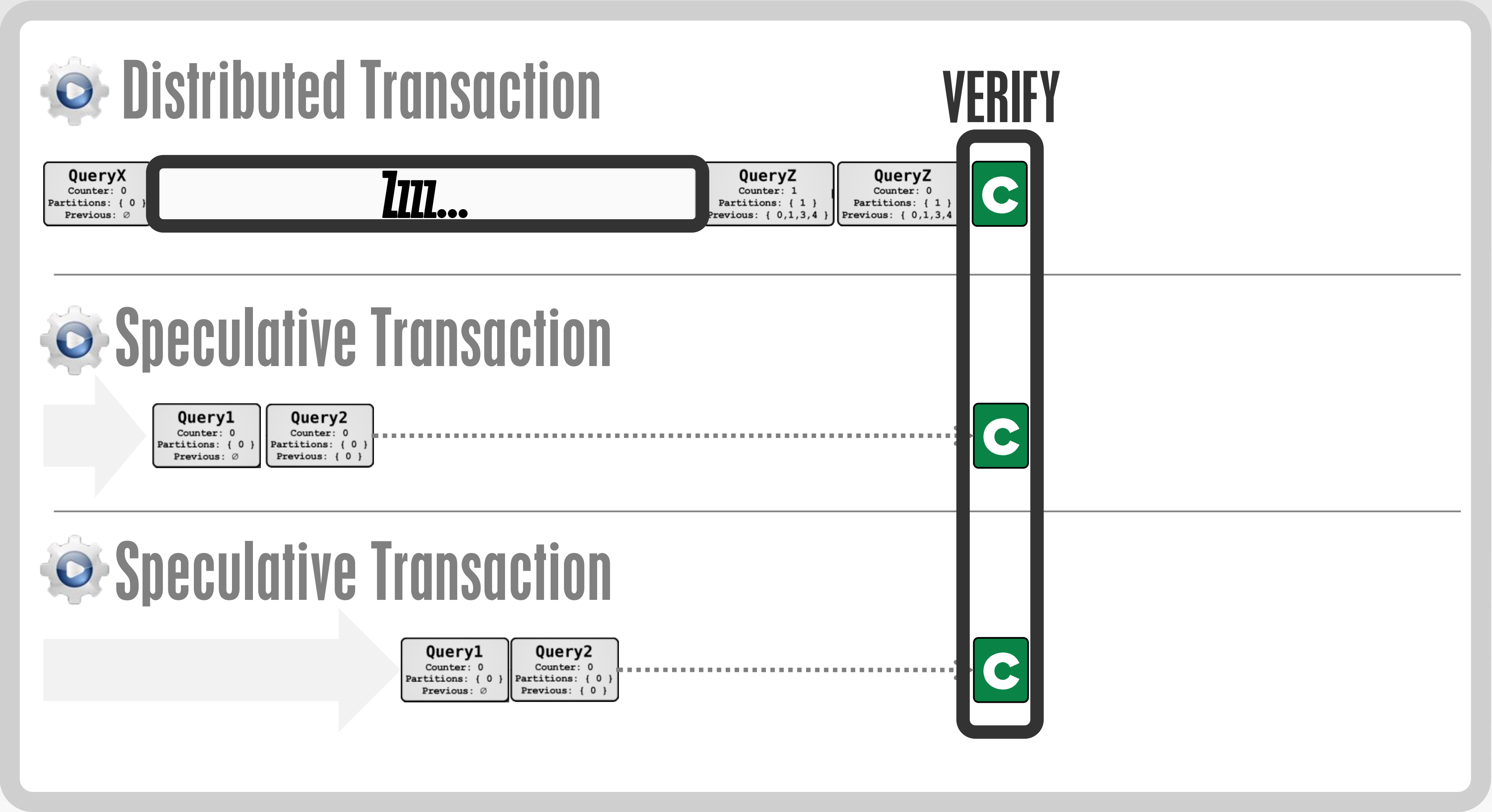


Single-Partition Transaction

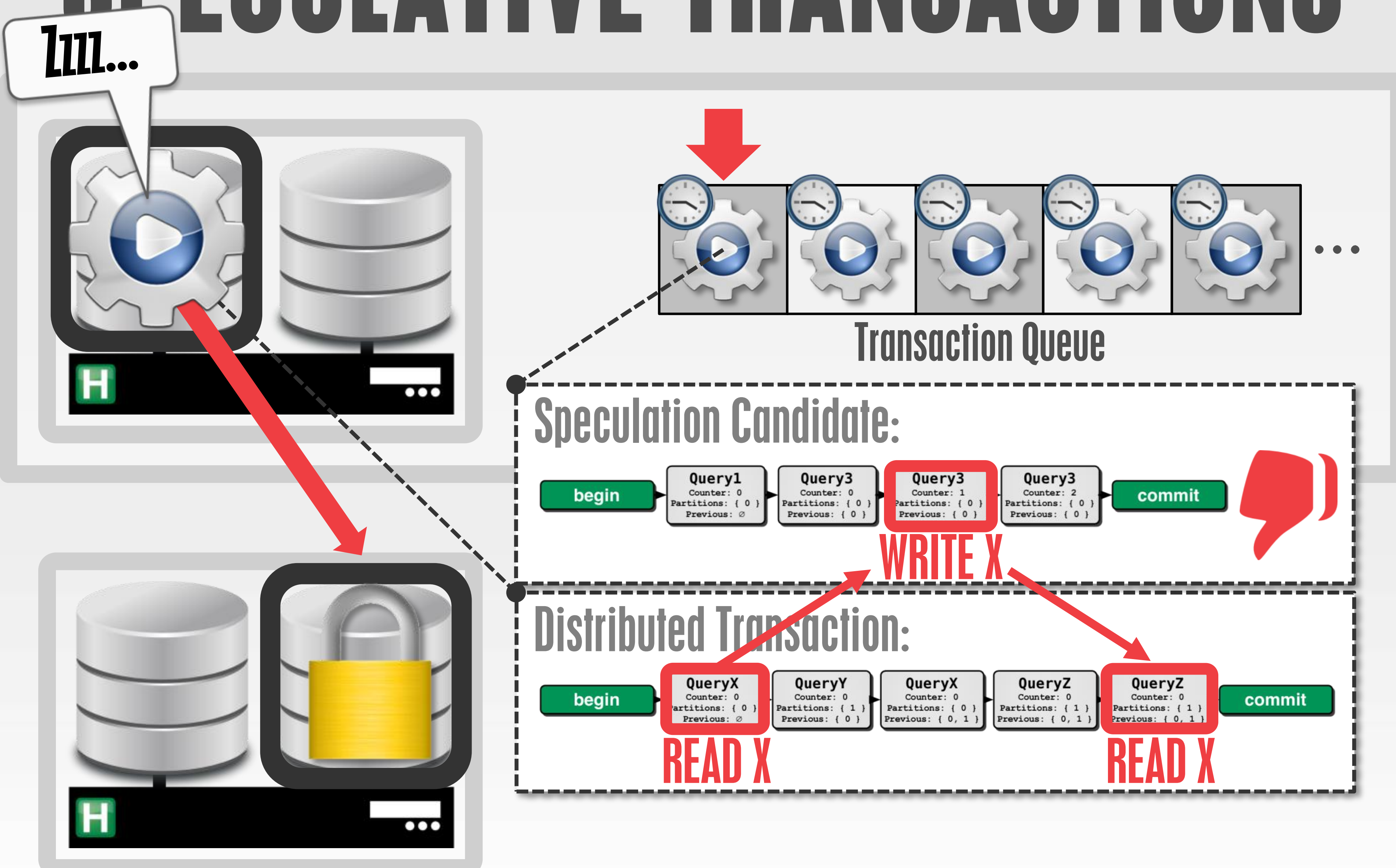


Hermes

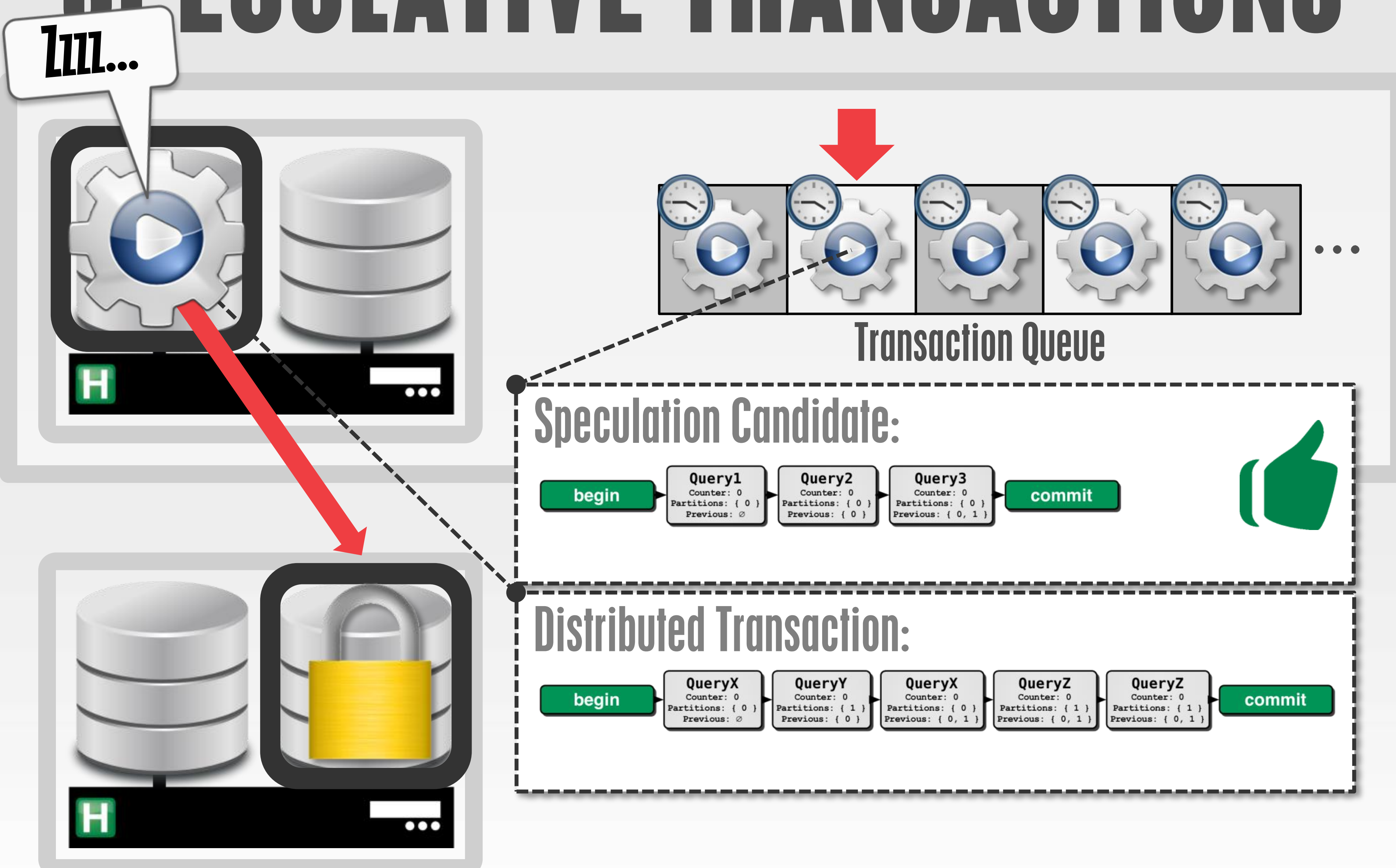
