The End of an Architectural Era (It's Time for a Complete Rewrite)

Michael Stonebraker
Stavros Harizopoulos
Samuel Madden
Nabil Hachem
Daniel Abadi
Pat Helland

Paper presentation:
Craig Hawkins
craig_hawkins@brown.edu
February 23, 2015

2007

An emergent time...

1

2

3

4

5

1:www.apple.com (Jan 2007 announcement)
2:www.toyota.com (mainstream ~ 2007)
3:www.intel.com (rollout late 2006 - early 2007)
4:www.wikipedia.org (worldwide rollout jan 2007.. 32 and 64 bit editions ...mac >> intel chips 2006)
5: www.facebook.com

While all of this is happening,

there is unrest in the database forest... [1]

1: phrase adapted from *Trees*, by the music group Rush

Meet the new boss:

1

(Same as the old boss)

2

Michael Stonebraker

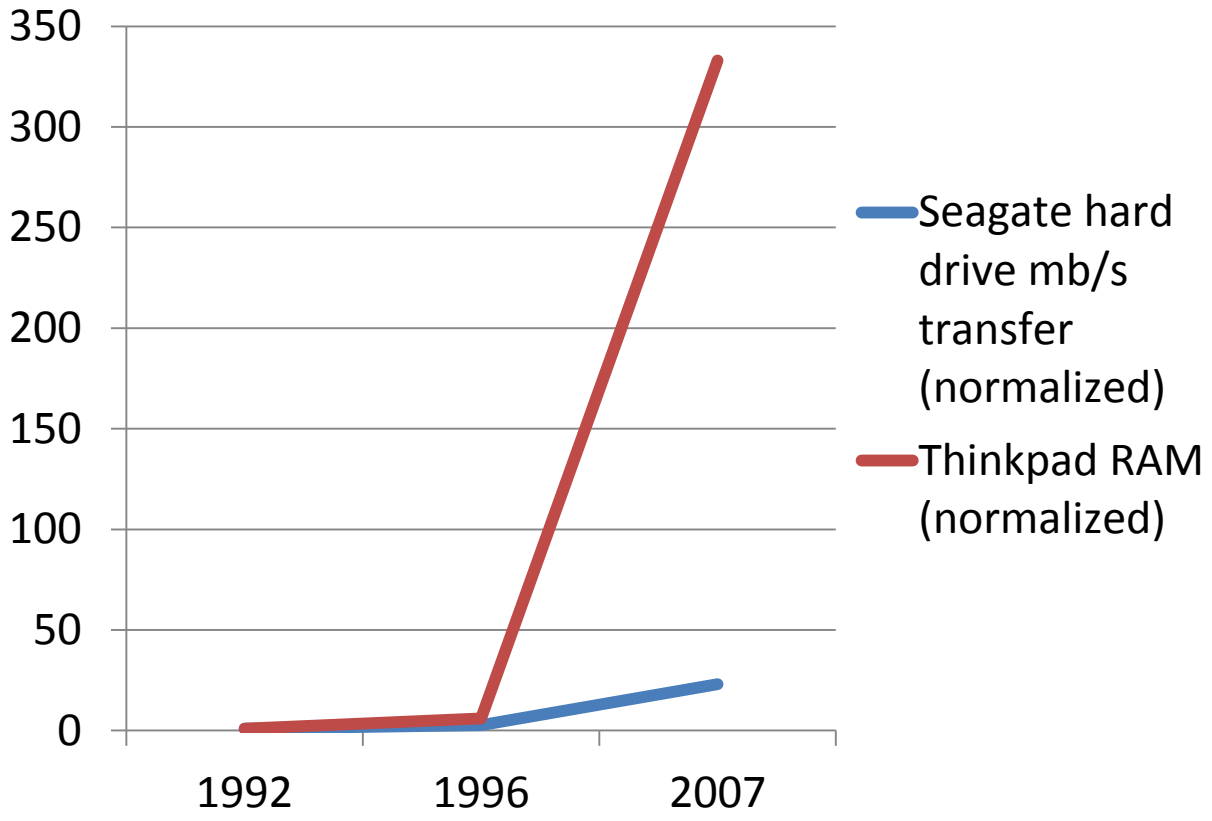
Princeton, Michigan

U.C. Berkeley, M.I.T.

