CROSS | CENTER FOR RESEARCH IN OPEN SOURCE SOFTWARE

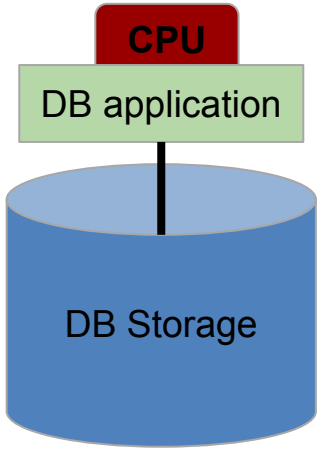
SkyhookDB: leveraging programmable storage toward DB elasticity

CROSS Incubator Project

Jeff LeFevre and Noah Watkins

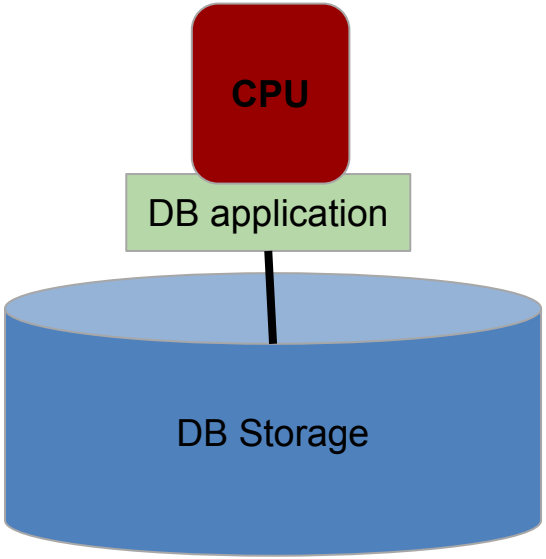
Growing your database now

- Single node architecture



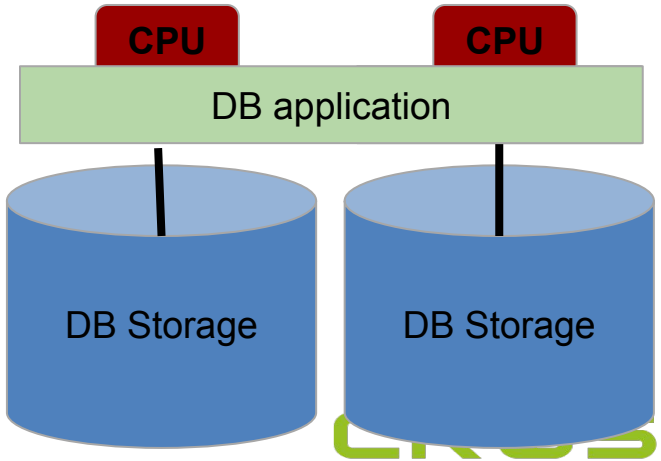
Growing your database now

- Single node architecture
  - Bigger machine!
  - No changes to DB app
- Challenge
  - Changing hardware
  - Big enough?



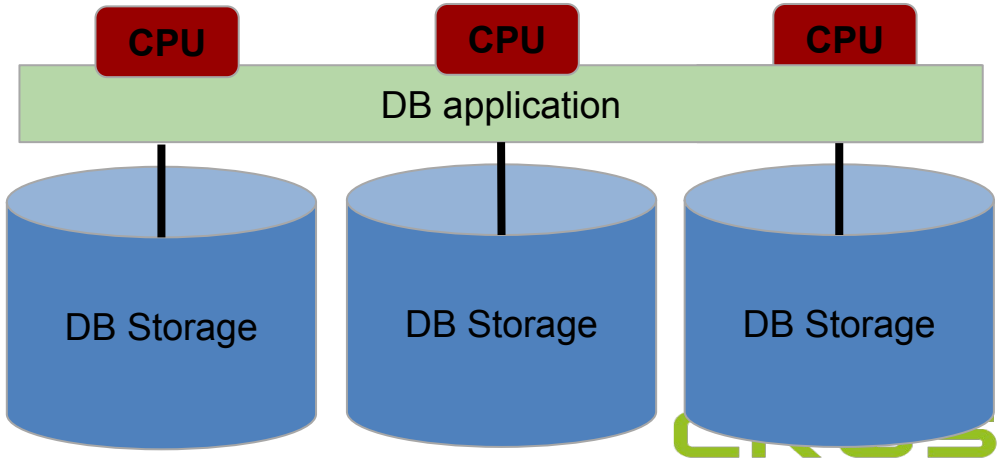
Growing your database now

- Multi-node architecture (Parallel DB / MPP)



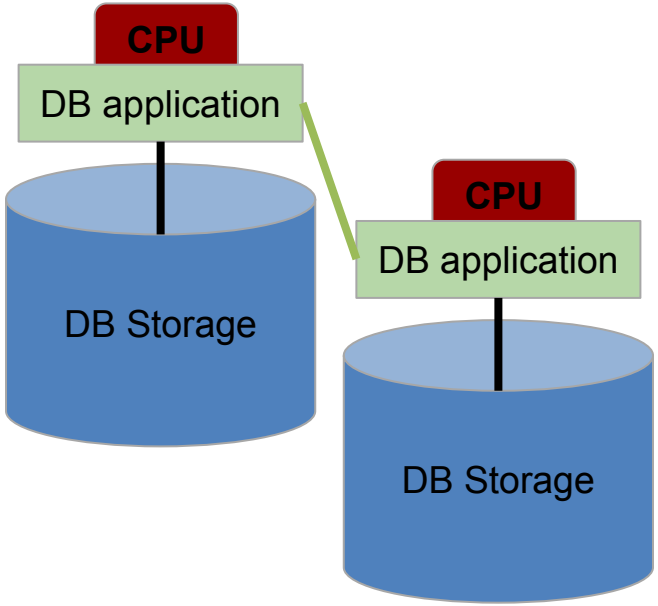
Growing your database now

- Multi-node architecture (Parallel DB / MPP)
  - More machines!
  - More Parallelism
- Challenge
  - Data redistrib
  - Complexity



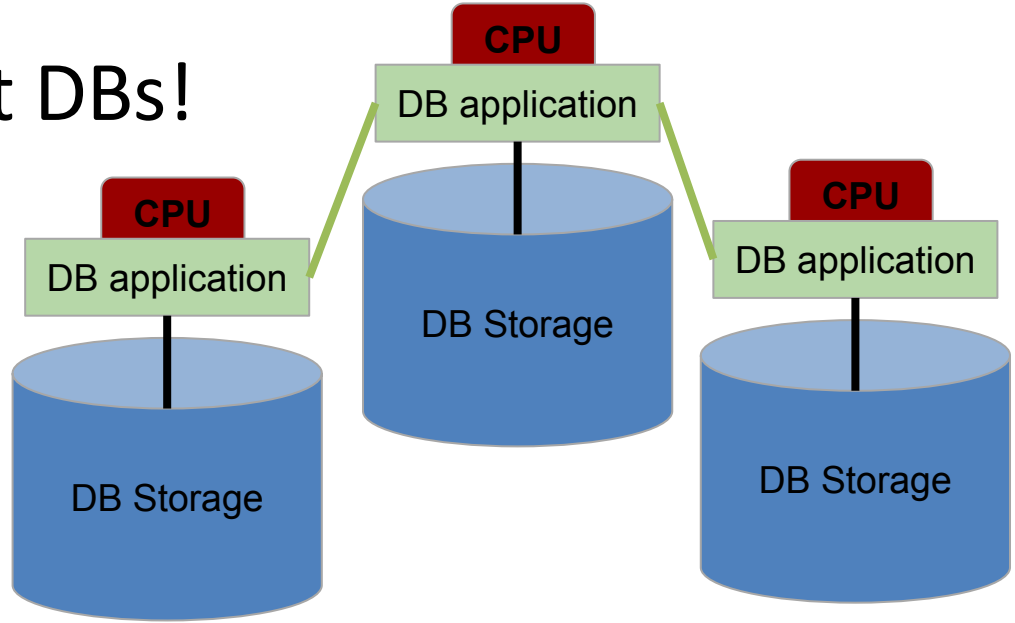
Growing your database now

- Hybrid Architecture
  - Link independent DBs!
  - More parallelism
- Challenges
  - Coordination
  - Data redistrib



