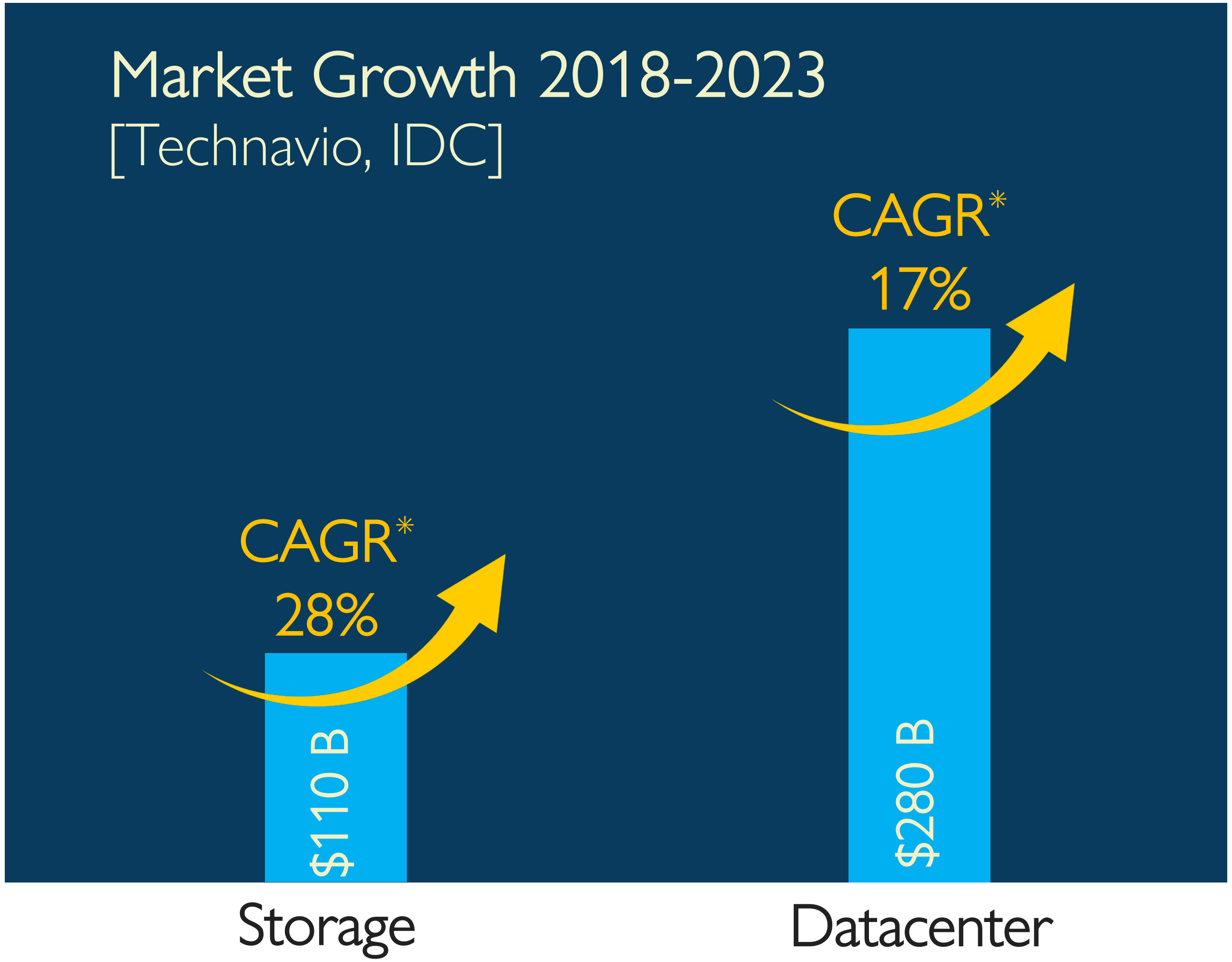
Server Benchmarking w/ CloudSuite 4.0

Team: Ali Ansari, Shanqing Lin, Rafael Pizarro Solar,
Ayan Chakraborty, Bugra Eryilmaz, Babak Falsafi, Michael Ferdman

ASPLOS'23, Vancouver

DATACENTER GROWTH



- Data → fuel for digital economy
- Exponential demand for digital services (e.g., search, media streaming)
- Many apps (e.g., AI) with higher exponential demand

*CAGR: Cumulated Annual Growth Rate

DATACENTERS ARE BACKBONE OF CLOUD

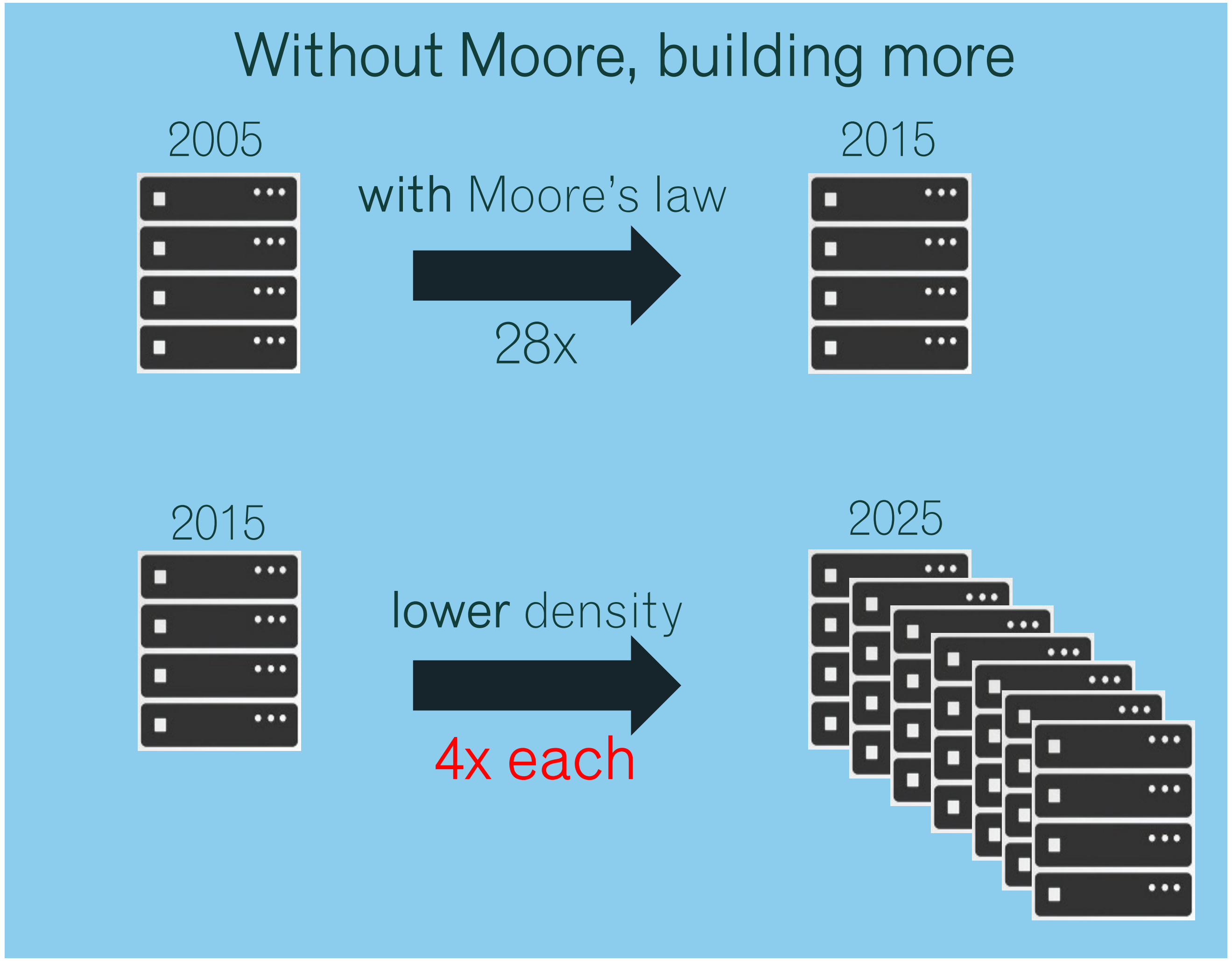
PARALLEL SYSTEMS ARCHITECTURE LAB

- 100s of 1000s of commodity or home-brewed servers
- Centralized to exploit economies of scale
- Network fabric w/ μ-second connectivity
- Often limited by
    - Electricity
    - Network
    - Cooling

350MW, Boydton

DATACENTERS NOT GETTING DENSER

PARALLEL SYSTEMS ARCHITECTURE LAB



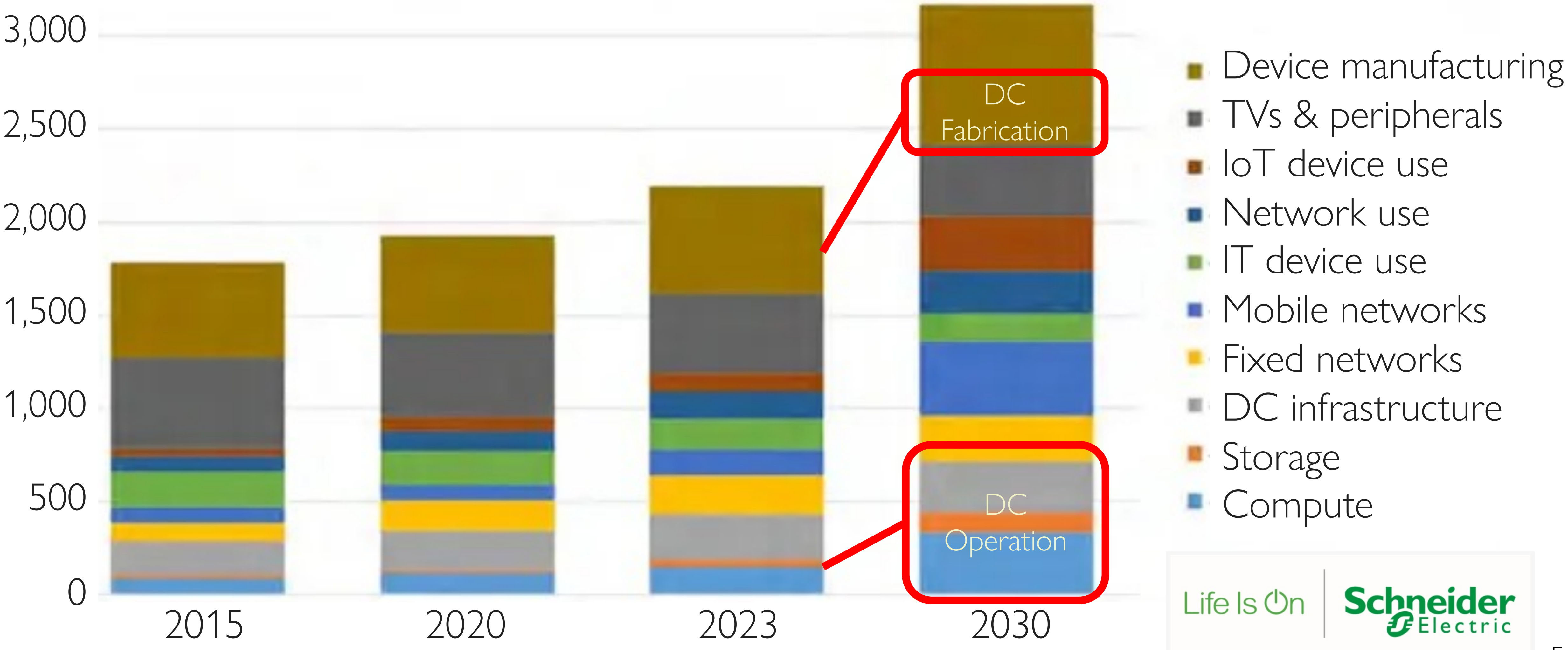
End of Moore's Law (of Silicon)

- Five decades of doubling density
- Recent slowdown in density
- Chip density limited by physics

