References

• The end of an Architectural Era: (It’s Time for a Complete Rewrite), M. Stonebraker et. al. VLDB ’07

• Online documentation: H-Store & VoltDB
VOTER BENCHMARK
Japanese “American Idol”

TRANSACTION
1. Check whether user has already voted.
2. Insert new vote entry.
3. Update vote count for contestant.
VOTER BENCHMARK
Japanese “American Idol”

TXN/SEC

CPU CORES

MySQL
Postgres
Argument from VLDB’07 paper

- Popular DBMSs based on designs from 70’s
- But computer architectures are changing
- And applications have new requirements

- Past 40 years have seen extensions to DBMS design but no major re-design

Discuss requirements from Section 2
SCALABILITY

HIGH
(Many Nodes)

LOW
(One Node)

NO SQL

NEW SQL

TRADITIONAL

WEAK
(Weak/Limited)

GUARANTEES

STRONG
(ACID)
CAN YOU SCALE UP WITHOUT GIVING UP TRANSACTIONS?
Fast  Repetitive  Small
Optimization

USE A LIGHTWEIGHT SYSTEM *DESIGNED* FOR OLTP TRANSACTIONS.
DISK ORIENTED
MAIN MEMORY STORAGE

CONCURRENT EXECUTION
SERIAL EXECUTION

HEAVYWEIGHT RECOVERY
COMPACT LOGGING
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);
}
```
VOTER BENCHMARK
Japanese “American Idol”

MySQL  Postgres  H-Store

TXN/SEC
0  1  2  3  4  5  6  7  8

CPU CORES
1  2  3  4  5  6  7  8

25x
Distributed Transactions

• Discussion based on VLDB’07 paper
Database Partitioning

TPC-C Schema

WAREHOUSE
DISTRICT
CUSTOMER
ORDERS
ITEM
STOCK
ORDER_ITEM

Schema Tree

WAREHOUSE
DISTRICT
CUSTOMER
ORDERS
ORDER_ITEM
STOCK
ITEM

Replicated
Database Partitioning

Schema Tree

Partitions

Replicated
Distributed Transaction Protocol

Procedure Name
Input Parameters

Distributed Transaction Protocol

<Timestamp, Counter, SiteId>

#2084922509960152064

P1

P2

P1

P2

<table>
<thead>
<tr>
<th>#208…</th>
<th>#216…</th>
<th>#229…</th>
<th>#231…</th>
</tr>
</thead>
<tbody>
<tr>
<td>#208…</td>
<td>#229…</td>
<td>#231…</td>
<td></td>
</tr>
</tbody>
</table>

20
Distributed Transaction Protocol

Two-Phase Commit

TransactionPrepare Request
TransactionPrepare Response
TransactionInit Request
TransactionInit Response
TransactionFinish Request
TransactionFinish Response