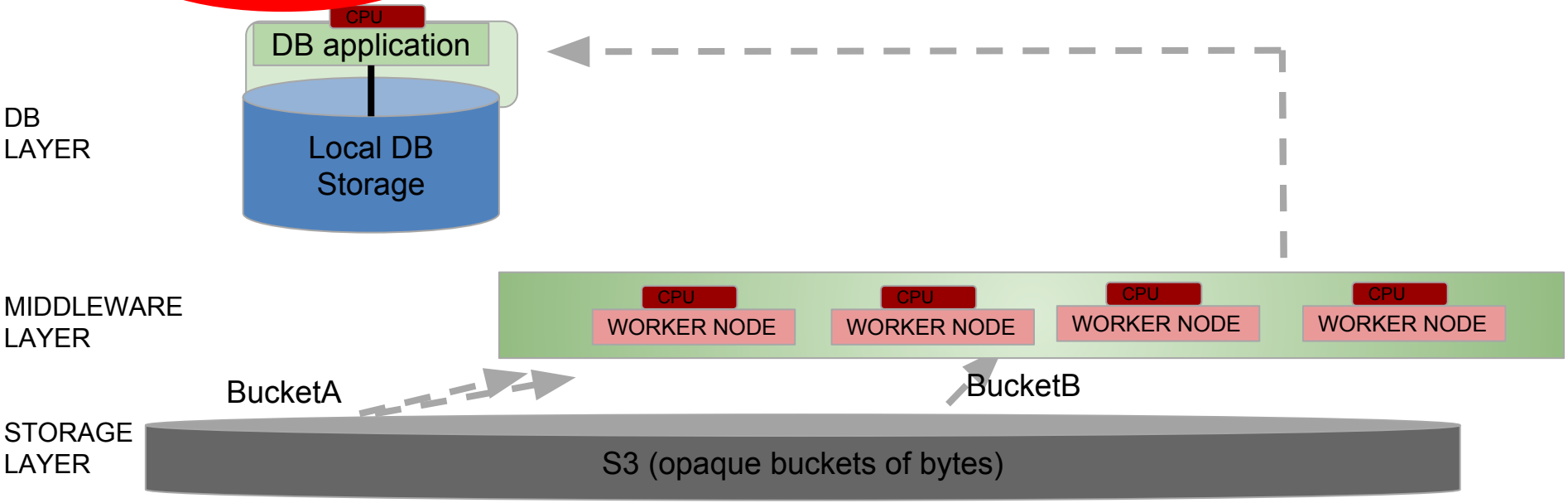
Growing your database now

- Hybrid Architecture
  - Link independent DBs!
  - More parallelism
- Challenges
  - Coordination
  - Data redistrib