Growth means building more

- 41%/year → 28x in ten years
- At 15%/year → 7x more DCs

IT ELECTRICITY IN TWH



Life Is On Schneider Electric

SCALE-OUT DATACENTERS

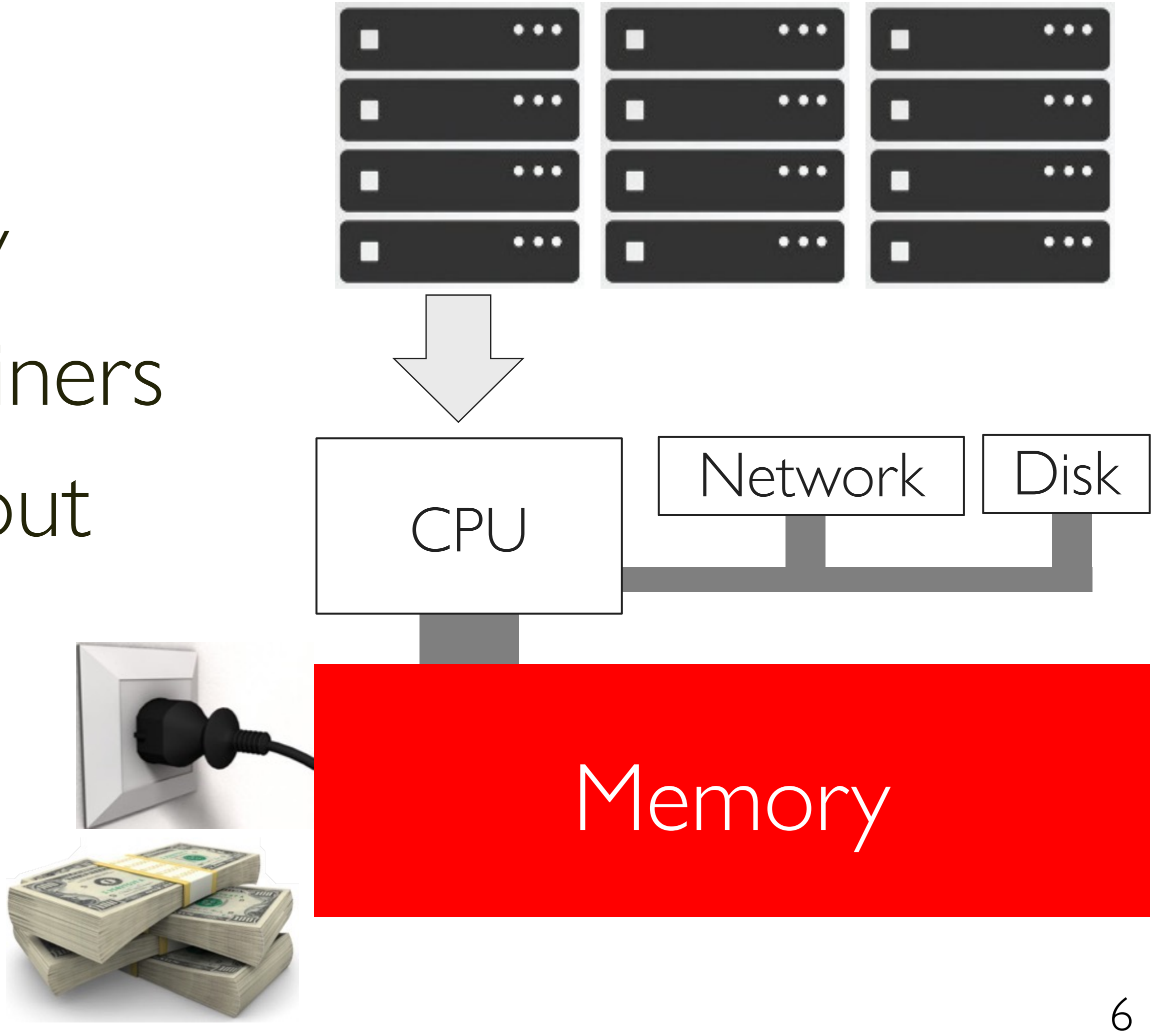
Cost is the primary metric

Online services hosted in memory

Consolidated w/ analytics & containers

Design server for low cost, scale out

☞ Memory most precious silicon



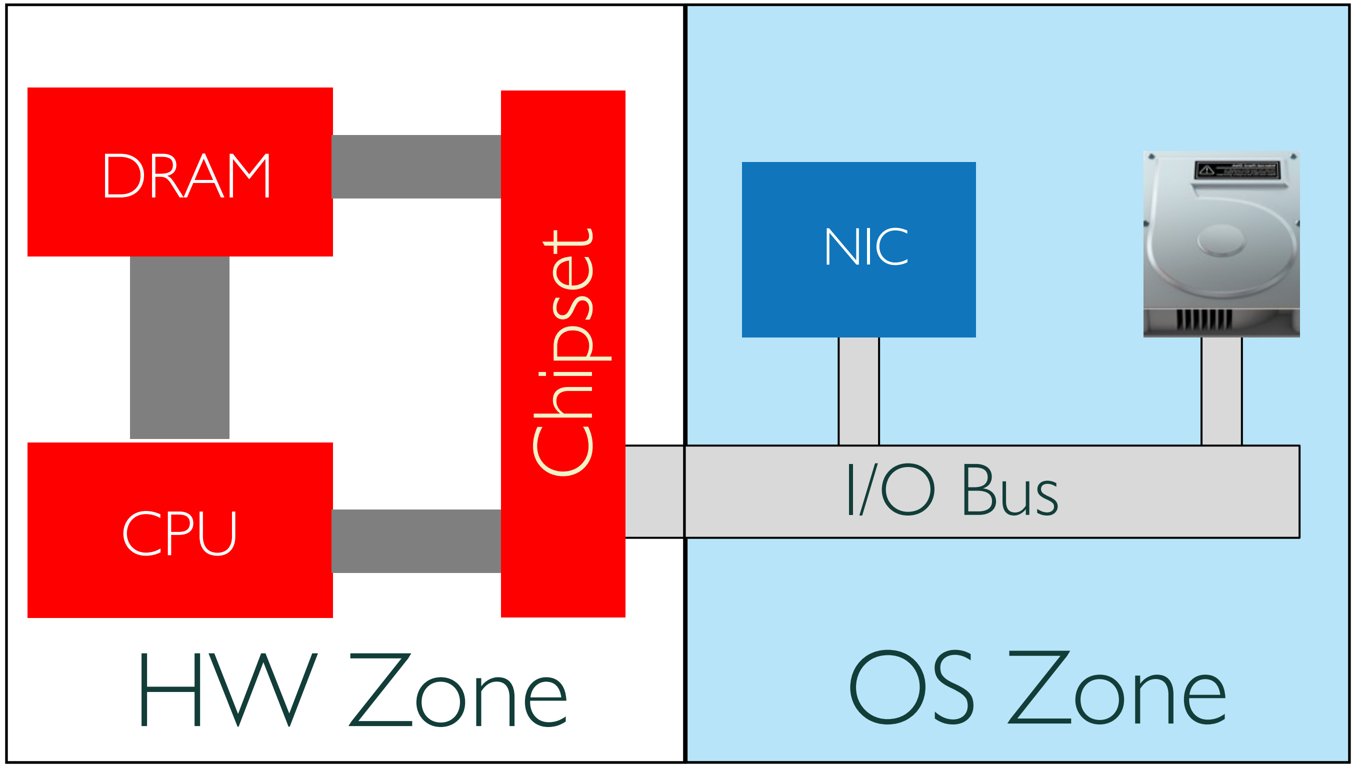
TODAY'S SERVERS

PARSA
PARALLEL SYSTEMS
ARCHITECTURE LAB

- Today's platforms are PC's of the 80's
  - CPU "owns" and manages memory
  - OS moves data back/forth from peripherals
  - Legacy interfaces connecting the CPU/mem to outside
  - Legacy POSIX abstractions

- Fragmented logic/memory:
  - Manycore network cards w/ own memory
  - Flash controllers with embedded cores and memory
  - Discrete accelerators with own memory

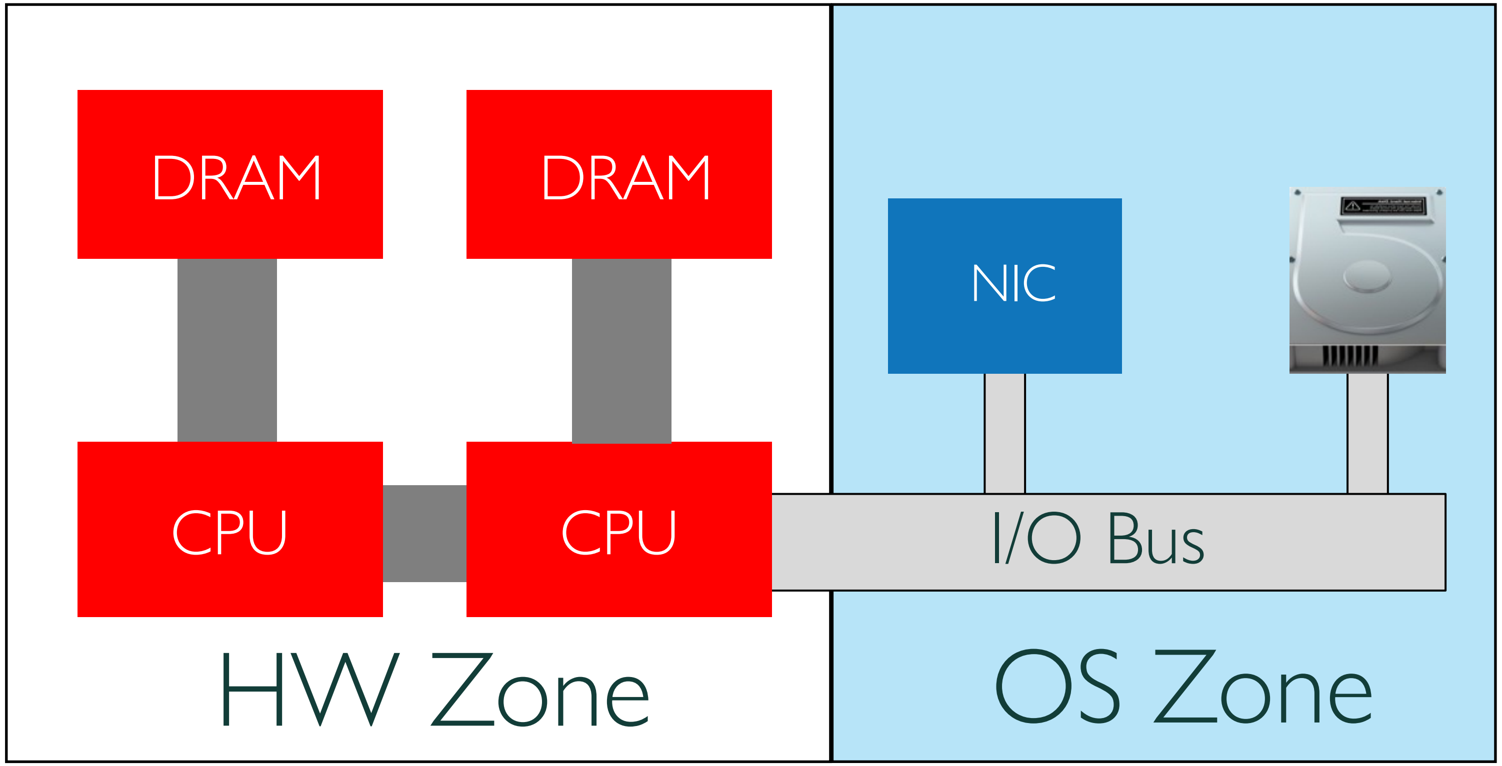
80'S DESKTOP



- 33 MHz 386 CPU, 250ns DRAM
- OS: Windows, Unix BSD (or various flavors)
- Focus: multi-programmed in-memory compute

TODAY'S SERVER vs 80'S DESKTOP

PAISA PARALLEL SYSTEMS ARCHITECTURE LAB



- Dual 2GHz CPU's, 50ns DRAM
- OS: Linux (and various distributions)

