AsterixDB
more engine less trunk

Big Data Storage:

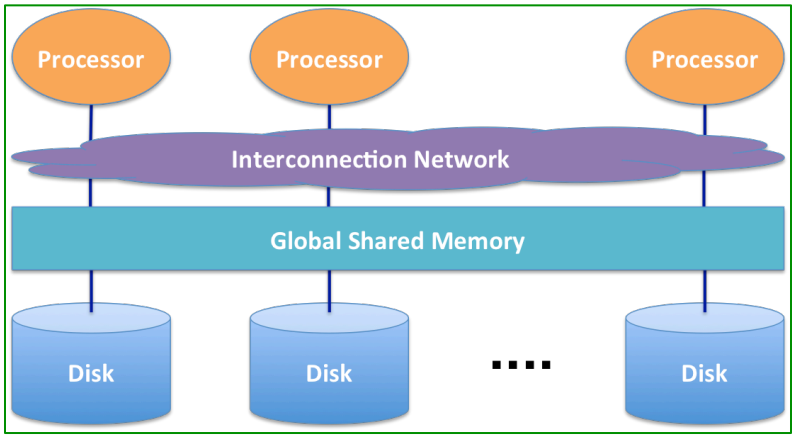
Excuse me, but –

Where can I keep my bits...?!?