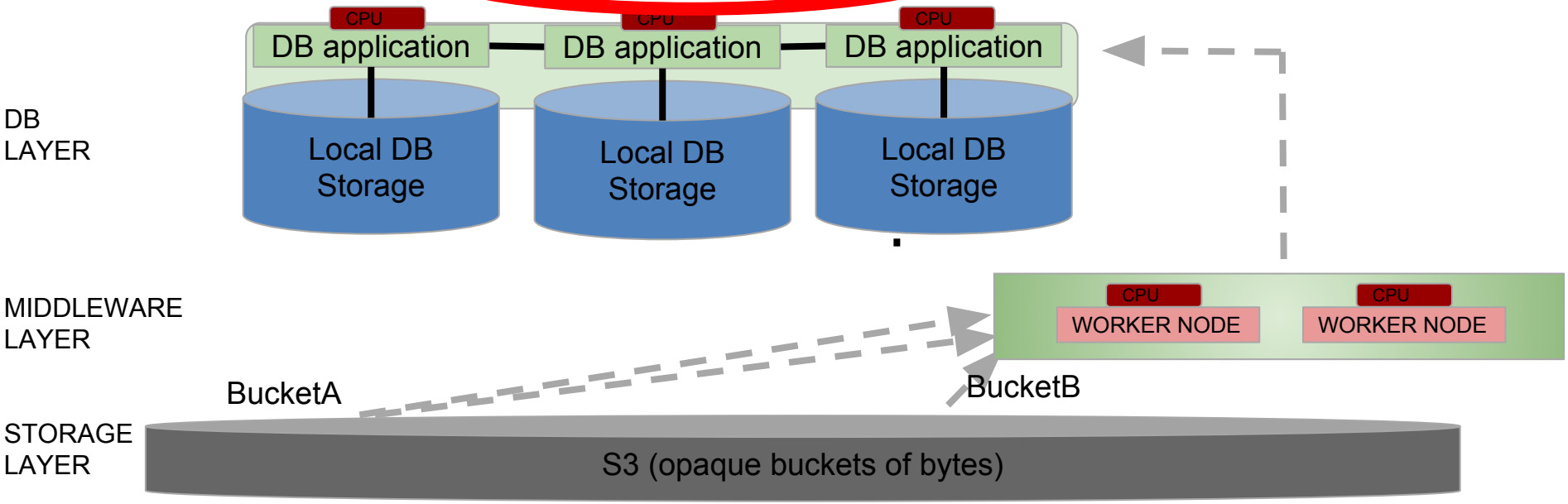
Cloud: single node architectures

- AWS Aurora, AWS Redshift Spectrum, Snowflake Computing



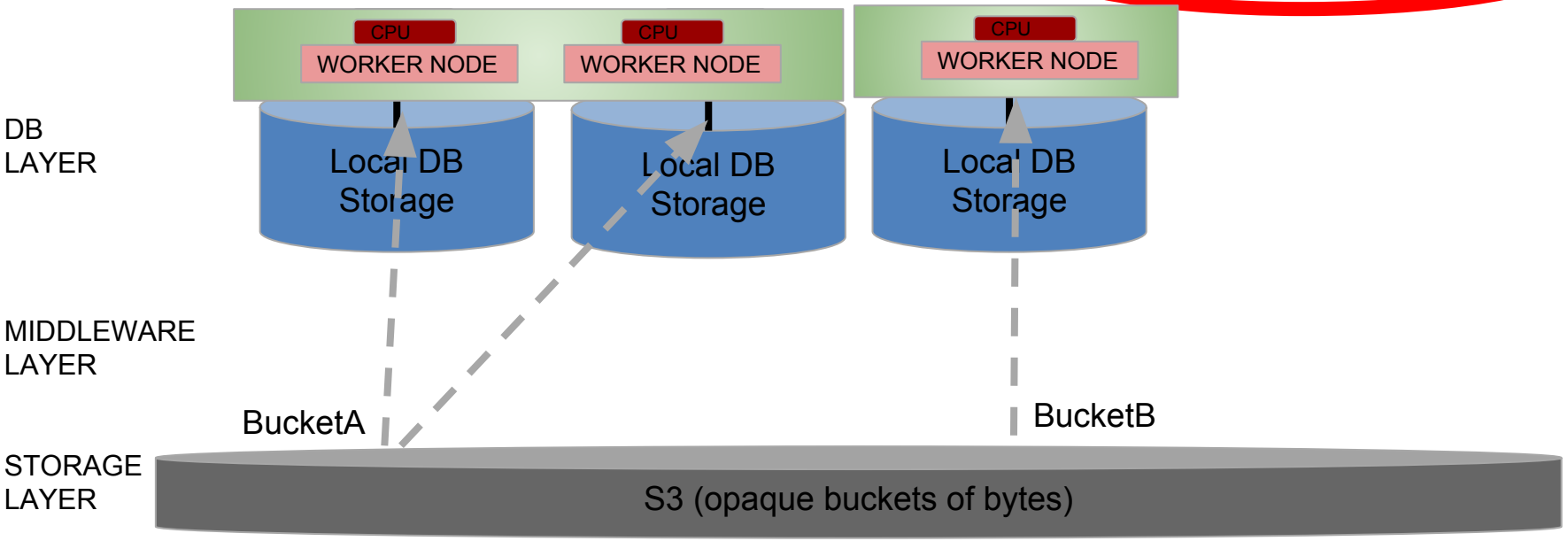
Cloud: MPP architectures

- AWS Aurora, AWS Redshift Spectrum, Snowflake Computing



Cloud: MPP architectures

- AWS Aurora, AWS Redshift Spectrum, Snowflake Computing



Cloud: DB architectures on S3

- AWS Aurora, AWS Redshift Spectrum, Snowflake Computing

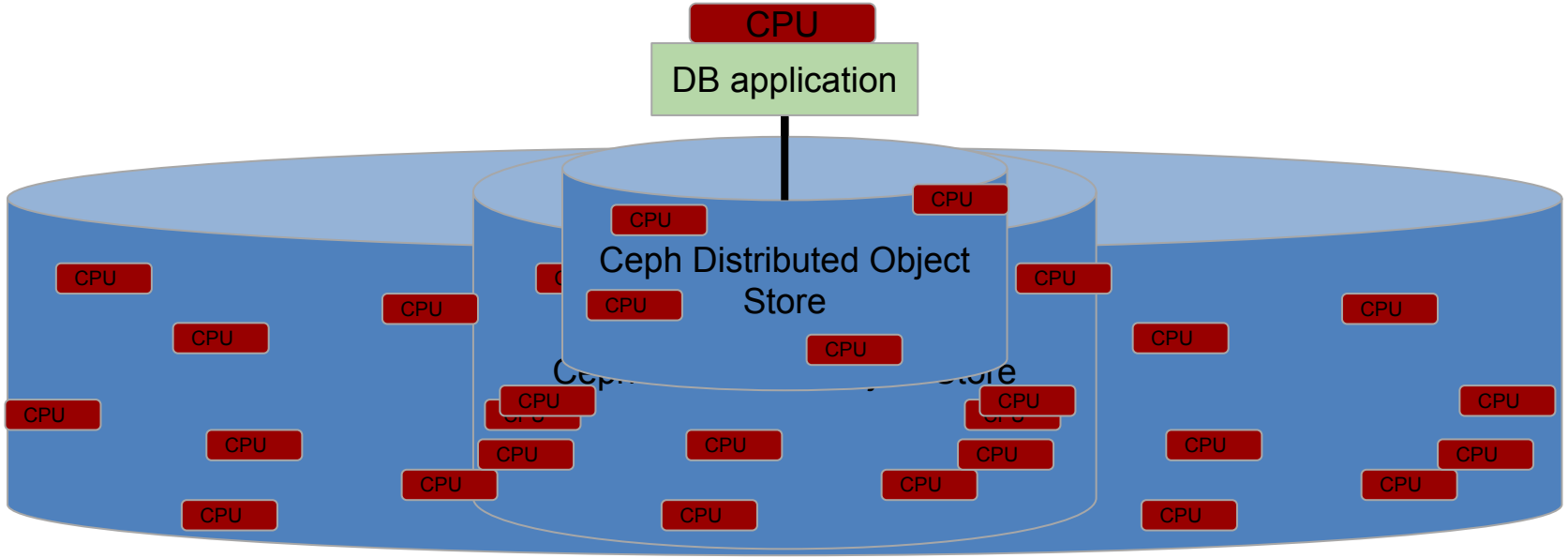
DB
LAYER

MIDDLEWARE
LAYER

STORAGE
LAYER

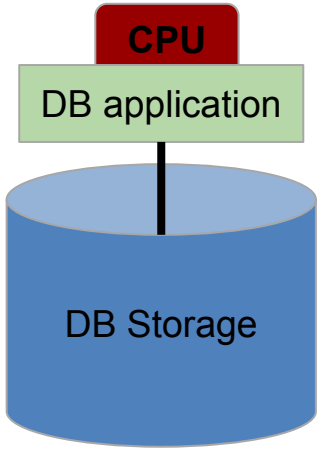
NO DB PROCESSING IS DONE "IN" S3

SkyhookDB



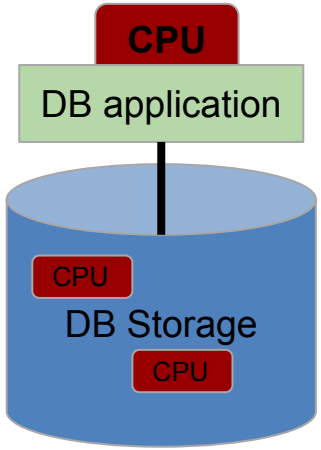
SkyhookDB

- Single node architecture
  - Simple
- Distrib Object storage
  - Parallel IO, Extensible



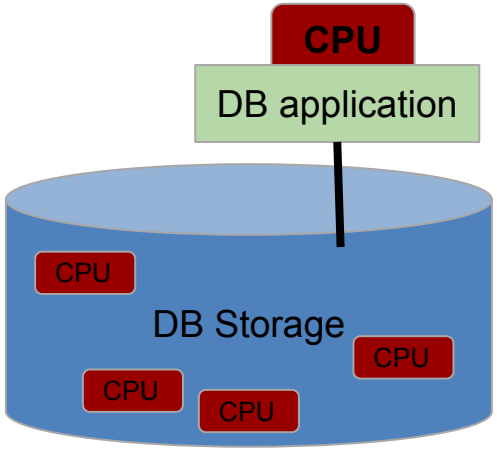
SkyhookDB

- Single node architecture
  - Simple
- Distrib Object storage
  - Parallel IO, Extensible



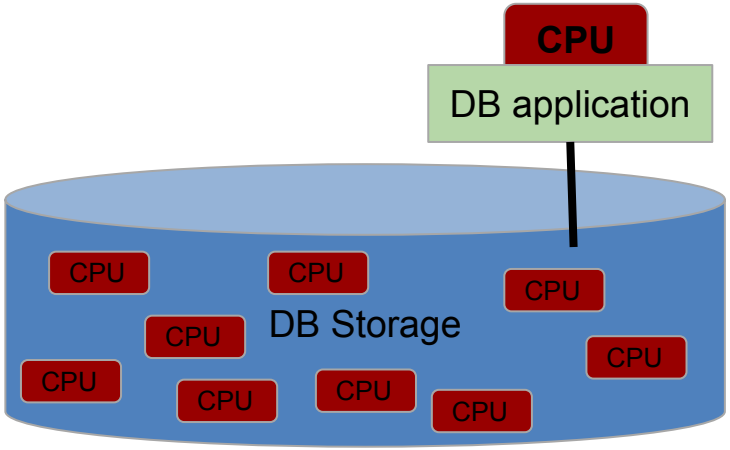
SkyhookDB

- Single node architecture
  - Simple
- Distrib Object storage
  - Parallel IO, Extensible
- Dynamically scale out
  - CPU with storage!