DESKTOP WORKLOADS

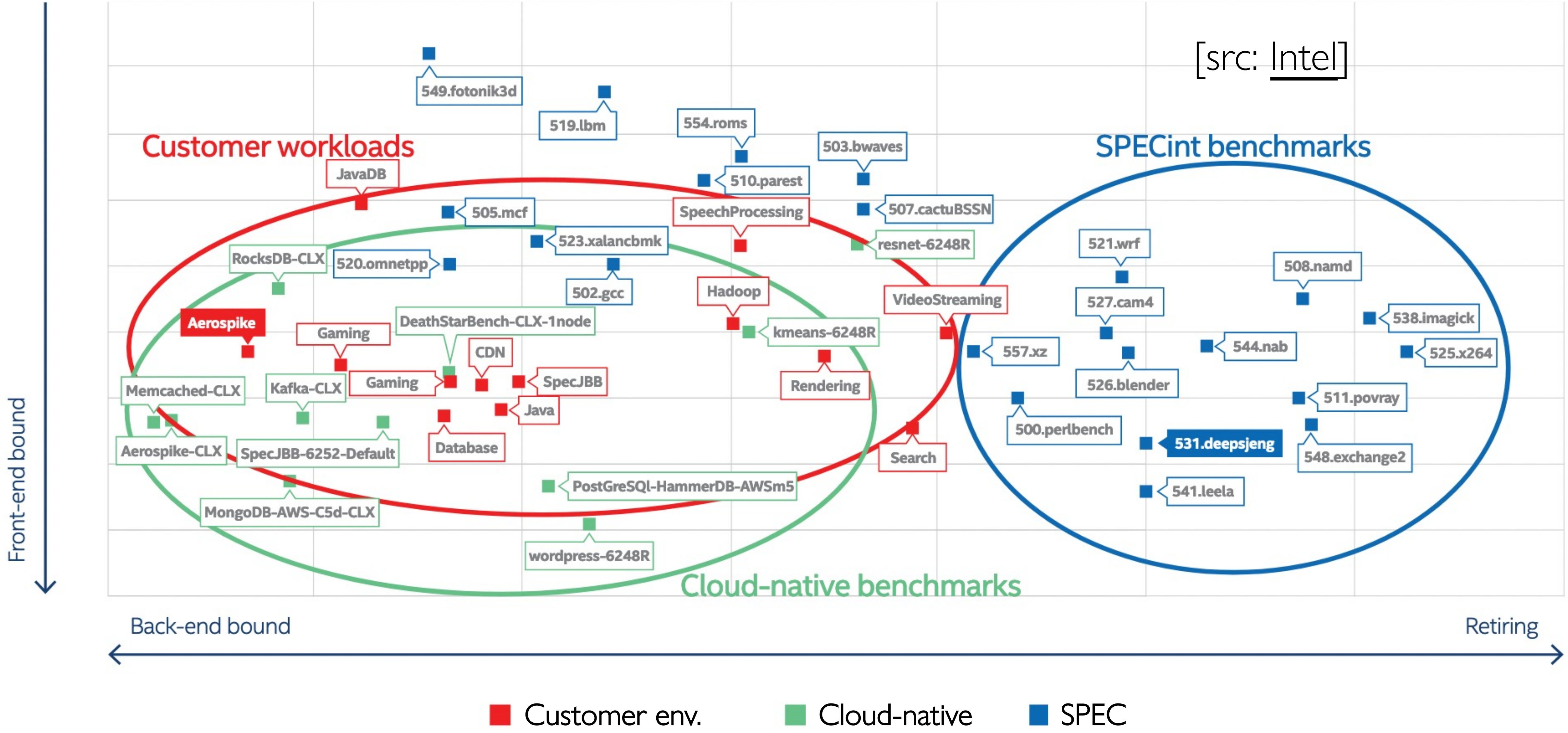
- SPECint
  - CPU integer performance
- SPECfp
  - CPU floating-point performance
- PARSEC
  - Multicore/manycore (parallel) CPU performance
- Renaissance
  - Java performance

SERVER WORKLOADS

- Independent requests/tasks
- Diverse workloads
- Deep software stacks
- Huge datasets
- Large instruction working sets
- Spend time in the OS
- Strict response-time constraints
- Various programming env.
  - Python, JAVA, Scala, Rust, C/C++, etc.

SERVER != DESKTOP WORKLOADS

PARALLEL SYSTEMS ARCHITECTURE LAB



DATACENTER SERVICES

PARALLEL SYSTEMS ARCHITECTURE LAB

"First-party" workloads (e.g., search, retail, media)

1. Online services
2. Analytics
   - A few tier monoliths (CloudSuite)
   - µServices (DeathStarBench)

"Third-party" workloads (cloud)

3. Virtualized
   - Container instances (run any suite)
   - Serverless (vHive)

CloudSuite

- Represents first-party cloud services
  - Two classes of workloads: analytics and online services
- Based on state-of-the-art open-source software stacks
- Containerized for use
- Various performance metrics
  - Throughput: requests per seconds (RPS)
  - Completion time for analytics workloads
  - Tail latency for online services
  - µArch characteristics

CloudSuite 1.0

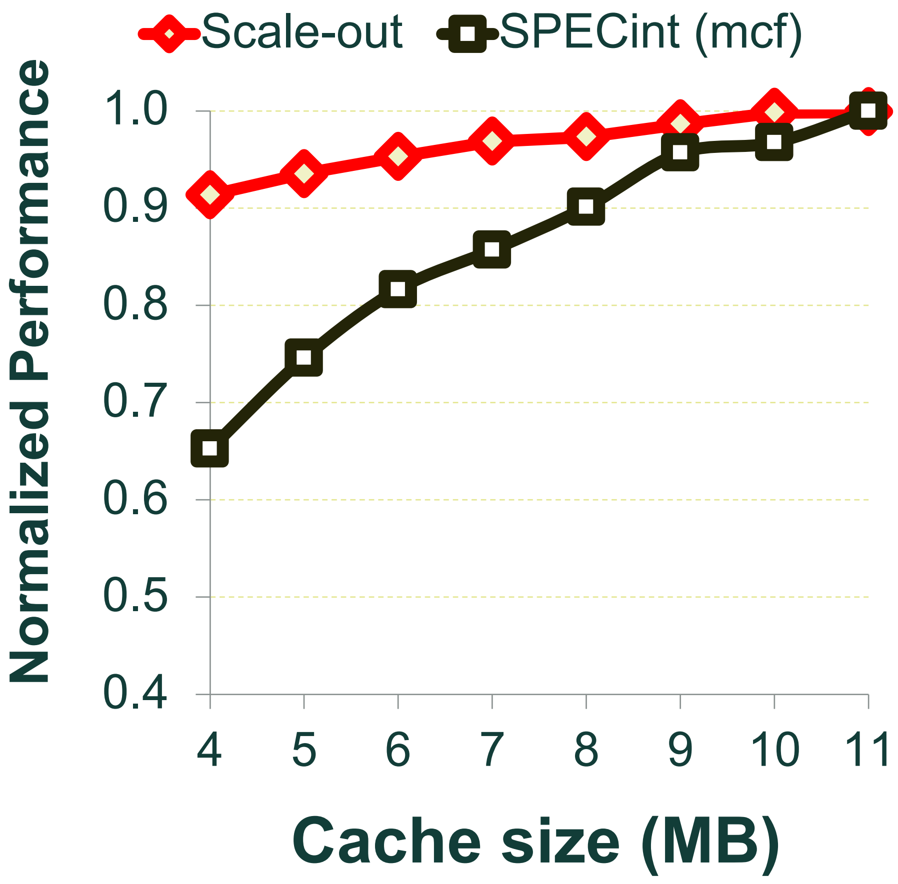
- Introduced in Clearing the Clouds [Ferdman, ASPLOS'12]
  - Best paper award
  - IEEE Micro Top Picks

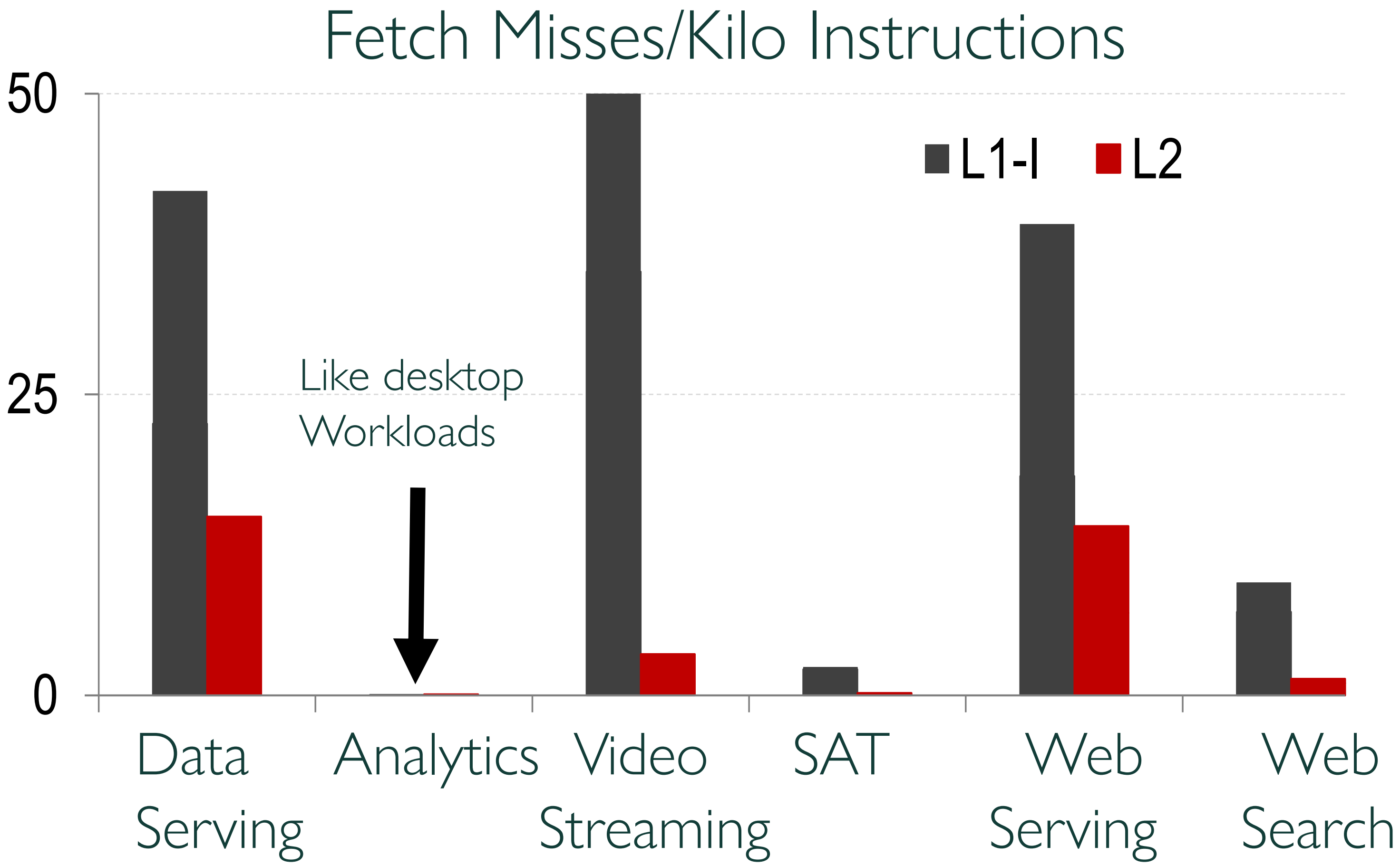
- Highlighted the characteristics of Cloud workloads
  - Instruction supply bottleneck
  - Low instruction- and memory-level parallelism
  - Data working sets beyond the on-chip cache capacities
  - Memory bandwidth overprovisioned

Mismatch between the Cloud workloads and the server CPUs

SERVICES STUCK IN MEMORY [ASPLOS'12]

PARALLEL SYSTEMS ARCHITECTURE LAB



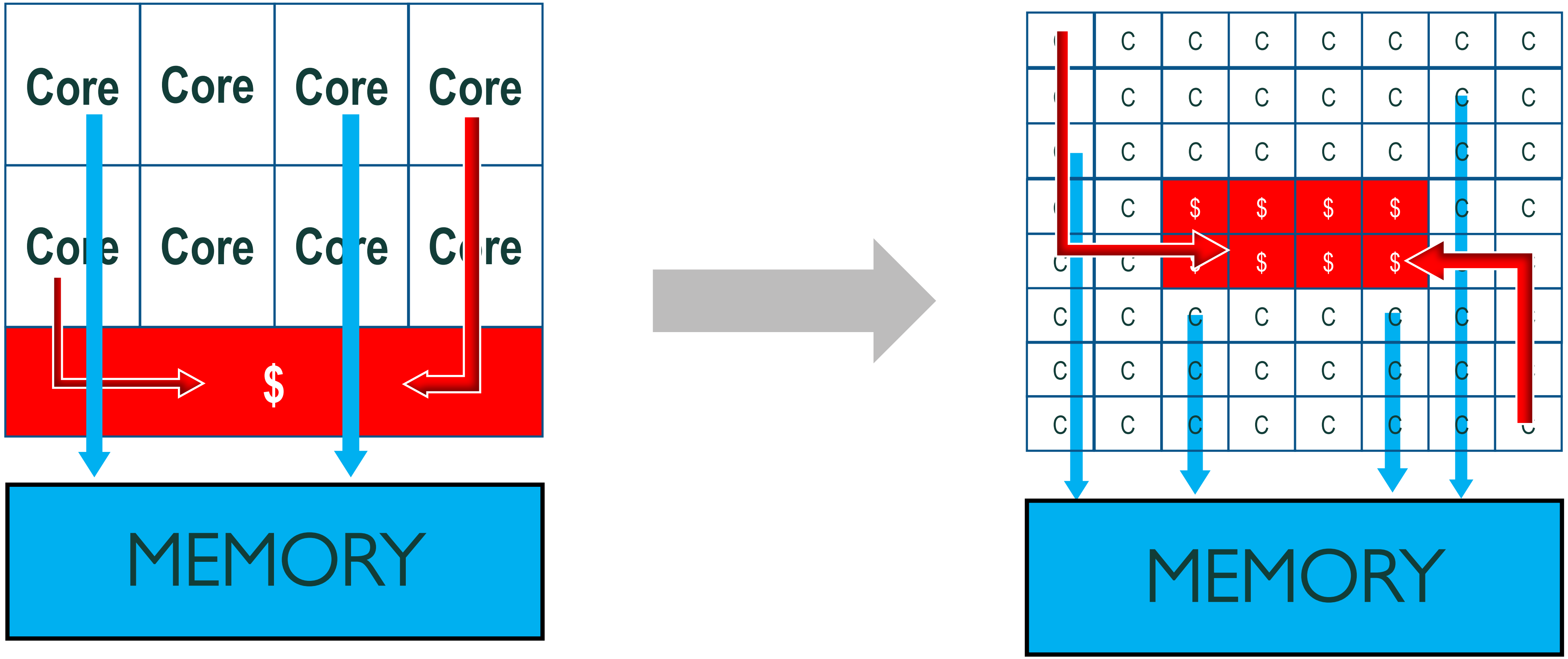


Cache overprovisioned

Instruction supply bottlenecked

SCALE-OUT PROCESSORS [ISCA'12]