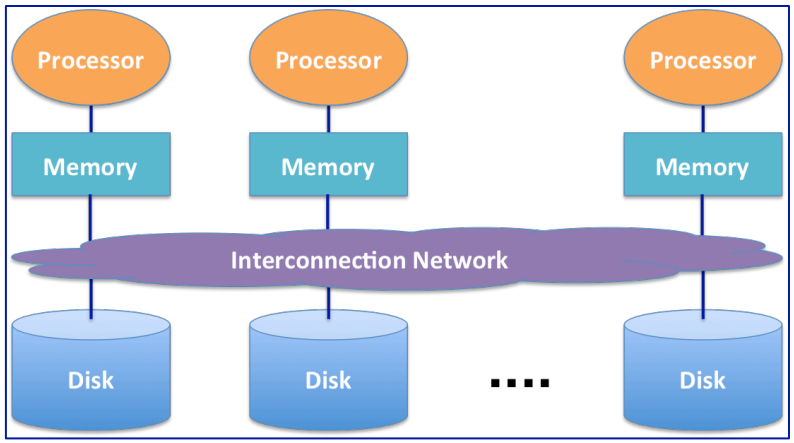
http://isg.ics.uci.edu

Michael Carey
University of California, Irvine

DB History: *Shared What?*

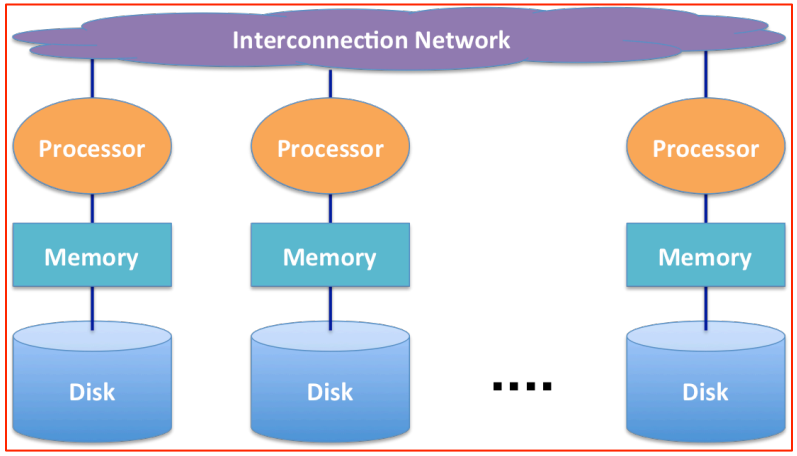


Shared-everything



Shared-disk

Shared-nothing

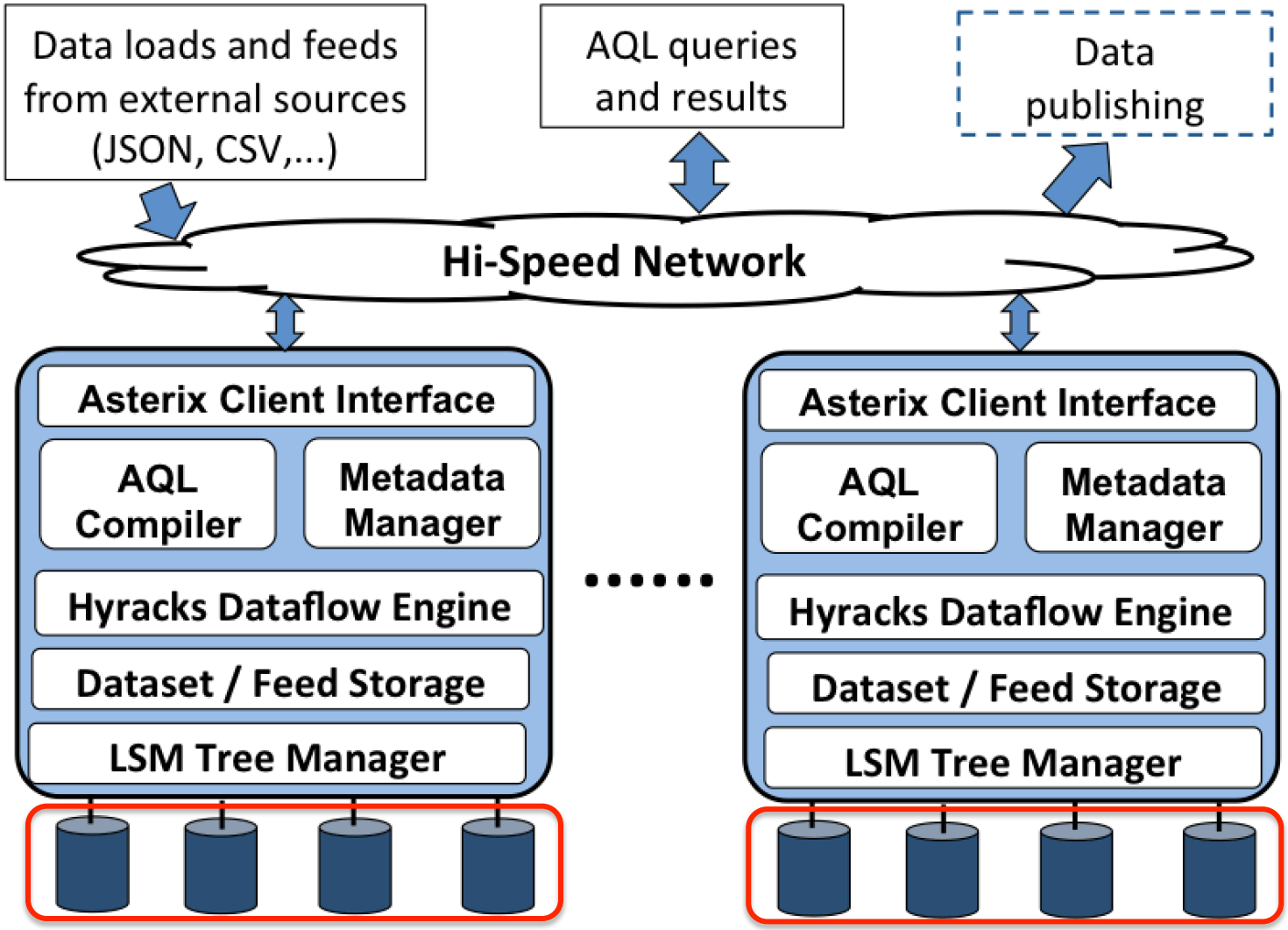


?

(1980's)

UCIrvine
http://isg.ics.uci.edu

AsterixDB (Cartoon View)

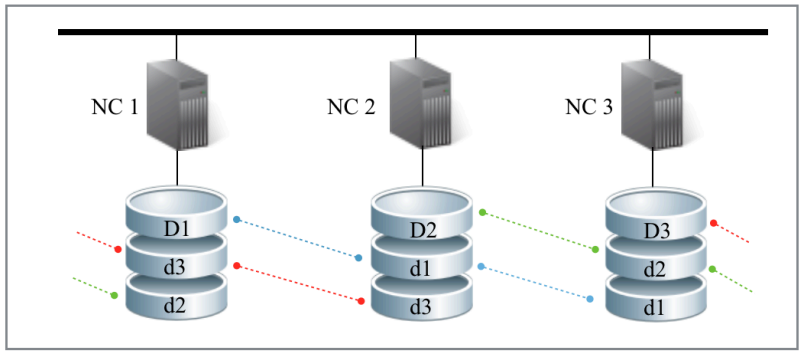


UCIrvine
http://isg.ics.uci.edu

Distributed Storage in AsterixDB

- Hash-partitioned, shared-nothing, *local drives*
  - Partitioning based on primary key (hashing)
  - Secondary indexes local to, and consistent with, corresponding primary partitions (all ***LSM-based***)
- Also support external datasets (for HDFS)
  - Multiple (Hive) formats, *secondary index support*
  - Index partitions *co-located* with data (if possible)
  - Developed to save space and offer "IT comfort"