SkyhookDB

- Single node architecture
  - Simple
- Distrib Object storage
  - Parallel IO, Extensible
- Dynamically scale out
  - CPU with storage!



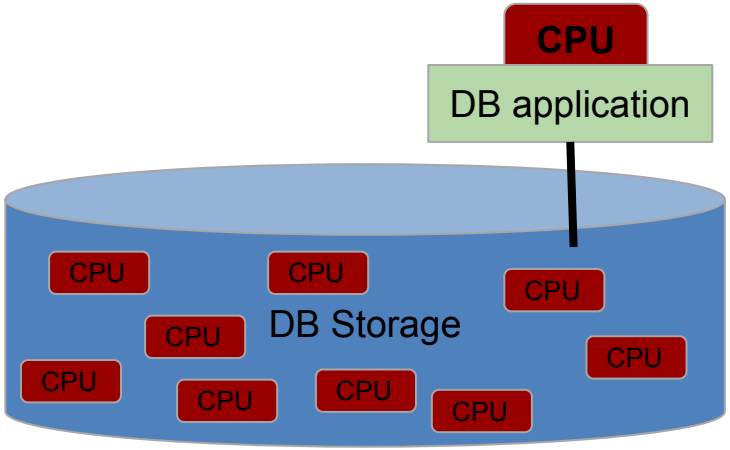
Leverage+Extend Object Storage

- Object-based ***data layout***
  - Horizontal partitioning (i.e., sets of rows)
- Object-based ***processing operations***
  - SQL, complex functions
- Object-based ***metadata management***
  - Indexes, statistics, collection-level optimizations

Utilize trusted open source software

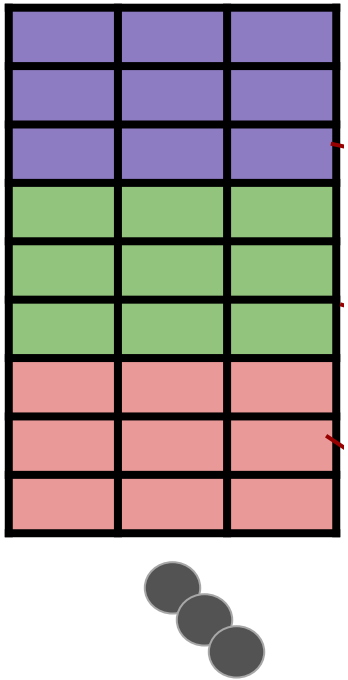
PostgreSQL database

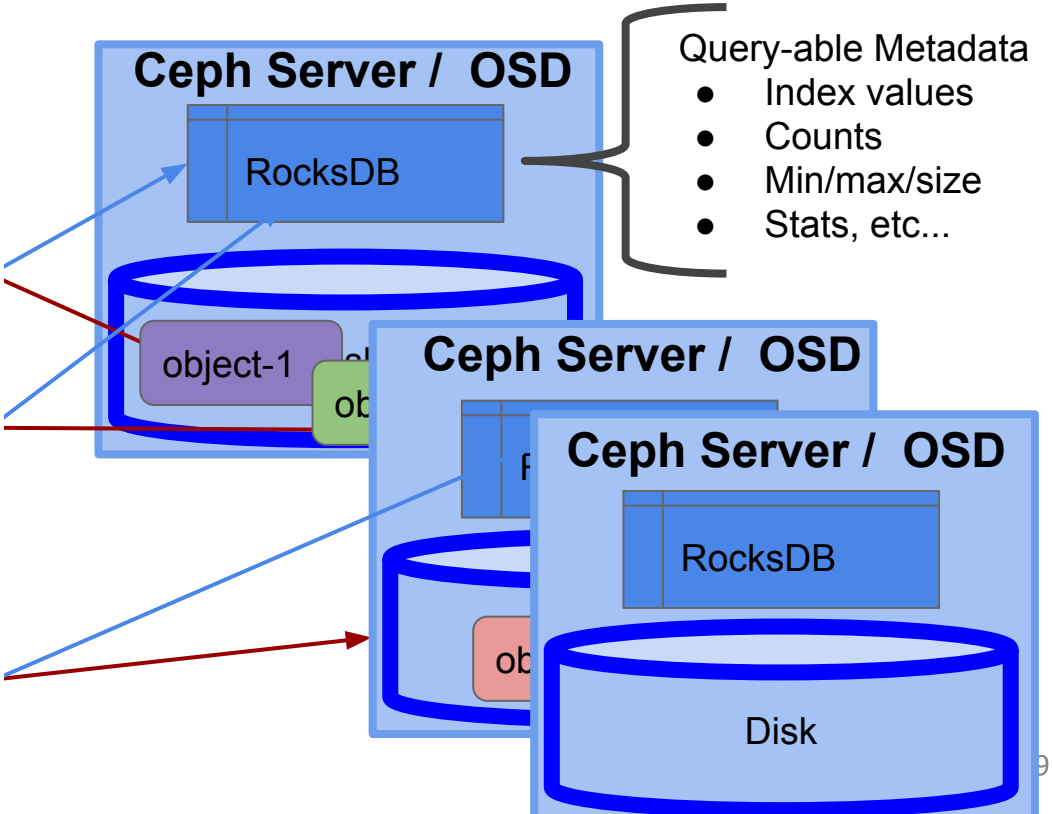
Ceph object storage



Design Details

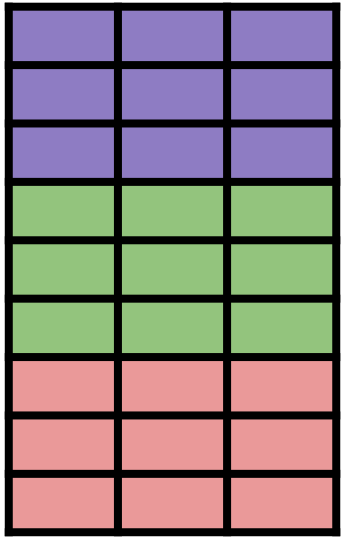
Table (rows)