Ingres, Postgres,
Vertica, VoltDB
H-Store, Aurora

1: www.wikipedia.org
2: lyrics from *Won't Get Fooled Again*, by the music group The Who
It's no longer a one-size-fits-all database world.

The rationale:



sources:
www.wikipedia.org
cs.berkeley.edu
http://www.eetimes.com/document.asp?doc_id=1209142
http://oldcomputers.net/ibm-thinkpad.html
http://www.seagate.com/staticfiles/support/disc/manuals/enterprise/cheetah/15K.5/SAS/100384784c.pdf

Key aspects of H-Store...

H-Store Architecture

A shared-nothing grid of computers.

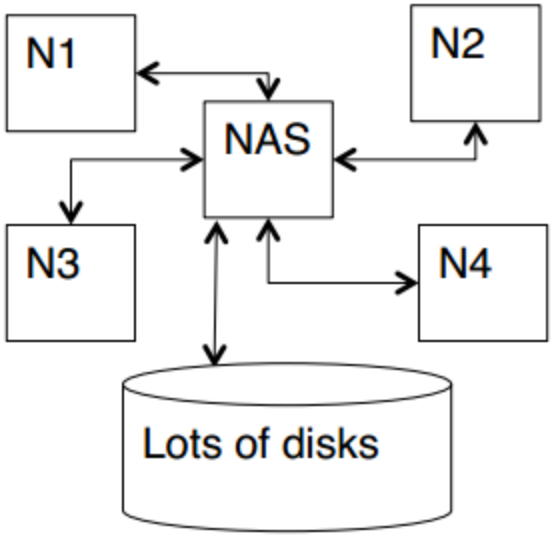
Each H-Store site is single-threaded, and performs incoming SQL commands to completion, without interruption.

Each site is decomposed into a number of single-threaded logical sites, one for each core; each logical site gets its own partition of the main memory. Each node manages one or more "partitions" i.e. disjoint sets of data.

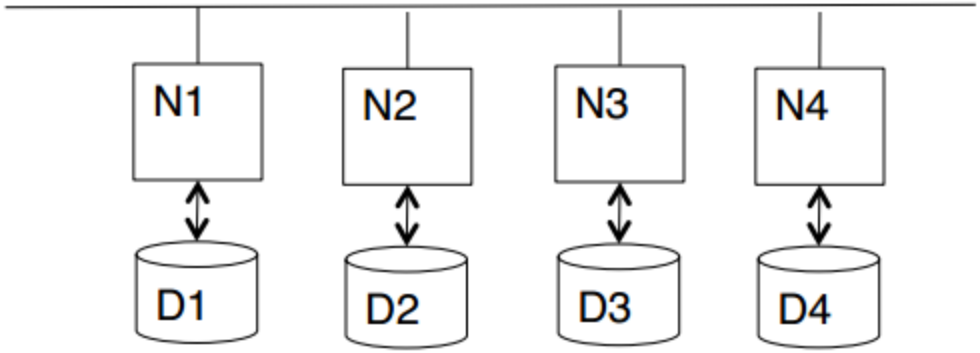
Transactions may be initiated from any site on the grid.
(VLDB paper, pg 5).

A shared nothing architecture (SN) is a distributed computing architecture in which each node is independent and self-sufficient, and there is no single point of contention across the system. More specifically, none of the nodes share memory or disk storage.
Shared-nothing architectures are highly scalable, Google et. al.
www.wikipedia.org

Shared-Disk



Shared Nothing



1

1. http://cs.brown.edu/courses/csci1270/files/lectures/L19_ParallelDBs_1.pdf
NAS: "network-attached storage" , RAID, etc.
contention/coordination doesn't scale well... the mythical man month
control node, non-standardized middleware, erlang, go

A single-threaded execution model

Advantage:
Simplification:
No need for multi-threaded data structures or concurrent B-trees

Disadvantage:
Not good for long-running, ad-hoc commands.

However, this does not happen often in OLTP environments.

An in-memory data store

Useful transaction work is completed in under 1 millisecond.

Factor of 82 speed up on TPC-C benchmark.

1

1. 2007 VLDB Conference, *The End of an Architectural Era*, pgs 1,3
TPCC: Transaction Processing Performance Council

The complete H-Store workload is specified in advance as a collection of transaction classes.