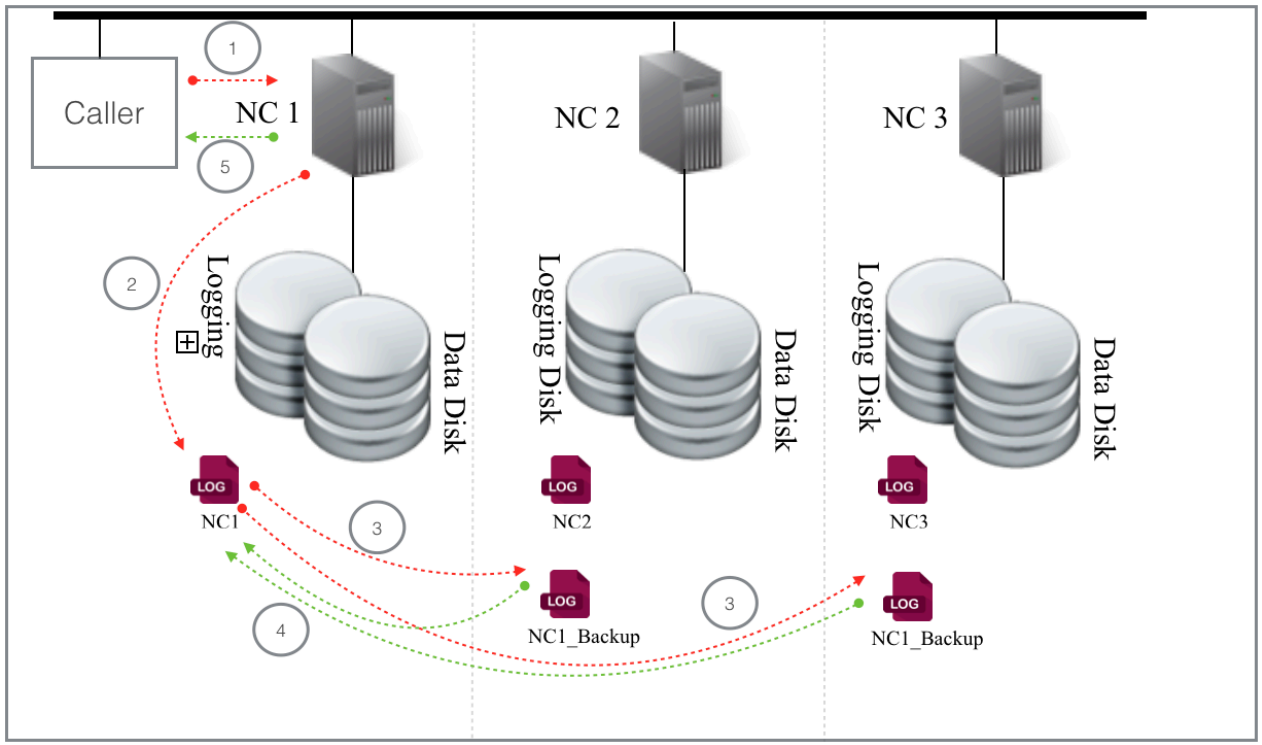
AsterixDB Data Replication (*WIP*)



Chained Declustering

Log-Based Replication

(synchronous, recovery-only copies kept)



UCIrvine
http://isg.ics.uci.edu

Where Should My Bits Go...?

- Computing may be transient and/or elastic – but accumulated ***data*** is ***not***...!!!
  - Native storage → hard to expand and contract!
  - Calls for an SD approach based on HPC (or cloud) storage facilities
  - Obviously workload-dependent (e.g., queries and/or analytics, Big ML, Big Science, ...)
- Serious experimentation is needed (IMO)
  - E.g., SAN-based HPC architectures?
  - E.g., Google persistent disks (in Google Cloud)?
  - Performance implications interesting to explore...

Where Should My Bits Go...?

Ex: SDSC's Gordon cluster...

- Dedicated cluster with 1024 compute nodes and 64 I/O nodes.
- Each compute node contains two 8-core 2.6 GHz Intel EM64T Xeon E5 (Sandy Bridge) processors and 64 GB of DDR3-1333 memory.
- Each I/O node contains two 6-core 2.67 GHz Intel X5650 (Westmere) processors, 48 GB of DDR3-1333 memory, and sixteen 300 GB Intel 710 solid state drives.
- Network is a 4x4x4 3D torus with adjacent switches connected by three 4x QDR InfiniBand links (120 Gbit/s). Compute nodes (16 per switch) and I/O nodes (1 per switch) are connected to the switches by 4x QDR (40 Gbit/s).
- Theoretical peak performance is 341 TFlop/s.

Hedging Our AsterixDB Bets

- Currently porting our LSM-based storage to also work *on top* of HDFS (and YARN)
  - Might somehow feel more "comforting" (and/or "environmentally friendly") to Big Data IT shops
  - A different path to replication & high availability
- Interesting experiments lie ahead!
  - Revisit Stonebraker-like OS issues (today's version)
  - Bake-off: Distributed record management *vs.* DFS, local versus remotely attached storage, ...
  - E.g., how well does HDFS do *w.r.t.* locality of writes?

Is History Repeating Itself?

Operating System Support for Database Management

Michael Stonebraker
University of California, Berkeley

Communications
of
the ACM

July 1981
Volume 24
Number 7

ting system services are examined
pplicability to support of database
hese services include buffer pool
em; scheduling, process manage-
communication; and consistency

7. Conclusions

The bottom line is that operating system services in many existing systems are either too slow or inappropriate. Current DBMSs usually provide their own and make little or no use of those offered by the operating system. It is important that future operating system designers become more sensitive to DBMS needs.

UCIrvine
http://isg.ics.uci.edu