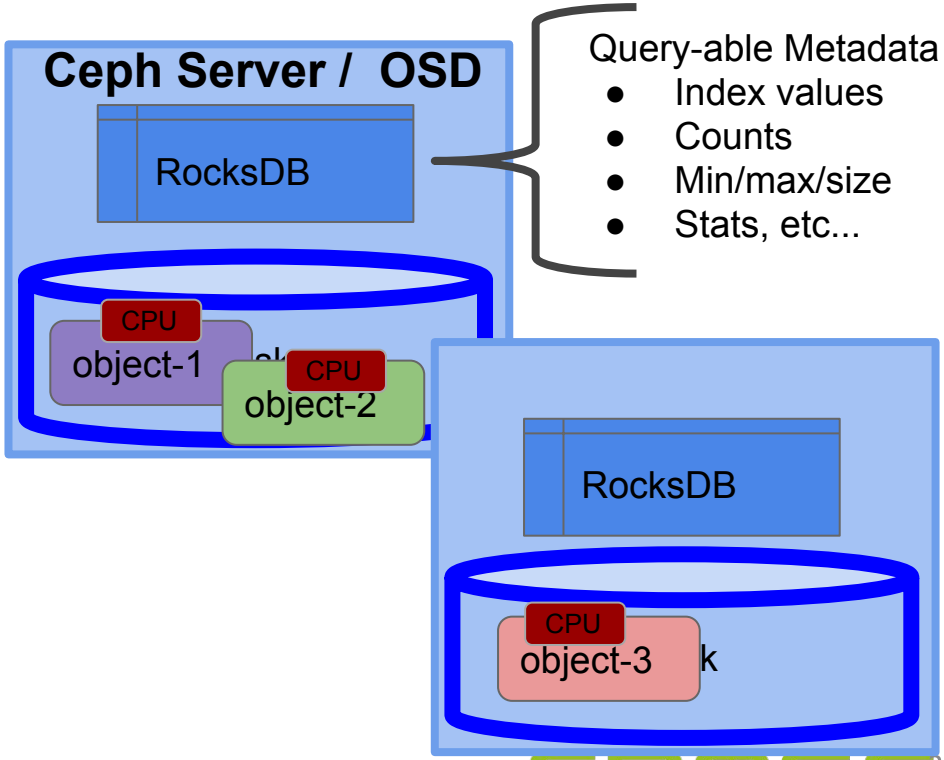




Unit of Parallelism is an Object

Table (rows)