1

1. 2007 VLDB Conference, *The End of an Architectural Era*, pg 4
some of these classes perform better than others

Optimal classes:

Transactions that can be completed entirely at a single node (i.e. *single sited*) can run without stalls. They can also run without much intra-node communication.

1. 2007 VLDB Conference, *The End of an Architectural Era*, pg 4

Optimal classes:

One-shot applications: all of their transactions can be executed in parallel with little or no intra-node communication.

Each transaction can be decomposed into a collection of single-site plans

1. 2007 VLDB Conference, *The End of an Architectural Era*, pg 4
~divide and conquer approach

H-Store's "automatic physical database designer":

Specify horizontal partitioning, replication locations, and indexing that maximizes the likelihood of transactions being single-sited.

1. 2007 VLDB Conference, *The End of an Architectural Era*, pg 4
other aspects: two-phase, strongly two-phase, commutable, sterile

The complete workload is specified in advance as a collection of transaction classes.[1]

SQL statements must be executed as stored procedures.

1. 2007 VLDB Conference, *The End of an Architectural Era*, pg 4

What on

1

is a stored procedure?

1. www.nasa.gov

A stored procedure is a named, parameterized SQL routine.

It is stored in the database as part of the schema.

What are some of its advantages:

Reduced network congestion

Security: permissioning abstraction, indirect user-access to tables

DBMS *can* optimize / compile once rather than each time.

What are some of its disadvantages:

Increased programmer sophistication

Varying vendor implementation

DB admin. overhead:
1000 tables X 4 CRUD = 4000
prepared statements.

How do you create one:

```
CREATE PROCEDURE get_balance(IN customerID CHAR(10))
SELECT balance
FROM accounts
WHERE id = customerID;
```

1. www.nasa.gov

How do you use one:

CALL get_balance(1004200011);

```
cs = this.con.prepareCall("{call SHOW_SUPPLIERS()}");
ResultSet rs = cs.executeQuery();
while (rs.next()) {
    String supplier = rs.getString("SUP_NAME");
    String coffee = rs.getString("COF_NAME");
    System.out.println(supplier + ": " + coffee);
}
```

1. http://docs.oracle.com/javase/tutorial/jdbc/basics/storedprocedures.html#calling_javadb_mysql
Java DB with mySQL example

H-Store's

Transaction management...

Each transaction receives a timestamp on entry into H-Store:

site_id

local_unique_timestamp

1

transactions are executed in timestamp order

clocks need to be nearly in sync

1. 2007 VLDB Conference, *The End of an Architectural Era*, pg 6
clock sync vis-a-vis NTP: Network Time Protocol
sterile transcactions: fully "commutable" H-Store classes. order doesn't matter... don't have to wait
otherwise wait a small amount to account for network delays, replication, clock variance...infiniband
all replicas update in the same order

H-Store takes an "optimistic" approach

Favors aborts over heavy locking and latch wait schemes

Uses a transaction coordinator, and a transaction monitor.

locks, rows. latch waits: protecting a block's integrity when moving in/out of disk

transaction coordinator: at arrival site. sends out subplan pieces to "worker sites".
when gets "ok" from all sites, proceeds to next collection of subplans.
when no more subplans...commits, otherwise aborts. (VLDB 2007 paper: pg 6)
small time stalls placed in to see if a subplan with an earlier timestamp appears (VLDB 2007 paper: pg 6)

transaction monitor: if notices too many aborts, will adjust plan (VLDB 2007 paper: pg 6)
OLTP: fast insert, update, delete

Where's the disk?

1

1. www.wikipedia.org

Data persistence is achieved through snapshots and a command log.

Snapshot: each DBMS node continuously write asynchronous snapshots of the entire database to disk at fixed intervals.

Command log: in-between snapshots, a disk-based command log records the successful transactions only. Interestingly, the log contains the *transactions*, not record-level logging.

ACID: client can't see effects of a transaction until it gets recorded in the command log.

Optimizations: batch processing of command log writes, delta updates for snapshotting

(VLDB 2013 paper pgs 3, 6)

Major drawback:

What happens when you run out of main memory?

This brings us to....

Anti-caching!

Go Christian!