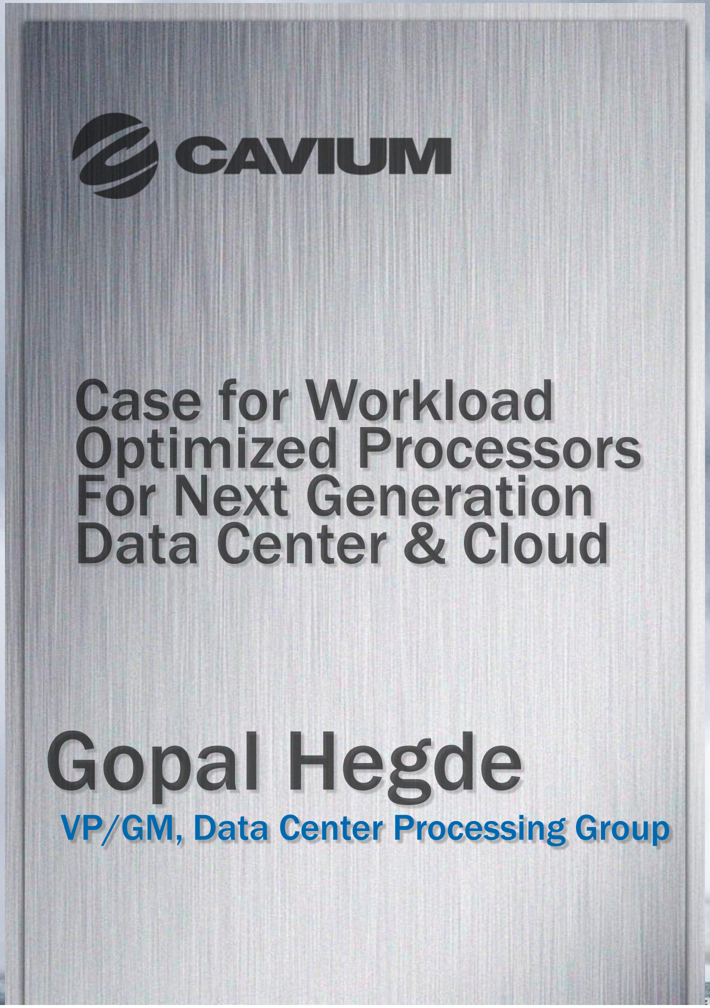
PARALLEL SYSTEMS ARCHITECTURE LAB



- General-purpose CPU
- ✗ Logic 60% of silicon
- ✗ 6x bigger cores

- 3-way OoO ARM
- ✓ 85% logic, 7x more cores
- ✓ Faster instruction supply

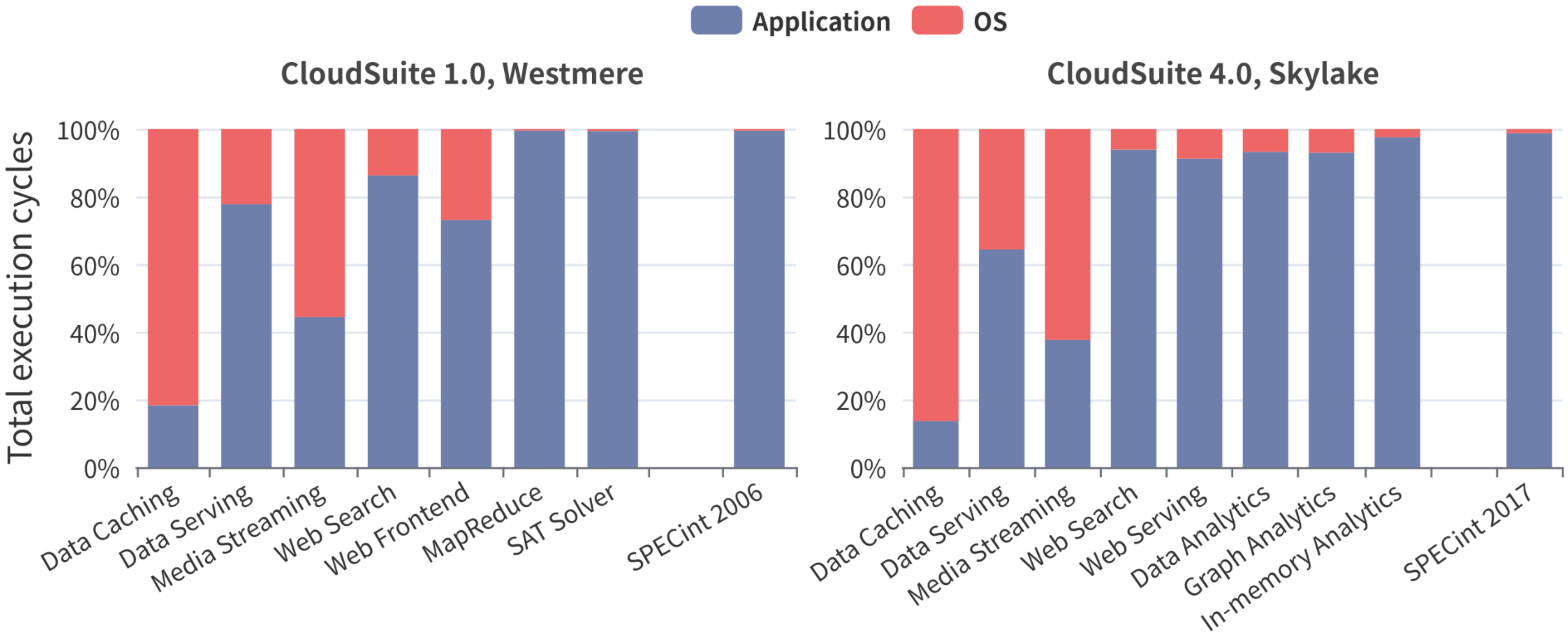
FIRST GEN. CLOUD-NATIVE CPU



Thunder X

- Based on SOP blueprint
- Designed to serve data
- 7x more core than cache
- Optimizes instruction supply
- Ran stock software
- 10x throughput over Xeon

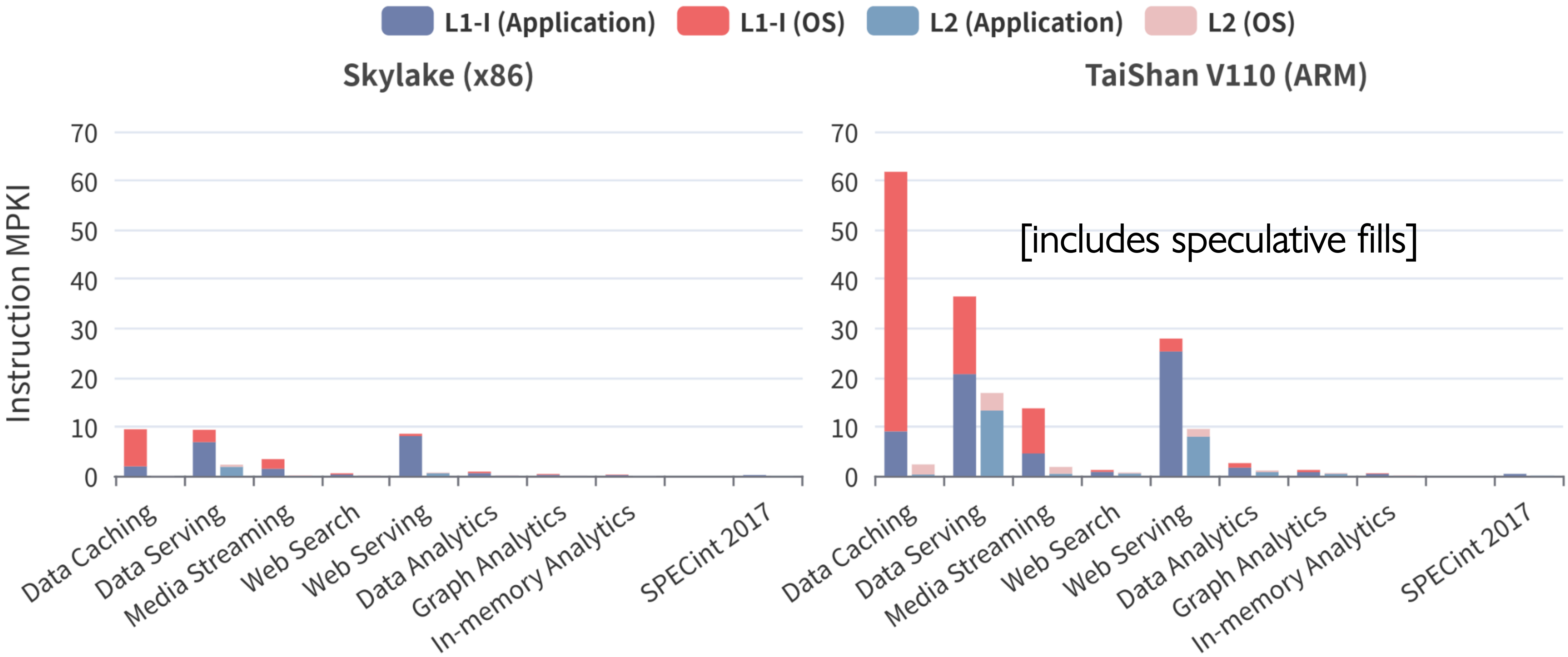
APPLICATION vs. OS CYCLES



Unlike SPEC, server workloads spend more execution time in the OS

INSTRUCTION MPKI

PARALLEL SYSTEMS ARCHITECTURE LAB



Unlike SPEC, server workloads suffer from instruction cache misses at L1-I and L2

HISTORY OF CloudSuite

PARALLEL SYSTEMS ARCHITECTURE LAB

- CloudSuite 1.0: the first release, presented at ASPLOS'12
- CloudSuite 2.0: Data Caching added, published at TOCS'12
- CloudSuite 3.0: Workloads are revisited and offered as Docker containers (Tutorials at EuroSys'16 and DATE'17)
- And now, CloudSuite 4.0 ...

CloudSuite

OUTLINE

PARALLEL SYSTEMS ARCHITECTURE LAB

Part 1: Why CloudSuite?

- Server workloads' benchmarking
- Introducing CloudSuite 4.0

Part 2: Hands-on experience

- CloudSuite on a real machine
  - Tuning the workload
  - Extracting μArch characteristics
- CloudSuite in a full-system emulator (QEMU)
  - Cache hierarchy simulation

CloudSuite 4.0

PARALLEL SYSTEMS ARCHITECTURE LAB

Data Analytics
Machine learning

Graph Analytics
Spark, GraphX

In-Memory Analytics
Recommendation System

Data Caching
Memcached

Data Serving
Cassandra, NoSQL

Media Streaming
Nginx, HTTP Server

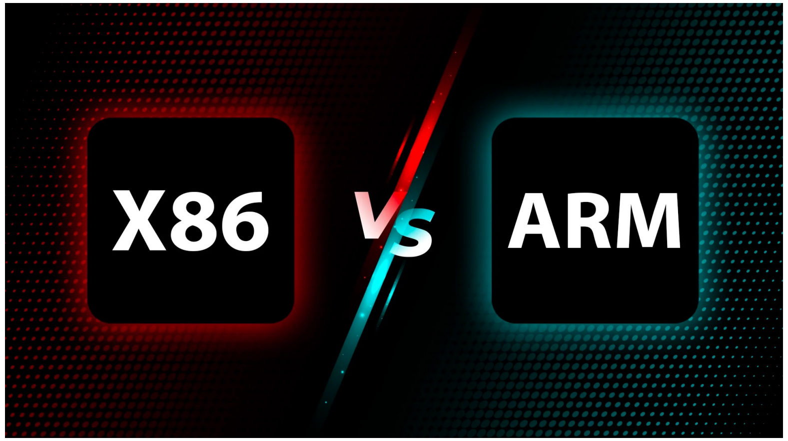
Web Search
Apache Solr & Nutch

Web Serving
Nginx, PHP server

NEW IN CloudSuite 4.0

PARSA PARALLEL SYSTEMS ARCHITECTURE LAB

- Multi-arch support
  - All workloads run on x86 and ARMv8
  - 4 workloads also run on RISC-V
- Software stack updates
  - Ubuntu 22.04
  - OpenJDK 17
  - latest releases of application software
- Ease of use
  - Enhanced docs
  - Useful tuning parameters