SERIALIZABLE SCHEDULE



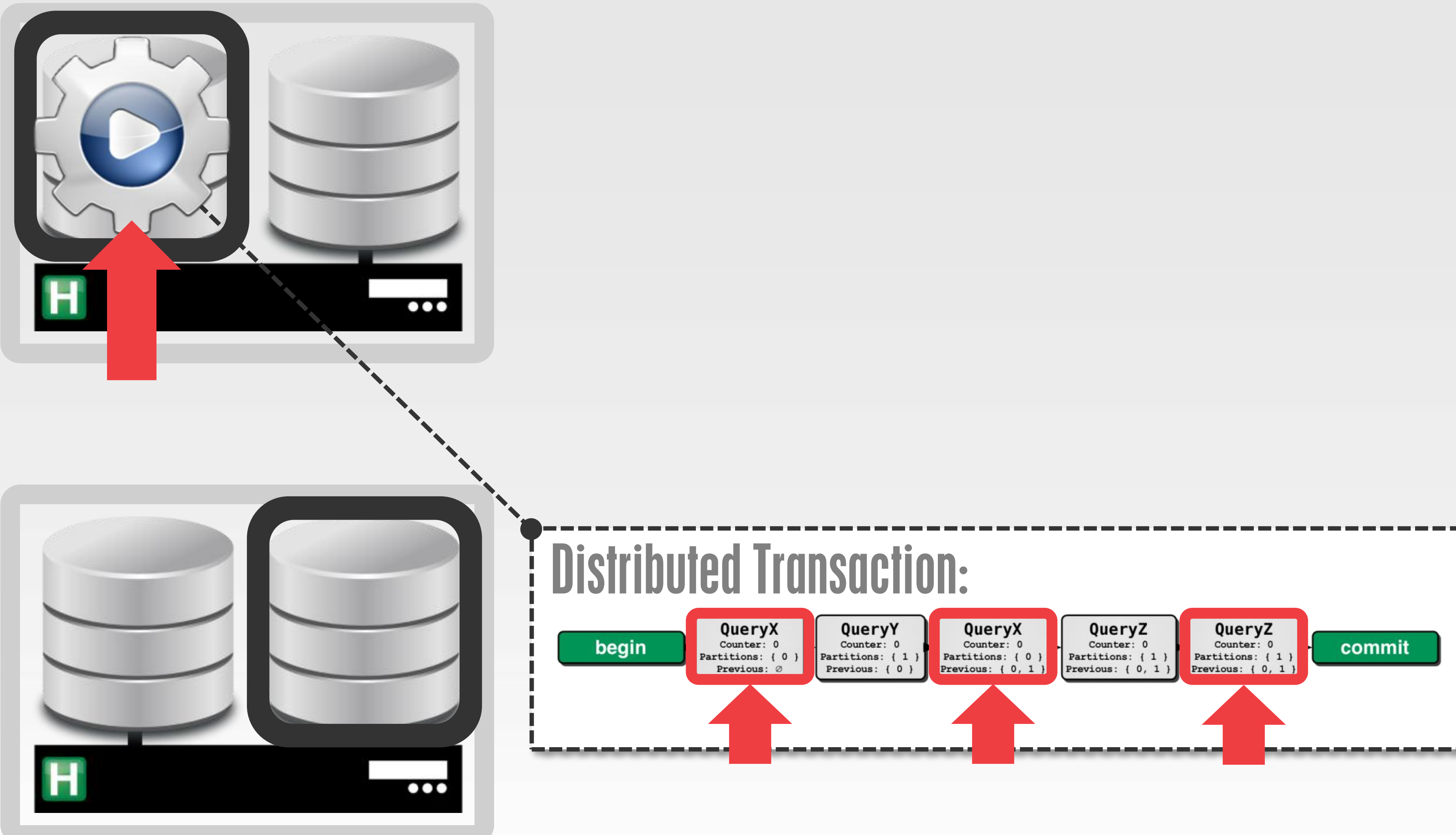
SPECULATIVE TRANSACTIONS



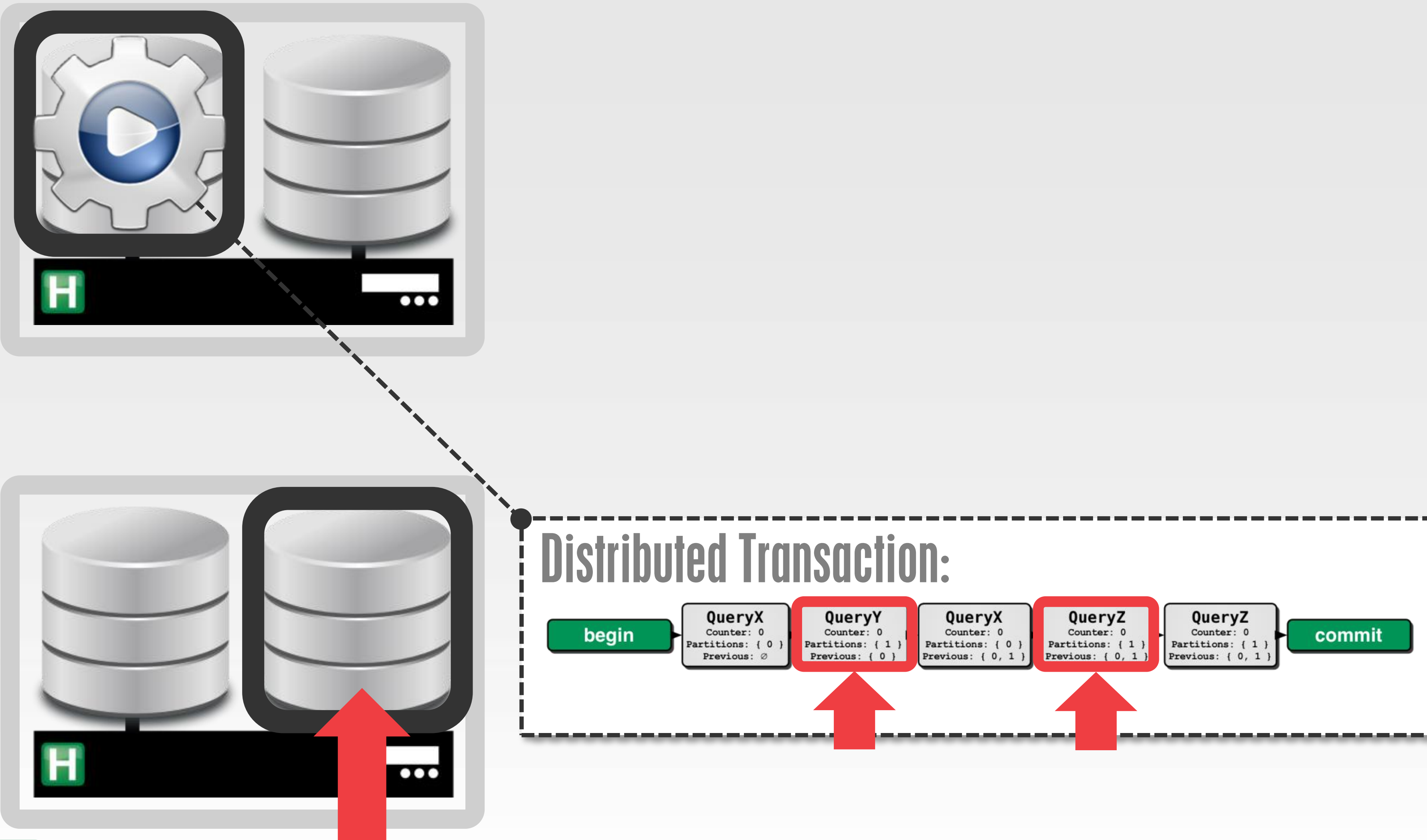