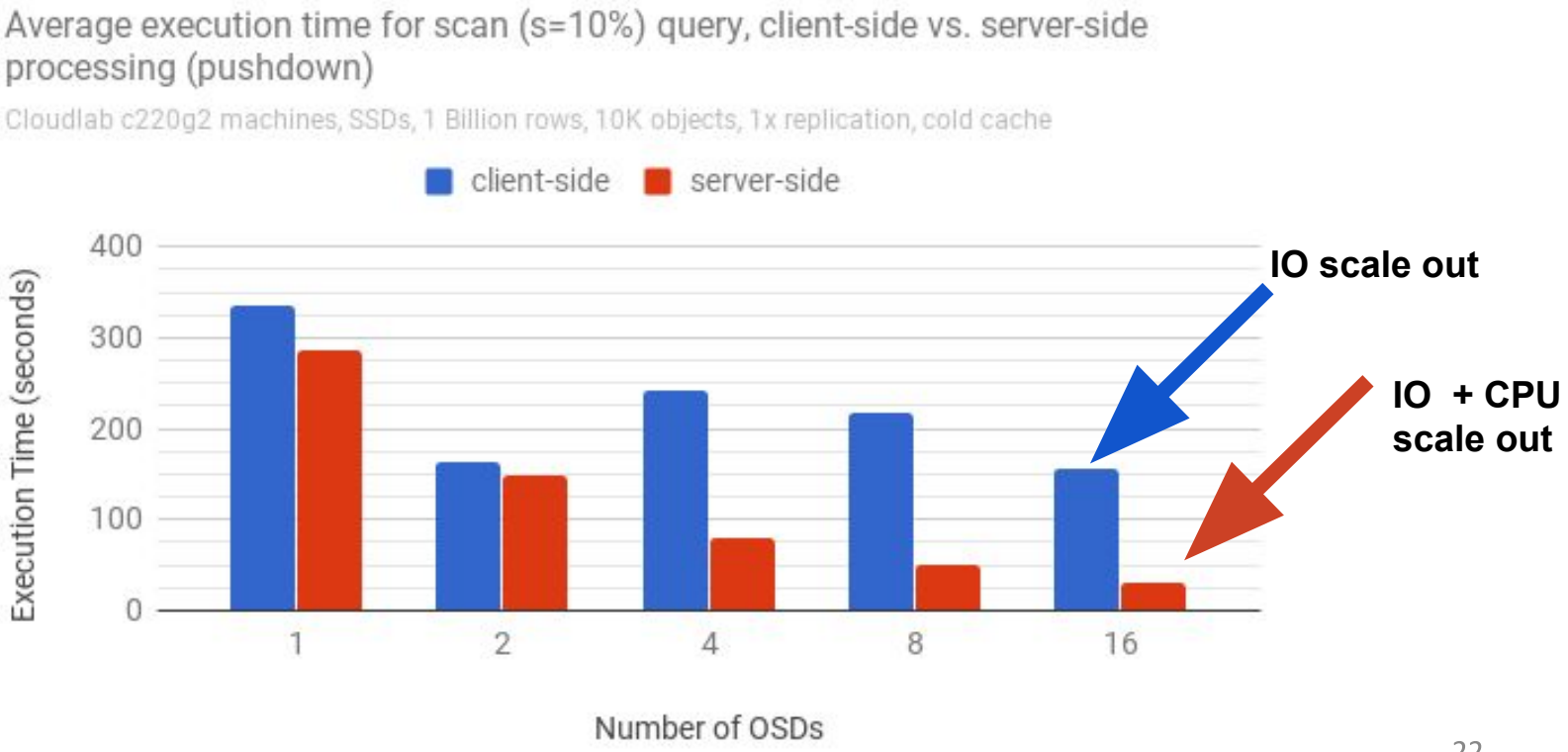




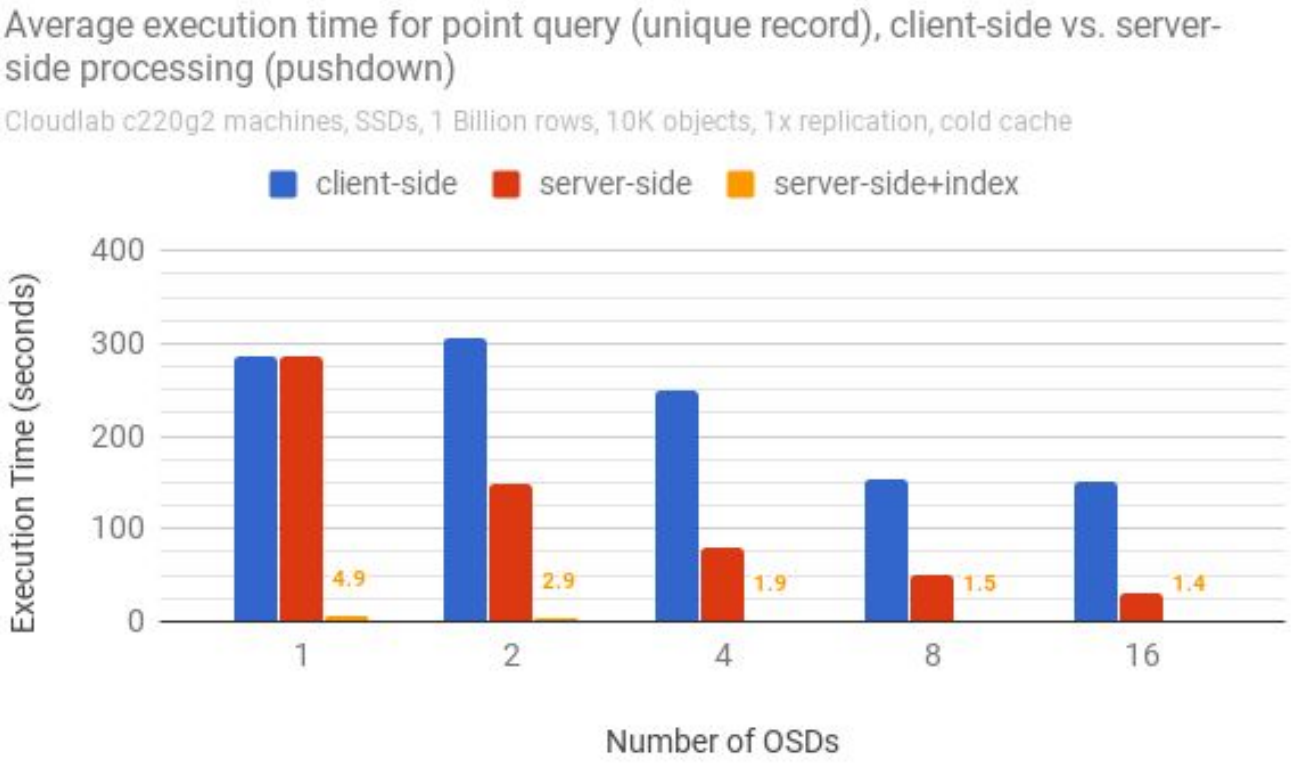
Experimental Setup

- Dataset:
  - TPC lineitem table, 1 billion rows ~140GB
  - Ceph 10K objects ~4MB each (each with index)
- Queries:
  - Range query, regex query (LIKE), point query
- Machines:
  - Ubuntu16, 20 Cores, 500GB SSDs, 10GbE (Cloudlab)
- Skyhook parallelism - 12/node IO dispatch

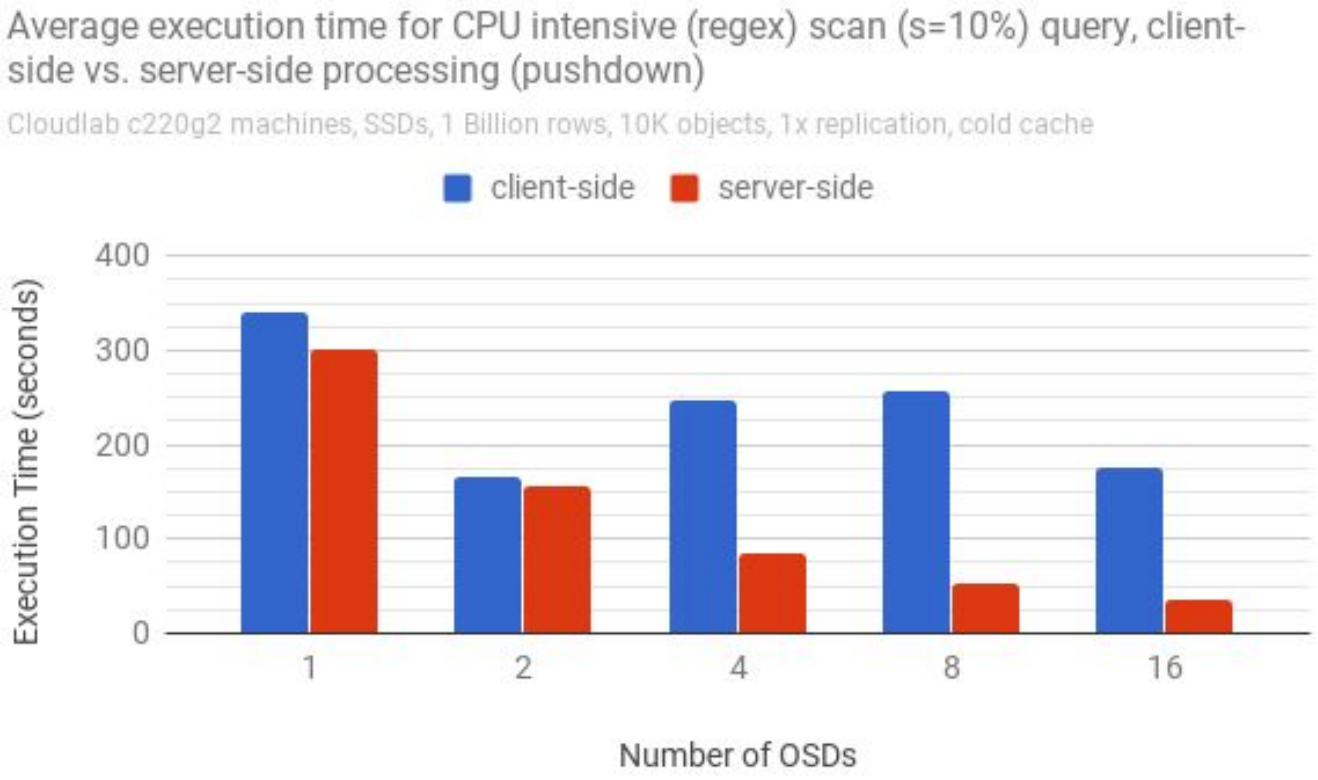
Range query (selectivity=10%)



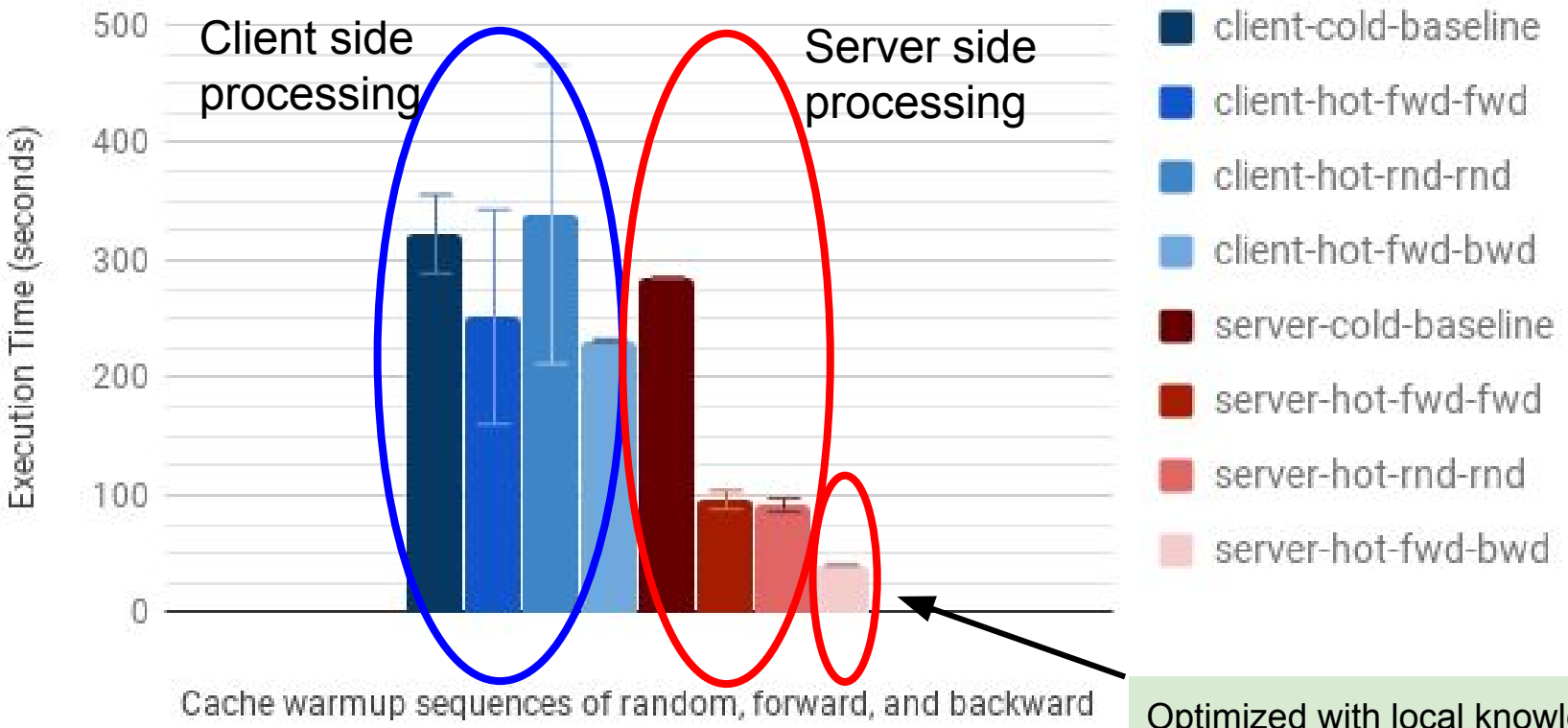
Point query(unique row)