OpenJDK

22.04 LTS

TARGET AUDIENCE

- System designers
  - Assess & compare systems' performance for cloud workloads
- Computer architects
  - Derive insights for future server design
- Cloud service providers
  - Measure performance and sustainability

CloudSuite 4.0

PARSA
PARALLEL SYSTEMS
ARCHITECTURE LAB

Data Analytics
Machine learning
Wikipedia · MAHOUT

Graph Analytics
Spark, GraphX
GraphX

In-Memory Analytics
Recommendation System
movielens · Spark

Data Caching
Memcached

Data Serving
Cassandra, NoSQL
Cassandra

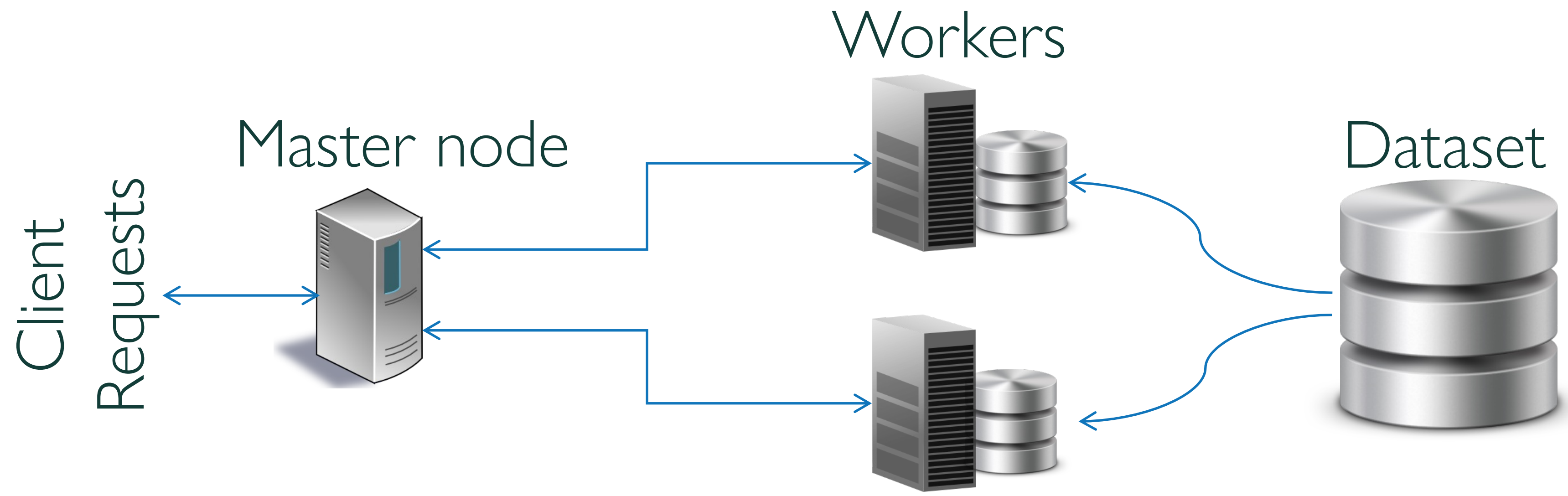
Media Streaming
Nginx, HTTP Server

Web Search
Apache Solr & Nutch
Solr · nutch

Web Serving
Nginx, PHP server
php · MariaDB

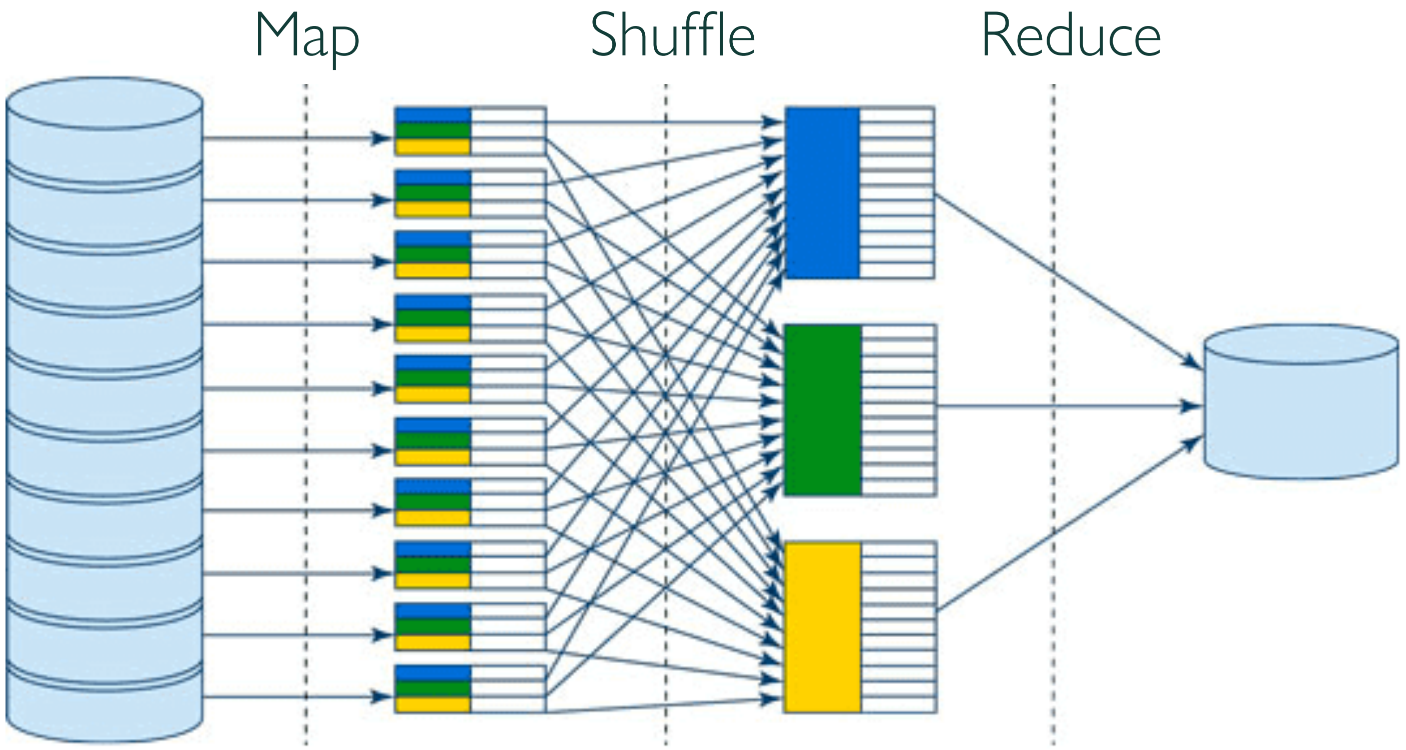
ANALYTICS

- Usually a machine learning algorithm running over large datasets
- No tail latency metric
- Performance metrics
  - Completion time (for a given input size)
  - Throughput (metric is benchmark-/algorithm-specific)



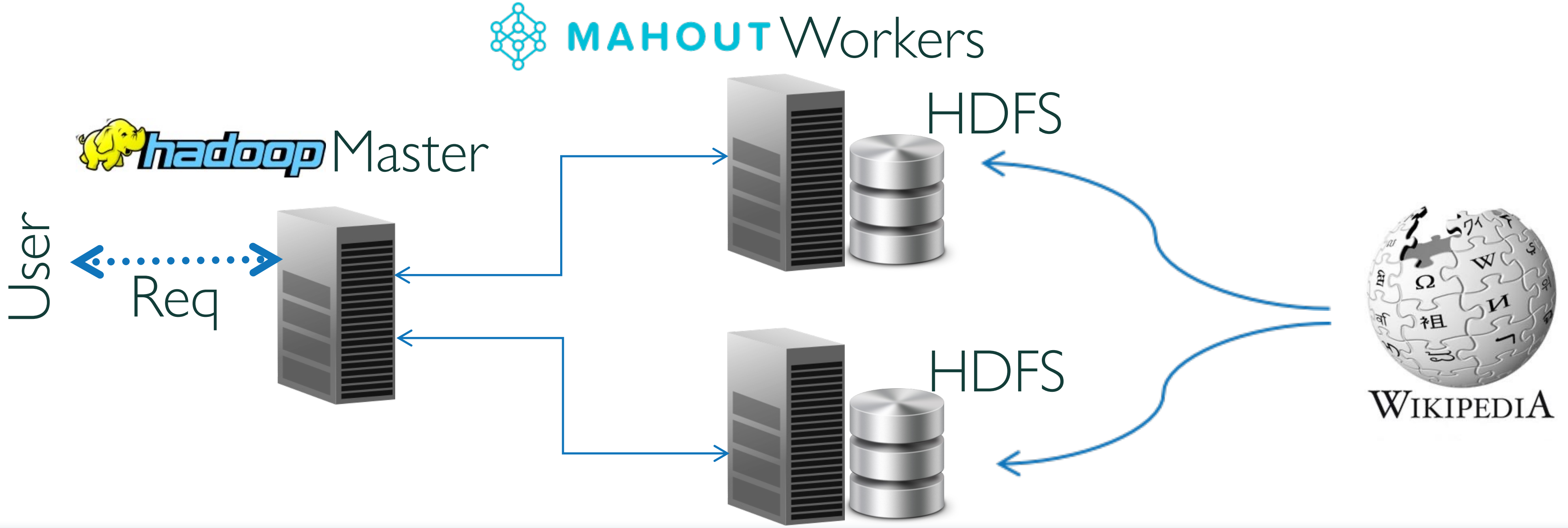
DATA ANALYTICS

- Extract useful information from massive amounts of data (Big Data)
  - Predict user preferences, opinions, behavior
  - Benefit from information (e.g., business, security)
- MapReduce execution model
  - Map: processing small parts
  - Reduce: aggregating the results
- Several examples
  - Song recommendation (Spotify)
  - Spyware detection (Facebook)



DATA ANALYTICS (cont.)

- Application: Text classification based on naïve Bayes classifier
  - Classes: Art, History, Economics, Health, Technology, etc.
- Software: Apache Hadoop and Apache Mahout
- Dataset: Wikipedia English page articles
- Performance metric: # pages classified per unit of time, completion time



CloudSuite 4.0

PARALLEL SYSTEMS ARCHITECTURE LAB

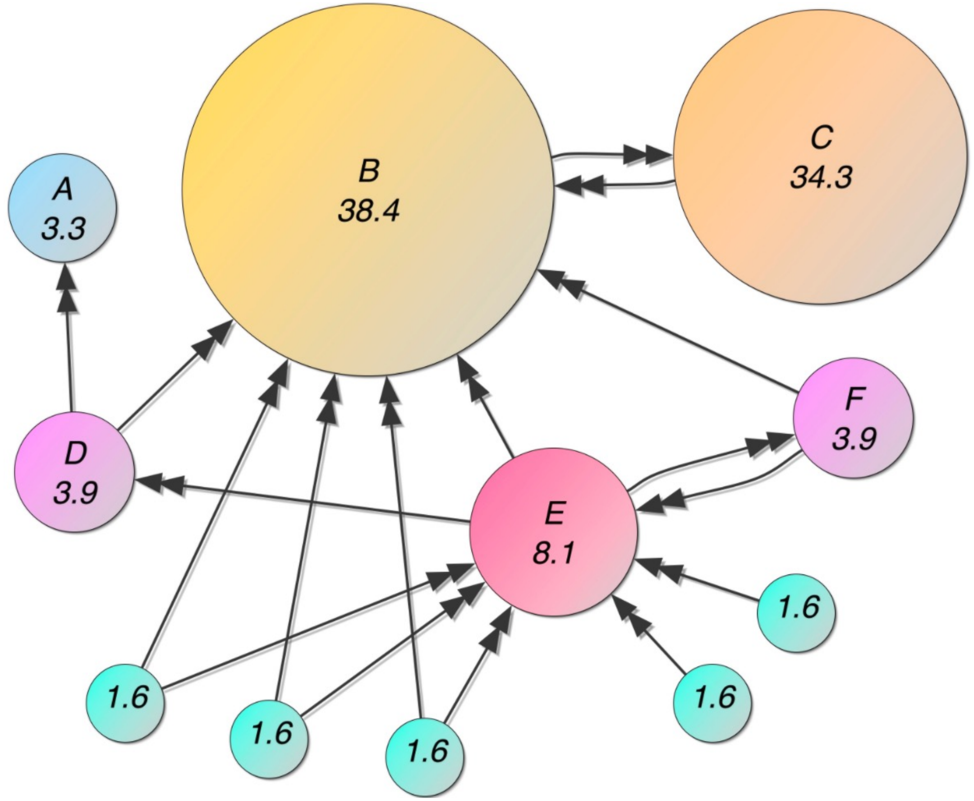
Data Analytics Machine learning	Graph Analytics Spark, GraphX	In-Memory Analytics Recommendation System
Data Caching Memcached	Data Serving Cassandra, NoSQL	
Media Streaming Nginx, HTTP Server	Web Search Apache Solr & Nutch	Web Serving Nginx, PHP server

GRAPH ANALYTICS

- Parallel distributed graph processing
- Data mining on graphs
- Graph examples
  - Social networks (Facebook, Twitter)
  - Web graph
  - Microservices in a datacenter

GRAPH ANALYTICS (cont.)