SPECULATIVE TRANSACTIONS



SPECULATIVE QUERIES



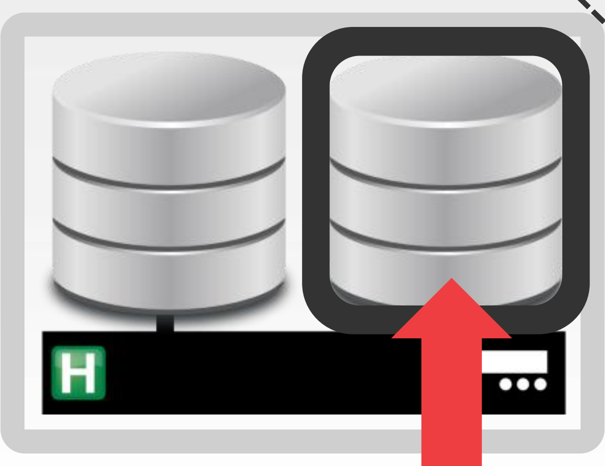
SPECULATIVE QUERIES



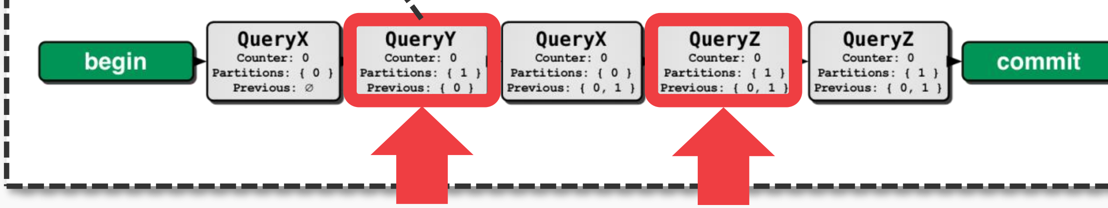
SPECULATIVE QUERIES