Regex query (selectivity=10%)



Memory Pressure: Data > Memory



Optimized with local knowledge

SkyhookDB Summary

- New architecture for DB scaling
  - Better matches the cloud storage model
  - Take some processing away from DB, push into storage layer
  - More OSDs == More parallelism
- Pushing functionality *directly into the Storage Layer*
  - Processing, indexing, statistics, counts, ***metadata querying***, batching, …, others?
    - Domain specific custom functions
    - Works well for highly parallelizable tasks

References (1)

[Oracle 2016a] Oracle database appliance, Nov 2016.

[Oracle 2012] A Technical overview of the Oracle Exadata database machine and Exadata Storage Server, Jun 2012.

[Oracle 2016b] Oracle database appliance evolves as an enterprise private cloud building block, Jan 2016.

[Kee] K. Keeton, D.Patterson, J.Hellerstein, *A case for intelligent disks (iDISKS)*, SIGMOD Record, Sep 1998.

[Bor] H.Boral, D.DeWitt, *Database machines, an idea whose time has passed?*, Workshop on Database Machines 1983.

[Woo] L. Woods, Z.Istvan, G.Alonso, *Ibex: an intelligent storage engine with support for advanced SQL offloading*, VLDB 2014.

[Teradata 2016] Marketing statement: "*Multi-dimensional scalability through independent scaling of processing power and storage capacity* via our next generation massively parallel processing (MPP) architecture"

[Sevilla+Watkins 2017] M. Sevilla, N. Watkins, I. Jimenez, P. Alvaro, S. Finkelstein, J. LeFevre, C. Maltzahn, *"Malacology: A Programmable Storage System"*, in EuroSys 2017.

[LeFevre 2014] J. LeFevre, J. Sankaranarayanan, H. Hacigumus, J. Tatemura, N. Polyzotis, M.J. Carey, *"MISO: Souping Up Big Data Query Processing with a Multistore System"*, in SIGMOD 2014.

[LeFevre 2016] J. LeFevre, R. Liu, C. Inigo, M. Castellanos, L. Paz, E. Ma, M. Hsu, *"Building the Enterprise Fabric for Big Data with Vertica and Spark"*, in SIGMOD 2016.

References (2)

[Anantharayanan 2011] Ganesh Anantharayanan, Ali Ghodsi, Scott Shenker, Ion Stoica, "*Disk-Locality in Datacenter Computing Considered Irrelevant*", in USENIX HotOS, May 2011.

[Dageville 2016] Benoit Dageville, Thierry Cruanes, Marcin Zukowski, et. al, "*The Snowflake Elastic Data Warehouse*", in SIGMOD 2016.

[FlashGrid 2017] White paper on Mission-Critical Databases in the Cloud. "*Oracle RAC on Amazon EC2 Enabled by FlashGrid® Software*". rev. Jan 6, 2017.

[Gupta 2015] Anurag Gupta, Deepak Agarwal, Derek Tan, et.al, "*Amazon Redshift and the Case for Simpler Data Warehouses*", in SIGMOD 2015.

[Srinivasan 2016] Vidhya Srinivasan, "*What's New with Amazon Redshift*", AWS re-invent, Nov 2016.

[Dynamo 2017] Amazon's DynamoDB, https://aws.amazon.com/dynamodb/, accessed Jan 2017.

[Brantner 2008] Matthias Brantner, Daniela Florescu, David Graf, Donald Kossmann, Tim Kraska, "*Building a database on S3*", in SIGMOD 2008.

[Shuler 2015] Bill Shuler, "*CitusDB Architecture for Real-time Big Data*", Jun 2015.

[Citusdata 2017] Citus Data, https://www.citusdata.com, accessed Jan 2017.

[Pavlo 2016] Andrew Pavlo, Matthew Aslett, "*What's Really New with NewSQL?*", in SIGMOD Record Jun 2016.

References (3)

[Xi 2015] Sam (Likun) Xi, Oreoluwa Babarinsa, Manos Athanassoulis, Stratos Idreos, "*Beyond the Wall: Near-Data Processing for Databases*", in DAMON 2015.

[Kim 2015] Sungchan Kim, Hyunok Ohb, Chanik Parkc, Sangyeun Choc, Sang-Won Leed, Bongki Moone, "*In-storage processing of database scans and joins*", Information Sciences, 2015.

[Do 2013] Jaeyoung Do, Yang-Suk Kee, Jignesh M. Patel, Chanik Park, Kwanghyun Park, David J. DeWitt, "*Query Processing on Smart SSDs: Opportunities and Challenges*", in SIGMOD 2013.

[Gkantsidis] Christos Gkantsidis, Dimitrios Vytiniotis, Orion Hodson, Dushyanth Narayanan, Florin Dinu, Antony Rowstron, "*Rhea: automatic filtering for unstructured cloud storage*", in NSDI 2013.

[Balasubramonian 2014] Rajeev Balasubramonian, Jichuan Chang, Troy Manning, Jaime H. Moreno, Richard Murphy, Ravi Nair, Steven Swanson, "*NEAR-DATA PROCESSING: INSIGHTS FROM A MICRO-46 WORKSHOP*", IEEE Micro, Aug 2014.

[Kudu 2017] Apache Kudu, Kudu: New Apache Hadoop Storage for Fast Analytics on Fast Data

[Kudu 2015] Lipcon et. al *"Kudu: Storage for Fast Analytics on Fast Data"*, white paper.