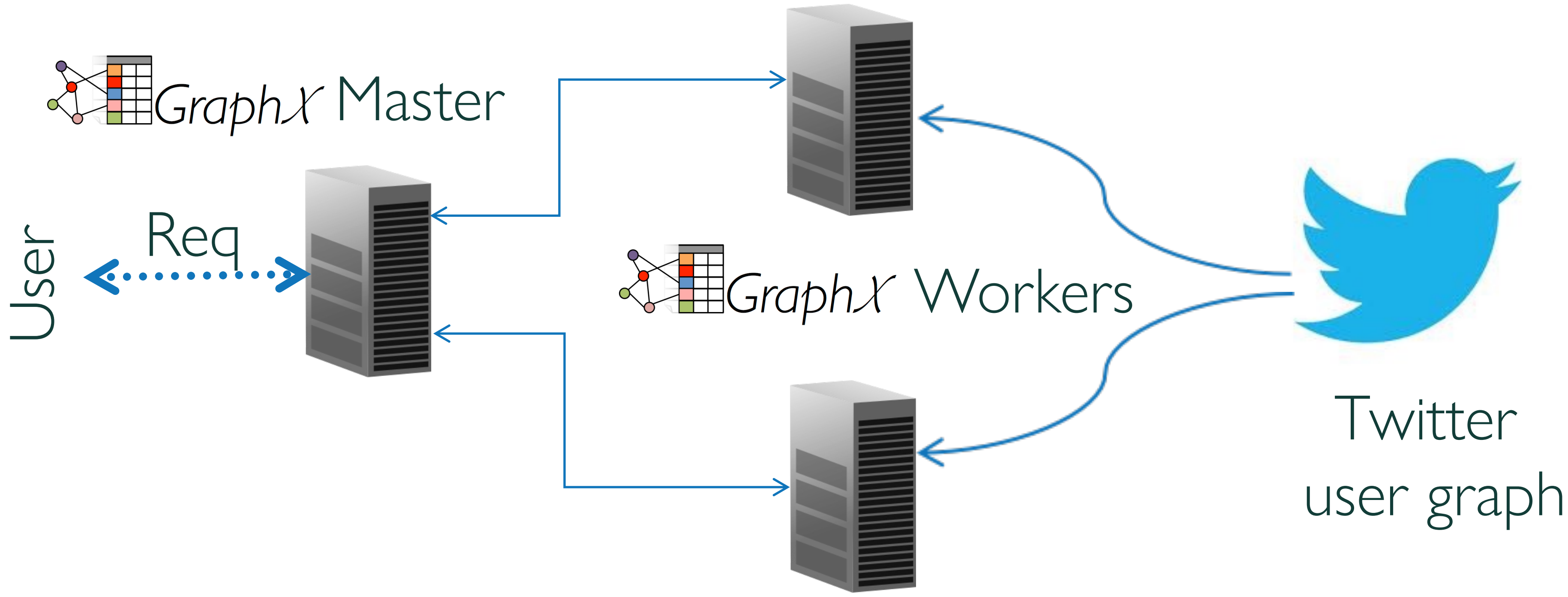
- Application: PageRank
  - Measures influence of Twitter users
  - How much attention followers pay to a user
- Software: Apache Spark, GraphX
  - Parallel framework for graph processing
- Dataset: Twitter user graph



GRAPH ANALYTICS (cont.)

PARALLEL SYSTEMS ARCHITECTURE LAB

- Distributes the graph across nodes
- Iterative computation with adjacent vertices
- Communication across machines for adjacent vertices
- Performance metric: completion time



CloudSuite 4.0

PARALLEL SYSTEMS ARCHITECTURE LAB

Data Analytics
Machine learning

Graph Analytics
Spark, GraphX

In-Memory Analytics
Recommendation System

Data Caching
Memcached

Data Serving
Cassandra, NoSQL

Media Streaming
Nginx, HTTP Server

Web Search
Apache Solr & Nutch

Web Serving
Nginx, PHP server

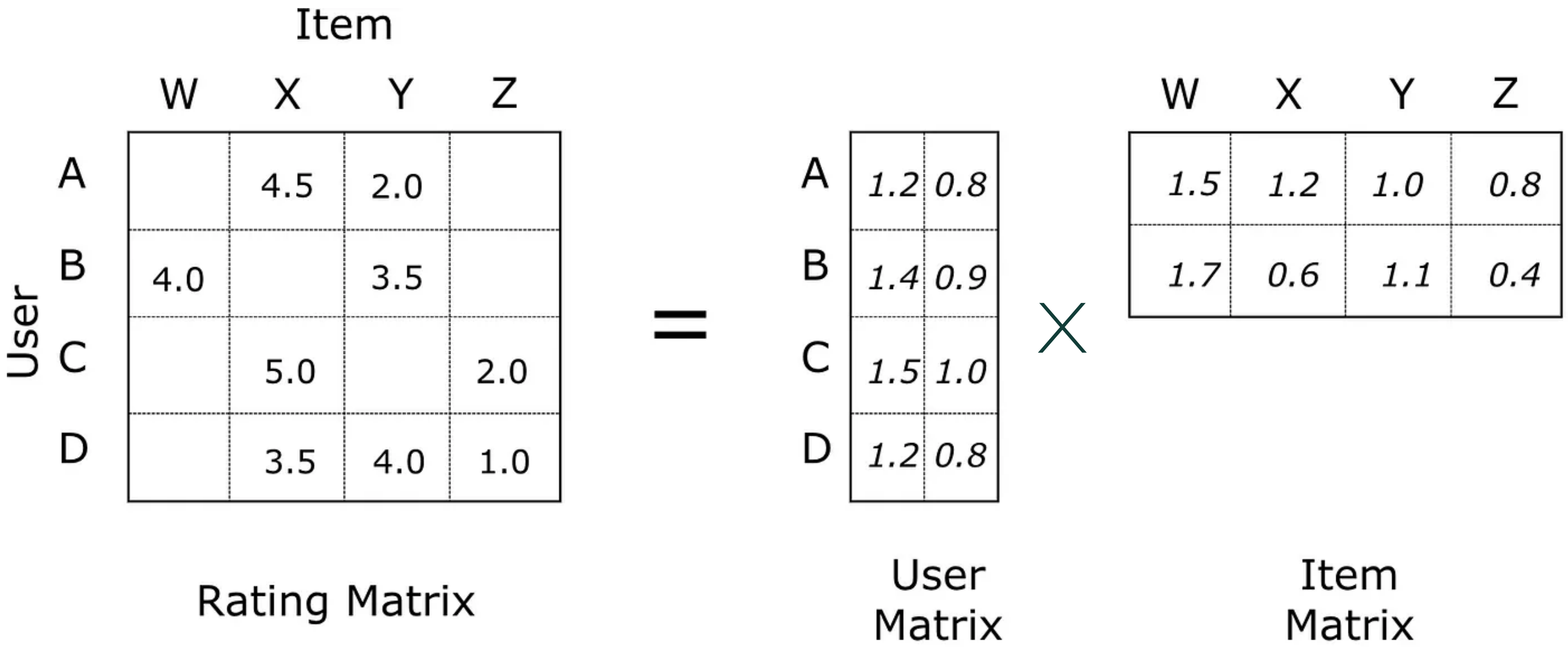
IN-MEMORY ANALYTICS

- In-memory processing of human-generated data
- Extract useful information from users' data
  - Predict users' preferences, rates
- Several examples
  - Movie recommendation (Netflix)
  - Item recommendation (Amazon)
  - Song recommendation (Spotify)
  - Recommending new friends (Social networks)

IN-MEMORY ANALYTICS (cont.)

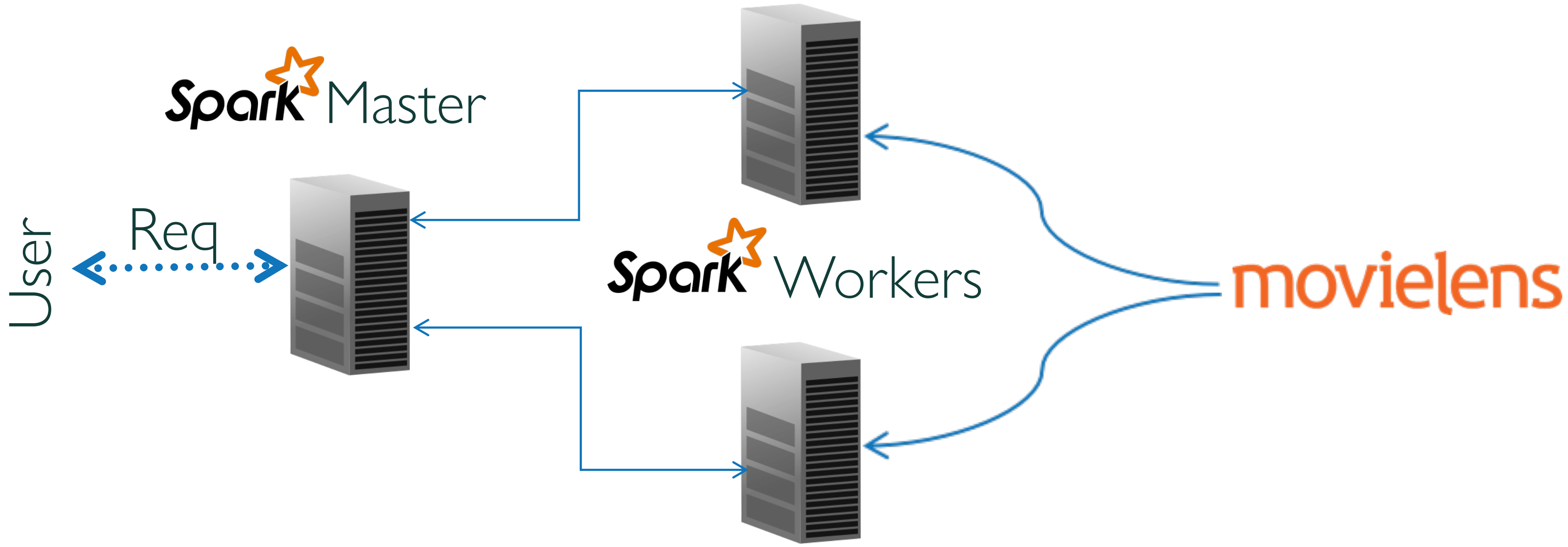
PARALLEL SYSTEMS ARCHITECTURE LAB

- Application: Alternating Least Squares (ALS), used in recommendation systems
- Software: Apache Spark, MLlib
- Dataset: Movielens video dataset



IN-MEMORY ANALYTICS (cont.)

- Trains a recommendation model with the ALS matrix factorization algorithm
- Master partitions user rating matrix and sends them to workers
- Workers perform local matrix factorization and send results to master
- Performance metric: completion time for factorizing the rating matrix