QueryY:

```
SELECT S_QTY FROM STOCK
 WHERE S_W_ID = ?
   AND S_I_ID = ?
```

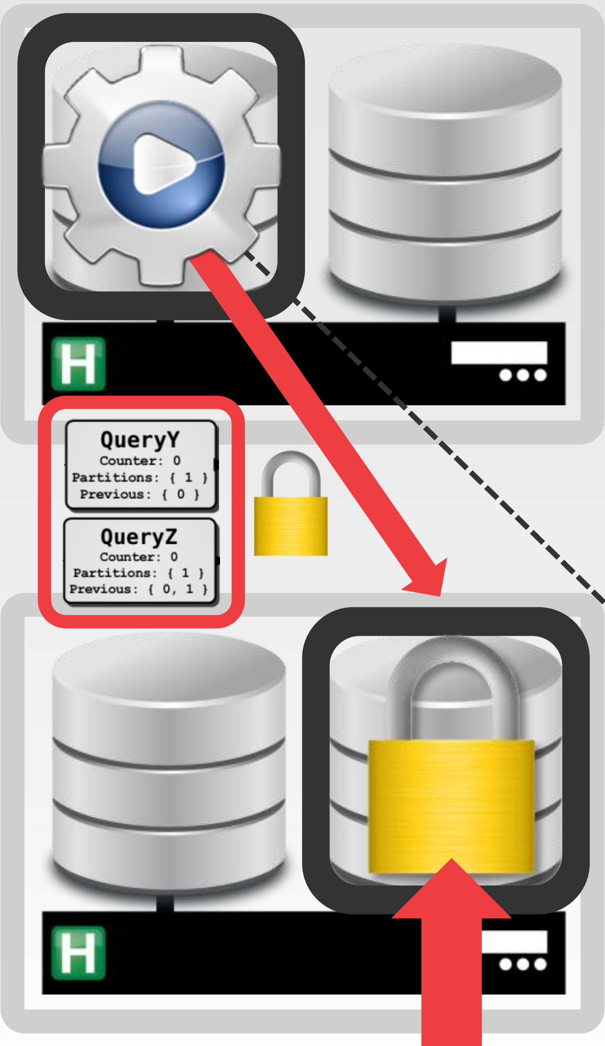


Distributed Transaction:



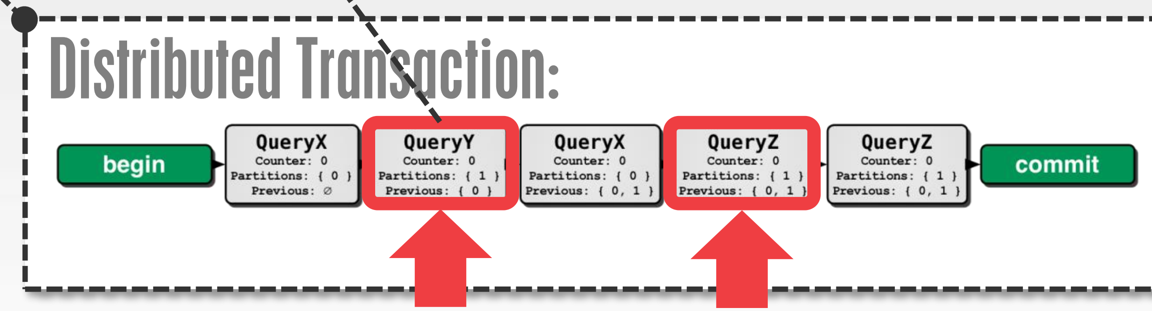
Hermes

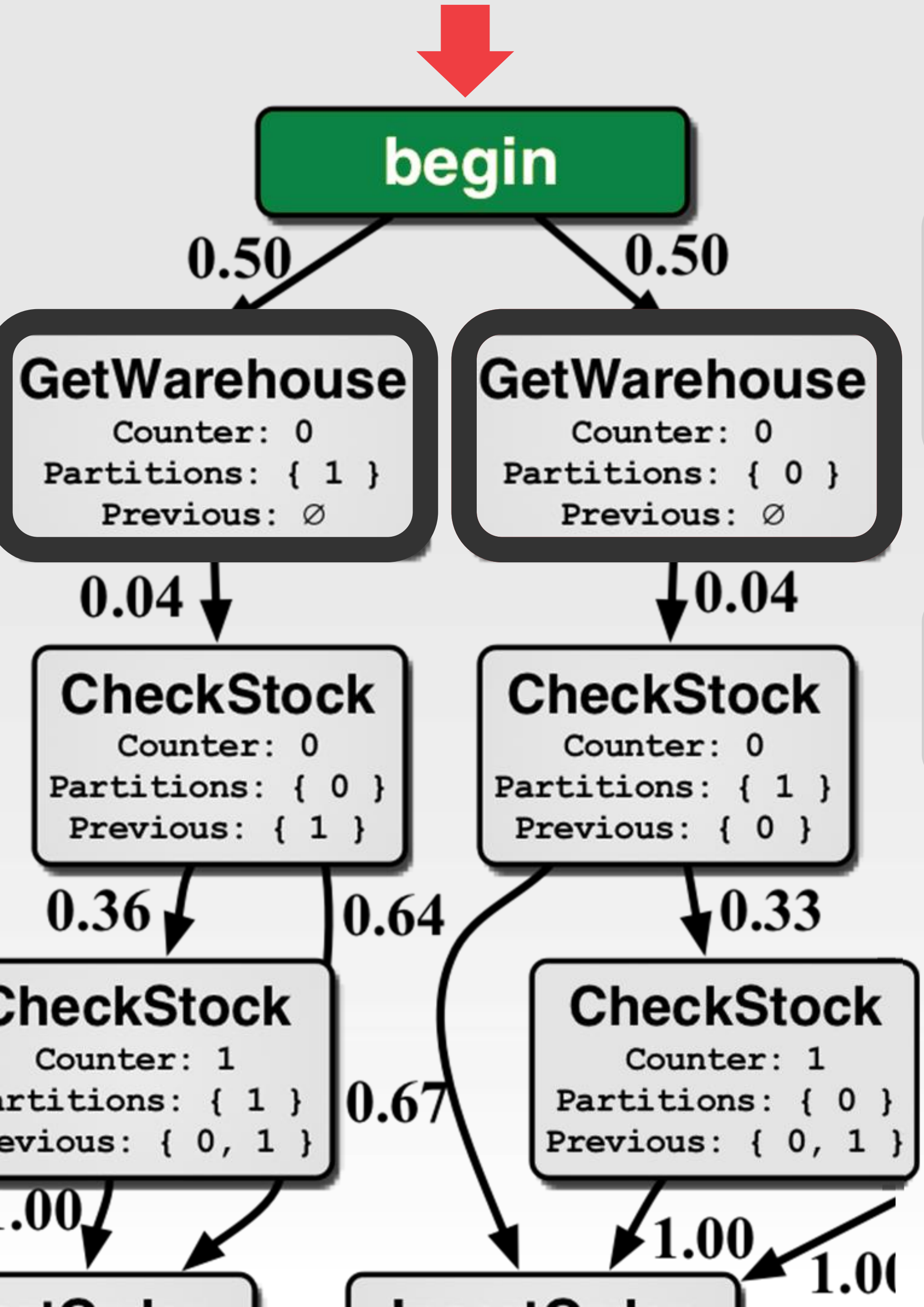
SPECULATIVE QUERIES



QueryY:

```
SELECT S_QTY FROM STOCK
 WHERE S_W_ID = ?
   AND S_I_ID = ?
```



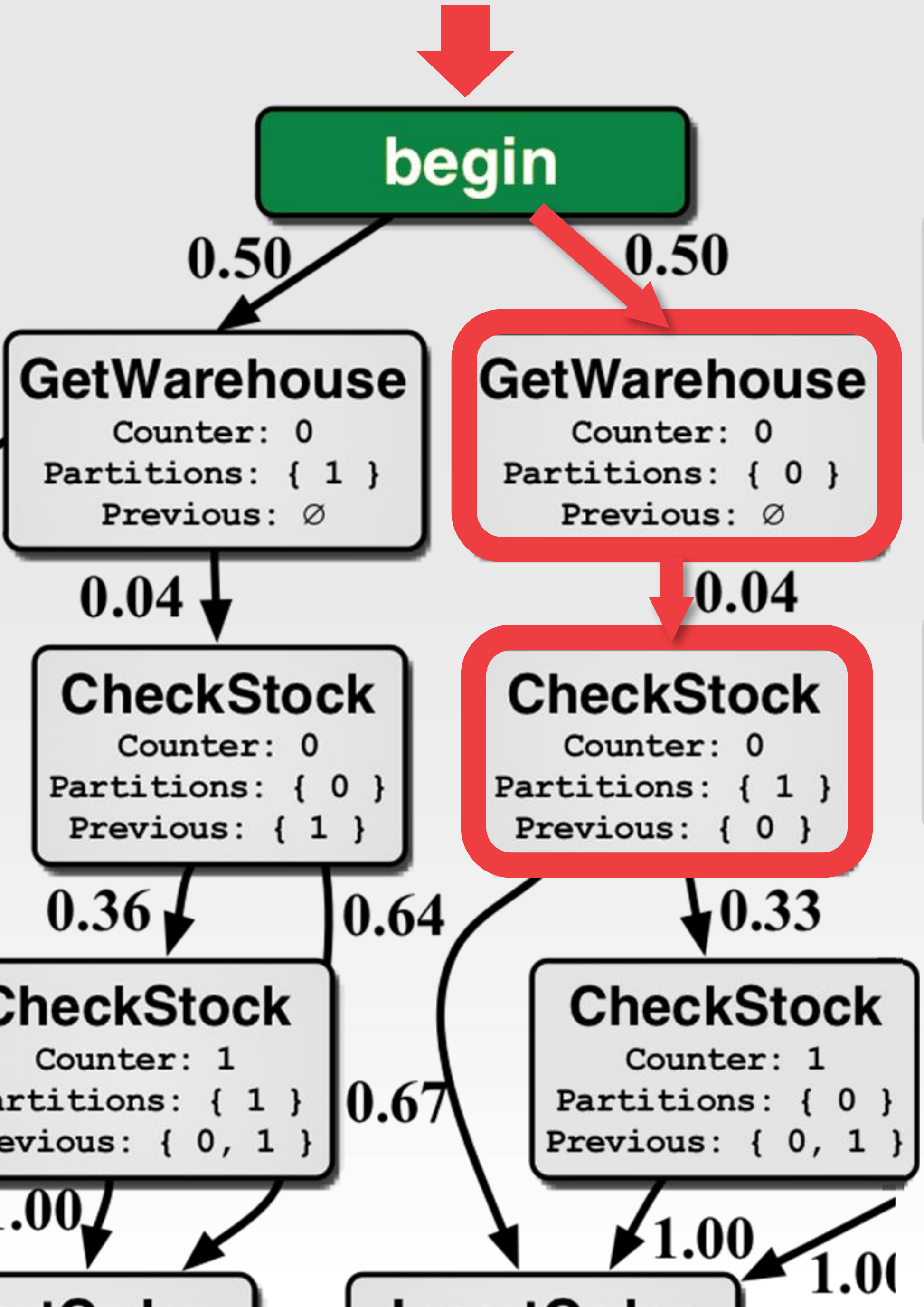


Transaction Parameters:

```
w_id=0