[Moatti 2017] Too Big to Eat: Boosting Analytics Data Ingestion from Object Stores with Scoop, ICDE 2017

Backup slides

Queries

Qa: **Count** query with 10% selectivity:

SELECT count(*) FROM lineitem1B WHERE l_extendedprice > 71000.0

Qb: Range query with 10% selectivity:

SELECT * FROM lineitem1B WHERE l_extendedprice > 71000.0

Qd: Point query (unique row) issued with and without index:

SELECT l_extendedprice FROM lineitem1B WHERE l_orderkey=5 AND l_linenumber=3

Qf: Regex query with 10% selectivity:

SELECT * FROM lineitem1B WHERE l_comment iLIKE '%uriously%'

Initial comparison with single-node Postgres (selectivity=10%)

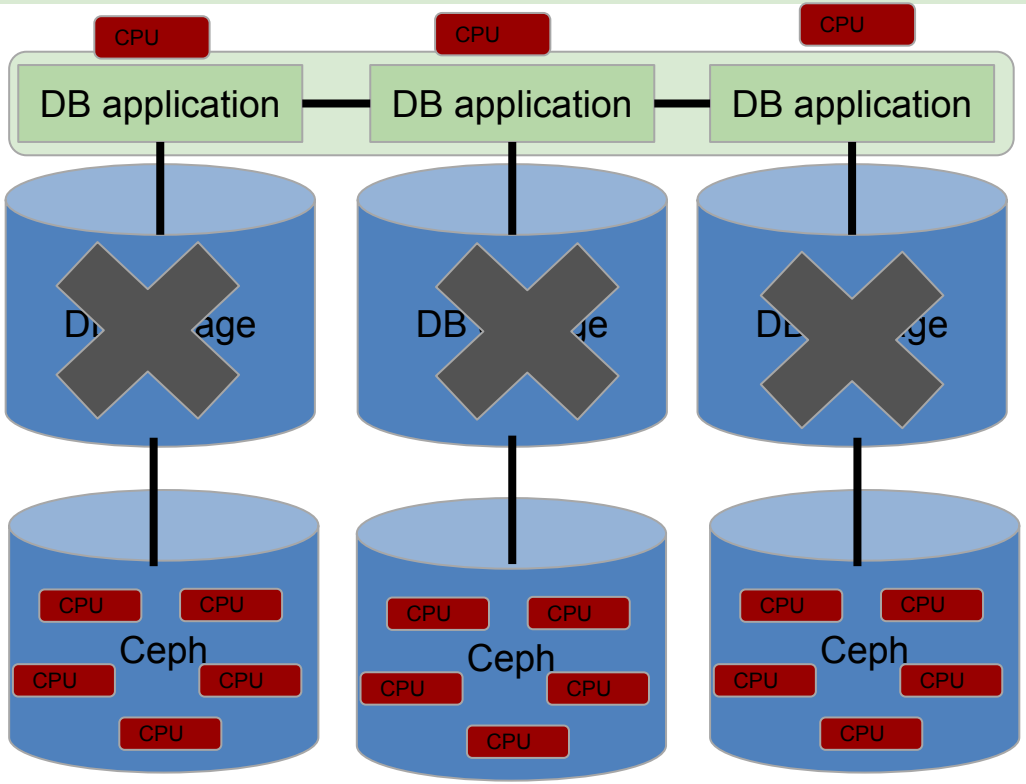
Filter: l_extendedprice >71000

Number of Storage Servers (OSDs)

Regex: l_comment like %uriously%

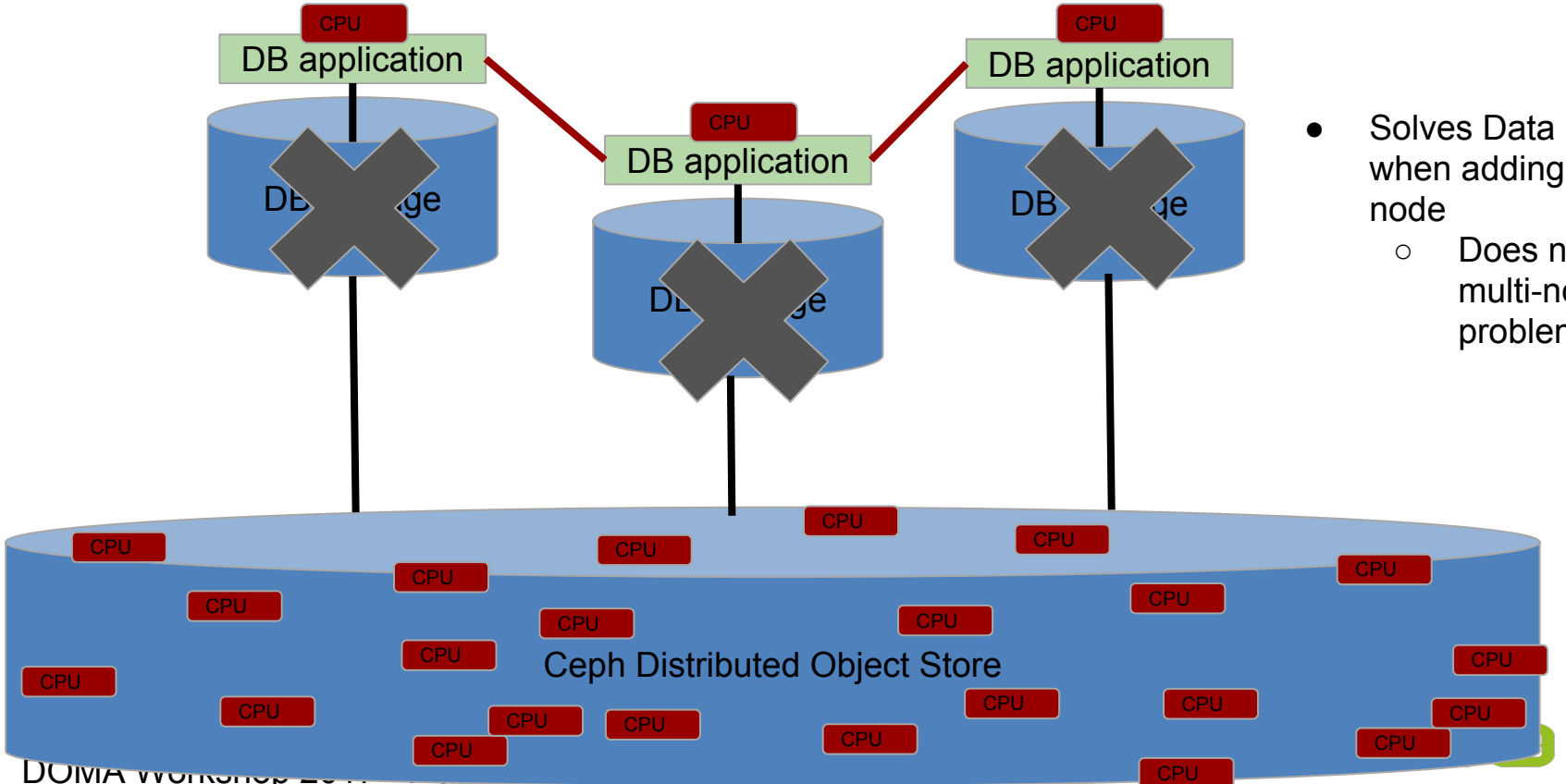
Number of Storage Servers (OSDs)

Solving the single node bottleneck - use with MPP arch



- Solves: scale up without adding new MPP nodes
  - Does not solve data redistrib/reassign prob when adding new MPP node

Solving the single node bottleneck - use with Hybrid MPP arch



- Solves Data redistrib when adding new single node
  - Does not solve multi-node xct problem