CloudSuite 4.0

Data Analytics
Machine learning

Graph Analytics
Spark, GraphX

In-Memory Analytics
Recommendation System

Data Caching
Memcached

Data Serving
Cassandra, NoSQL

Media Streaming
Nginx, HTTP Server

Web Search
Apache Solr & Nutch

Web Serving
Nginx, PHP server

ONLINE SERVICES

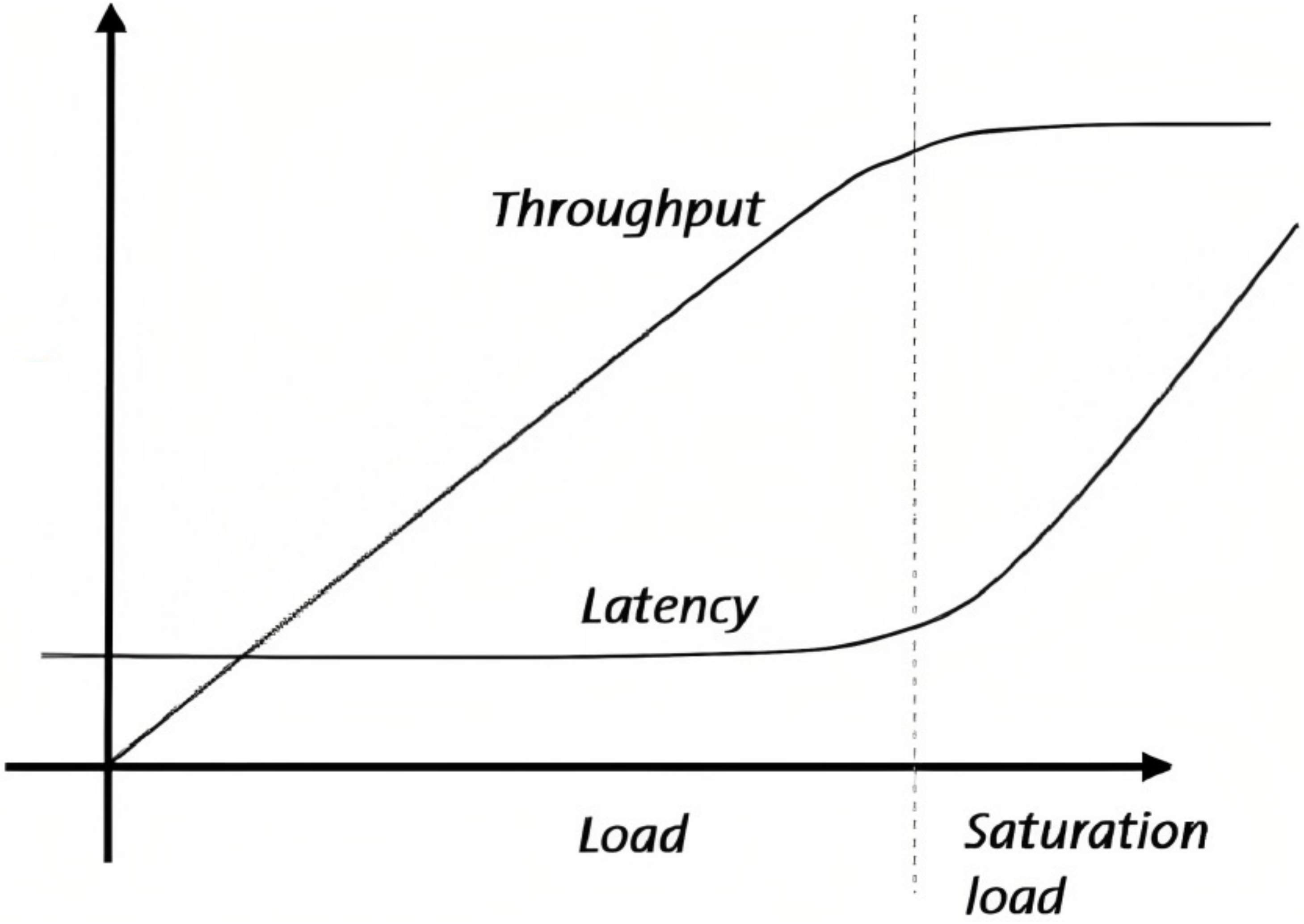
PARALLEL SYSTEMS ARCHITECTURE LAB

- Operate on large datasets

- Throughput is important, but also need high service quality
  - Tail latency of requests is critical for service quality
  - Goal: Maximizing throughput *under Service-Level Objective (SLO)*

- Performance metrics
  - Throughput (metric is benchmark-specific)
  - Latency (expressed in terms of the N-th percentile tail latency)

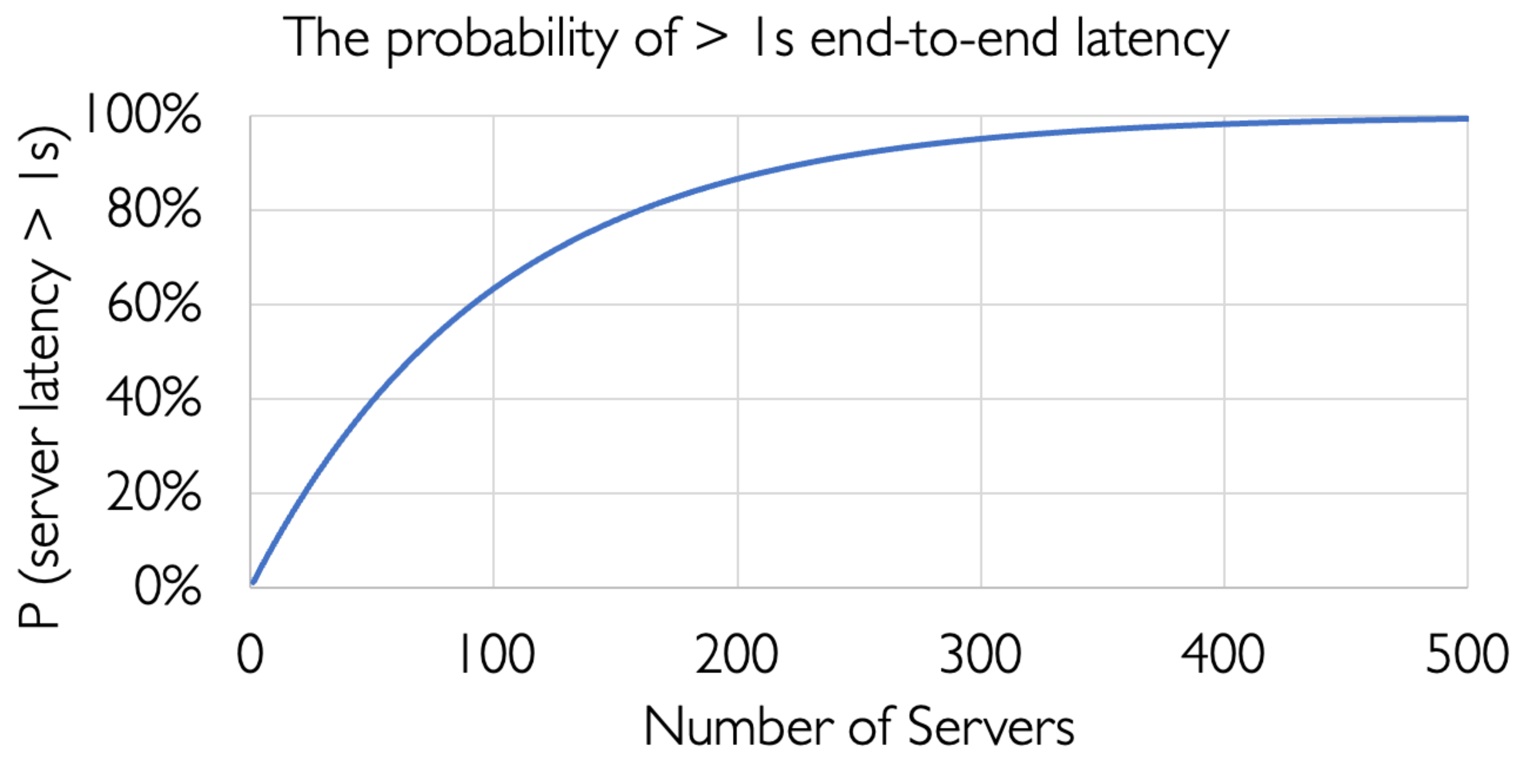
THROUGHPUT vs. LATENCY



Online services target finding the maximum throughput under SLO

TAIL LATENCY

- Slowest response times affect the end-to-end QoS [Dean, CACM'13]
- Tail latency is usually 95, 99, or 99.9 percentile of the requests' latency



Performance hiccups have to be rare in large-scale distributed systems

DATA CACHING

PARALLEL SYSTEMS ARCHITECTURE LAB

- Online services are often latency-sensitive
- Fetching data from disk is slow
- Data is cached in memory for fast data access
  - General-purpose, in-memory key-value store
  - Caches data for other apps, another tier before back-end

DATA CACHING (cont.)

PARALLEL SYSTEMS ARCHITECTURE LAB

- Application: Memcached
  - High throughput objects retrieval
  - Free & open-source, high-performance, distributed object caching system

- Dataset: Twitter object popularity dataset
  - Keys' distribution and their values' sizes
  - Configurable size of the dataset

- Performance metric: # req/s, under SLO (e.g., 1ms)

CloudSuite 4.0

PARALLEL SYSTEMS ARCHITECTURE LAB

Data Analytics
Machine learning
WIKIPEDIA, MAHOUT

Graph Analytics
Spark, GraphX
GraphX

In-Memory Analytics
Recommendation System
movielens, Spark

Data Caching
Memcached

Data Serving
Cassandra, NoSQL
Cassandra

Media Streaming
Nginx, HTTP Server

Web Search
Apache Solr & Nutch
Solr, nutch

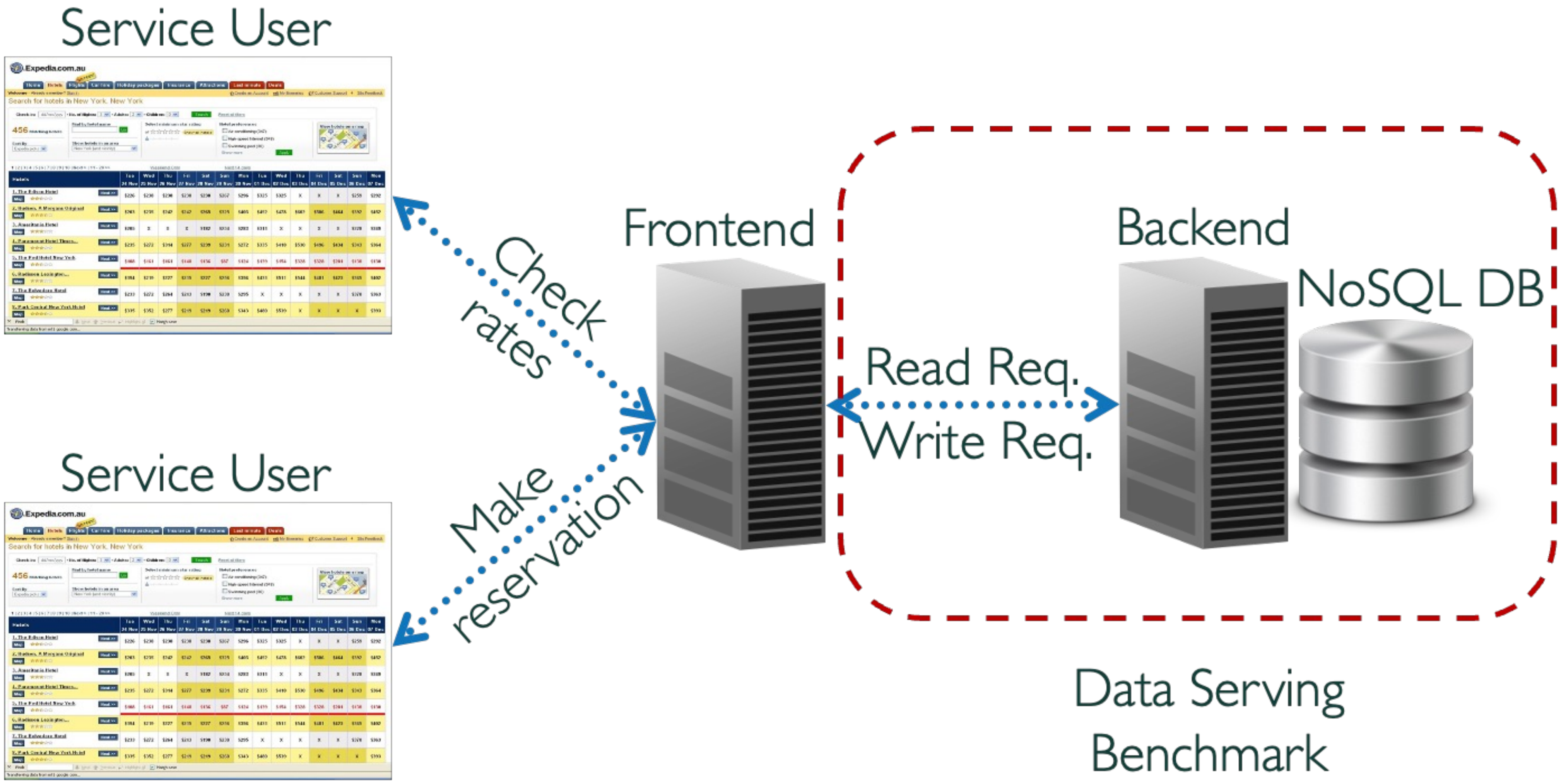
Web Serving
Nginx, PHP server
php, MariaDB

DATA SERVING

- Global-scale online services rely on NoSQL databases
  - Inherently scalable
  - Suitable for unpredictable schema changes
- Scale out to meet service requirements
  - Accommodate fast data generation rate

DATA SERVING (cont.)