i_w_ids=[1,0]
i_ids=[1001,1002]
```

GetWarehouse:

```
SELECT * FROM WAREHOUSE
 WHERE W_ID = ?
```

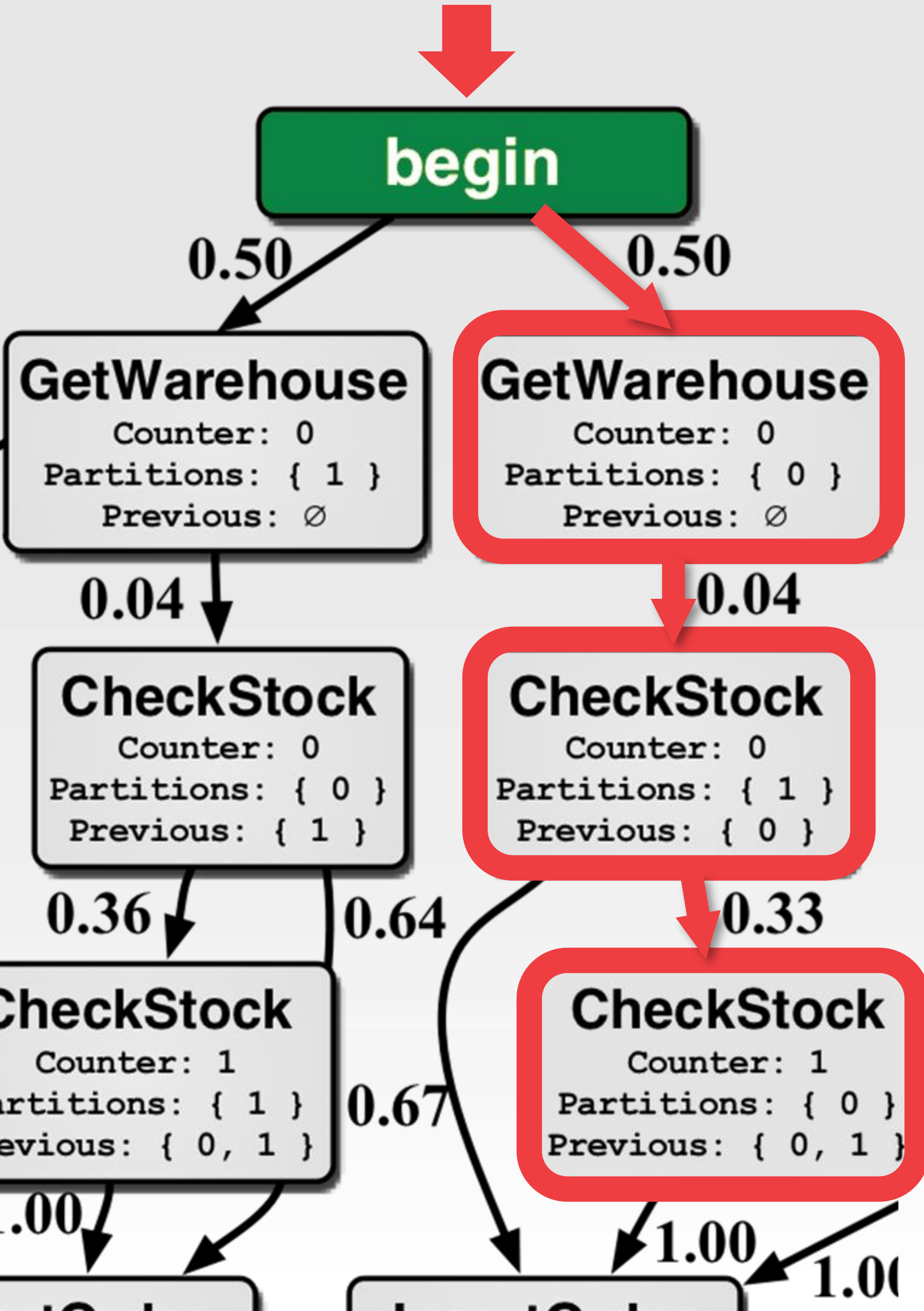


Transaction Parameters:

```
w_id=0
i_w_ids=[1,0]
i_ids=[1001,1002]
```

CheckStock:

```
SELECT S_QTY FROM STOCK
 WHERE S_W_ID = ?
   AND S_I_ID = ?
```



Transaction Parameters:

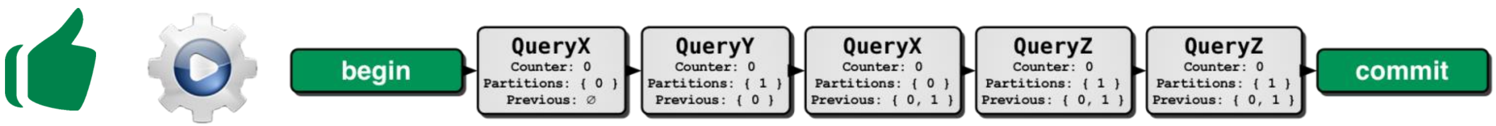
```
w_id=0
i_w_ids=[1,0]
i_ids=[1001,1002]
```

CheckStock:

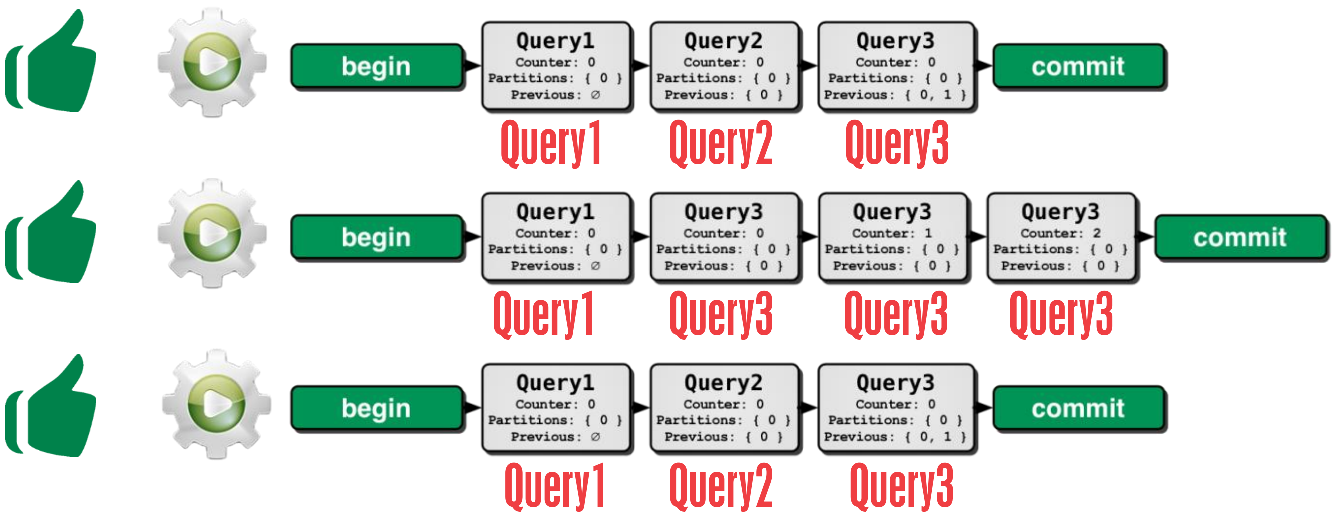
```
SELECT S_QTY FROM STOCK
 WHERE S_W_ID = ?
   AND S_I_ID = ?;
```

VERIFICATION

Distributed Transaction



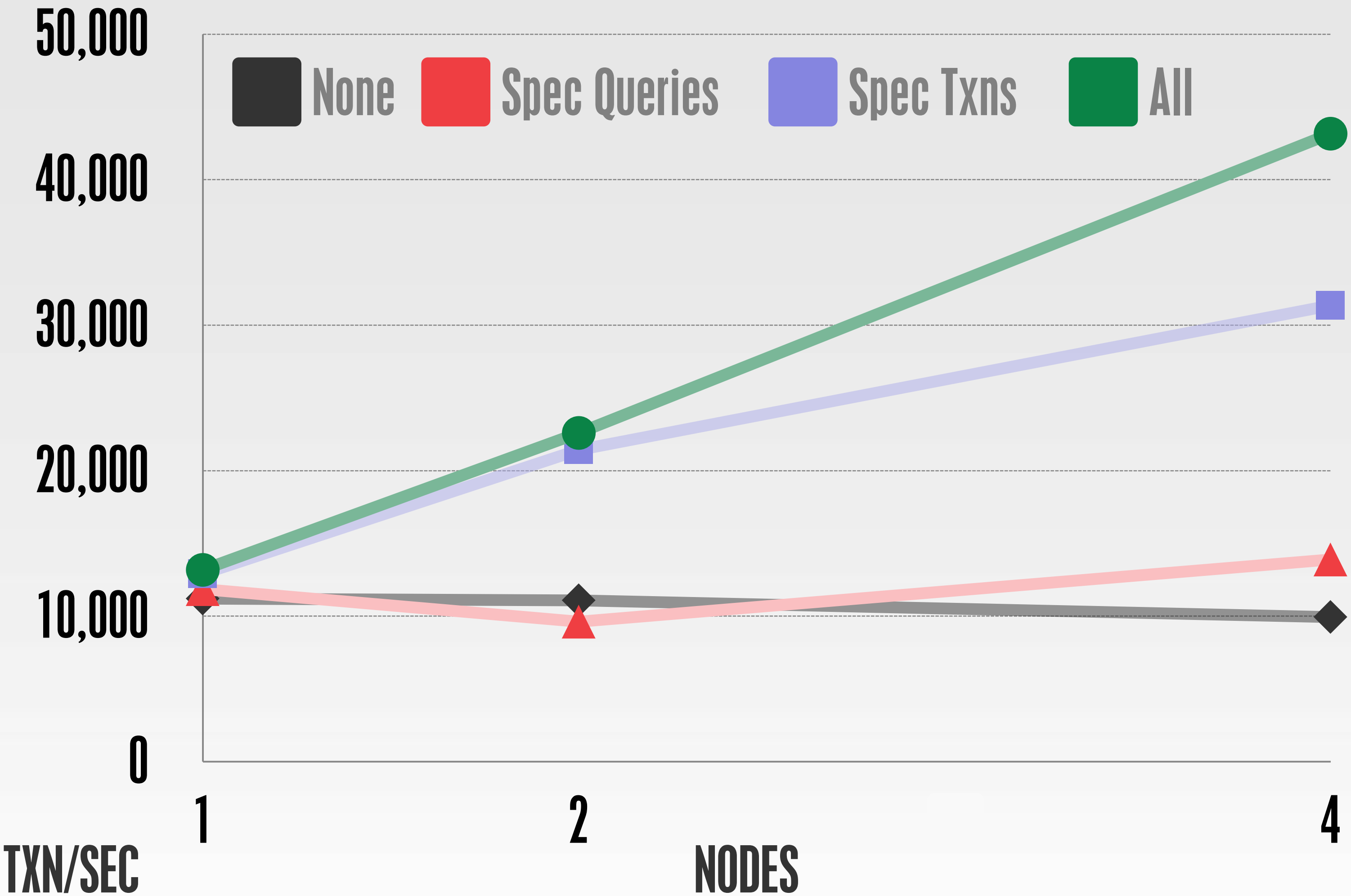
Speculative Transactions



Hermes

TPC-C BENCHMARK

8 Cores per Node
10% Distributed Transactions



Hermes

Optimize Single-Partition Execution

H-STORE: A HIGH-PERFORMANCE, DISTRIBUTED MAIN MEMORY TRANSACTION PROCESSING SYSTEM

Proc. VLDB Endow., vol. 1, iss. 2, pp. 1496-1499, 2008.

Minimize Distributed Transactions

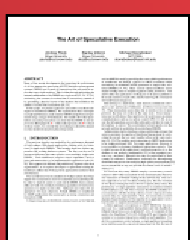
SKEW-AWARE AUTOMATIC DATABASE PARTITIONING IN SHARED-NOTHING, PARALLEL OLTP SYSTEMS

Proceedings of SIGMOD, 2012.

Identify Distributed Transactions

ON PREDICTIVE MODELING FOR OPTIMIZING TRANSACTION EXECUTION IN PARALLEL OLTP SYSTEMS

Proc. VLDB Endow., vol. 5, pp. 85-96, 2011.



Utilize Transaction Stalls

THE ART OF SPECULATIVE EXECUTION

In Progress (August 2013)

FUTURE WORK

One Size **Almost** Fits All™

H-STORE

S-STORE

N-STORE

Escape From Planet
Zdonik

(i.e., Andy Needs to Get Tenure)

Beyond the 'Stores

- Non-Partitionable Workloads.
- The Poor Man's Spanner.
- Scientific Databases.


DON’T MESS IT UP
WITH KB.

Carnegie Mellon University

BROWN Database Group

Stan **Zdonik**

"The Thrill" **Stonebraker**

Sam **Madden**

Ugur **Cetintemel**

David **DeWitt**

Dan **Abadi**

Evan **Jones**

Saurya **Velagapudi**

Xin **Jia**

Carlo **Curino**

Justin **DeBrabant**

Yang **Zou**

Visawee Angkana.

Ning **Shi**

John **Meehan**