PARALLEL SYSTEMS ARCHITECTURE LAB

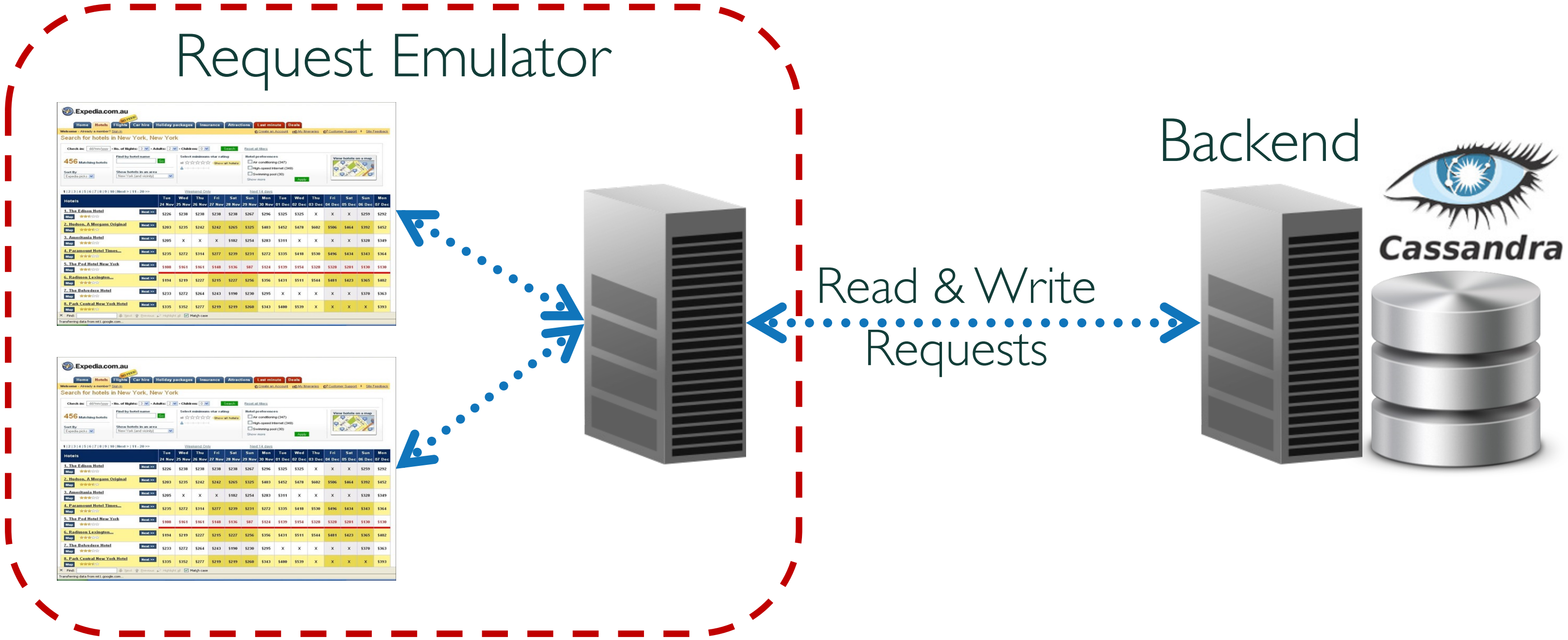


DATA SERVING (cont.)

- Application: Apache Cassandra
  - A popular NoSQL database: many use cases (e.g., Expedia, eBay, Netflix)
- Performance metric: # R/W ops/s under SLO (e.g., ~10ms)

DATA SERVING (cont.)

- Database: generated and populated by YCSB
  - Defines number & size of fields, and total entries
- Predefined mixes of read/write operations
- Popularity of access distributions (e.g., Zipfian)



CloudSuite 4.0

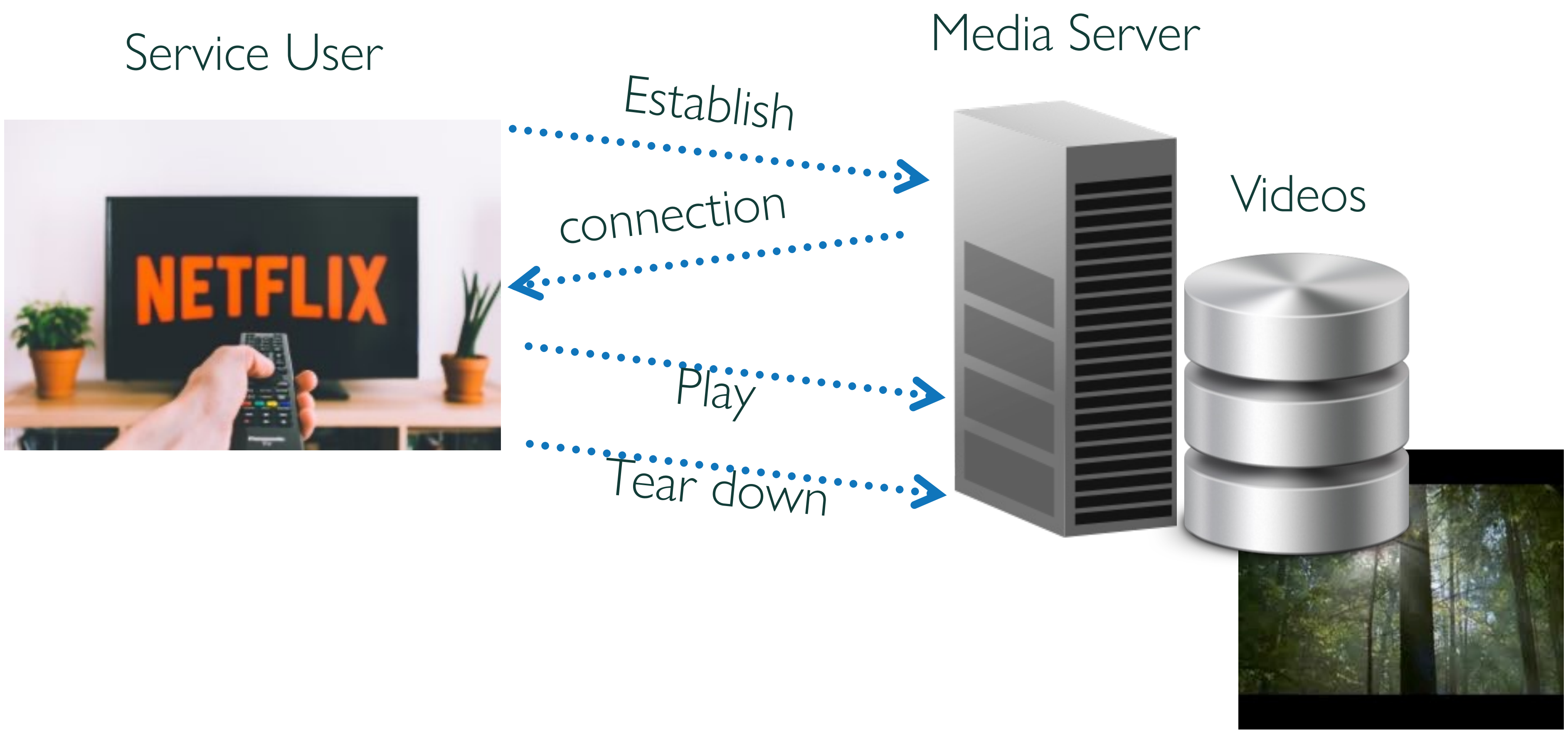
PARALLEL SYSTEMS ARCHITECTURE LAB

Data Analytics Machine learning	Graph Analytics Spark, GraphX	In-Memory Analytics Recommendation System
Data Caching Memcached	Data Serving Cassandra, NoSQL	
Media Streaming Nginx, HTTP Server	Web Search Apache Solr & Nutch	Web Serving Nginx, PHP server

MEDIA STREAMING

- Media streaming expected to dominate internet traffic
  - Around 80% of global web traffic [globenewswire]
- Increasing popularity of media streaming services
  - Video sharing sites, movie streaming services, podcasts, etc.
- Fast and low latency Internet access
  - High quality media streaming even on affordable devices

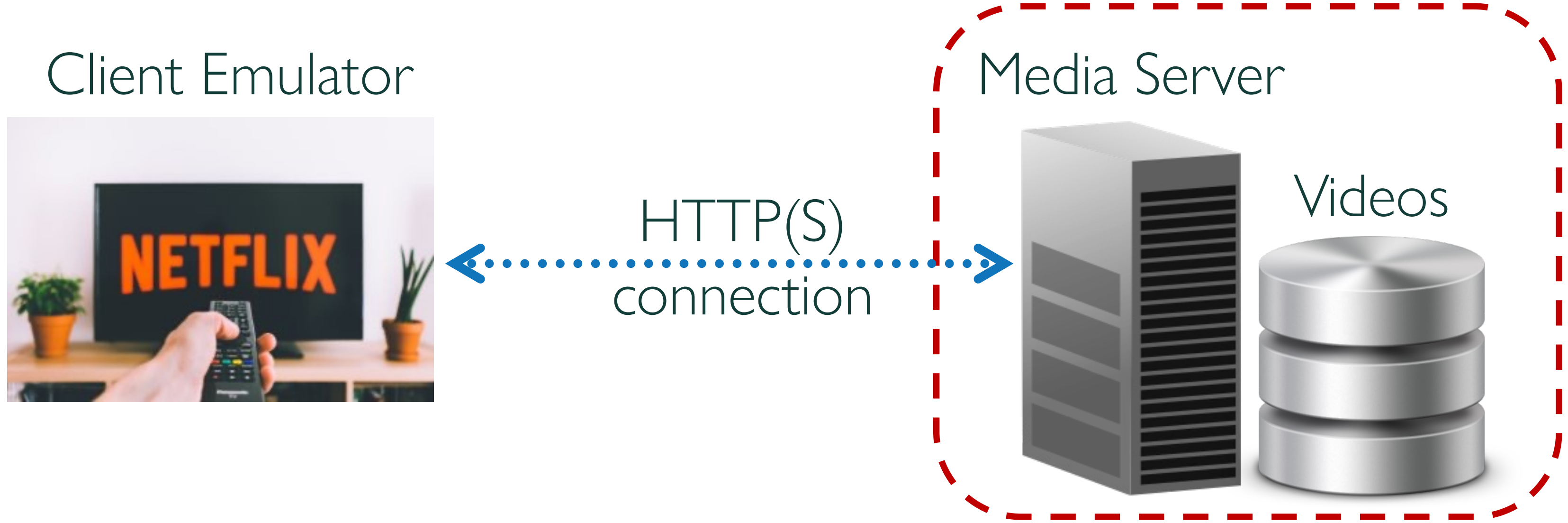
MEDIA STREAMING (cont.)



MEDIA STREAMING (cont.)

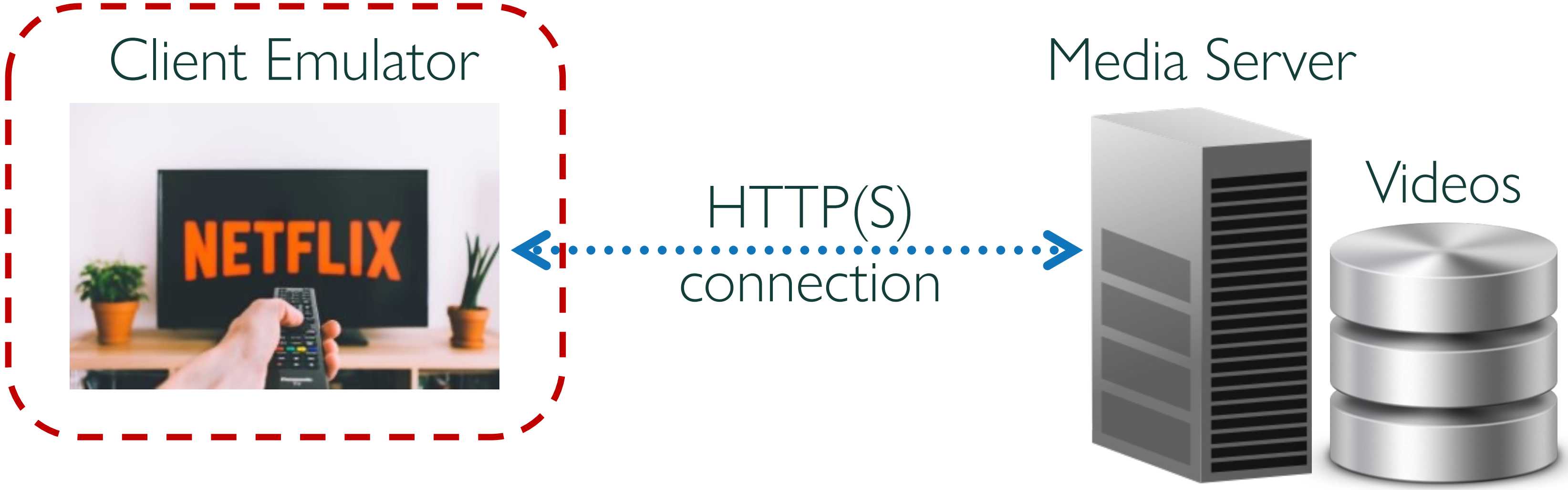
PARALLEL SYSTEMS ARCHITECTURE LAB

- Application: Nginx server with TLSv1.3 enabled
- Dataset: mix of pre-encoded videos
  - Configurable dataset size
  - Four video resolutions (240p, 360p, 480p, 720p)
  - Zipfian popularity distribution
- Performance metrics: utilized bandwidth (Kbps) without connection timeout



MEDIA STREAMING (cont.)

- Imitates real video streaming users' behavior
  - Requests for video chunks with delay
  - Different video lengths and resolutions
- Implements HTTPS connection with TLSv1.3
- Uses the videoperf client, based on the httperf traffic generator



CloudSuite 4.0

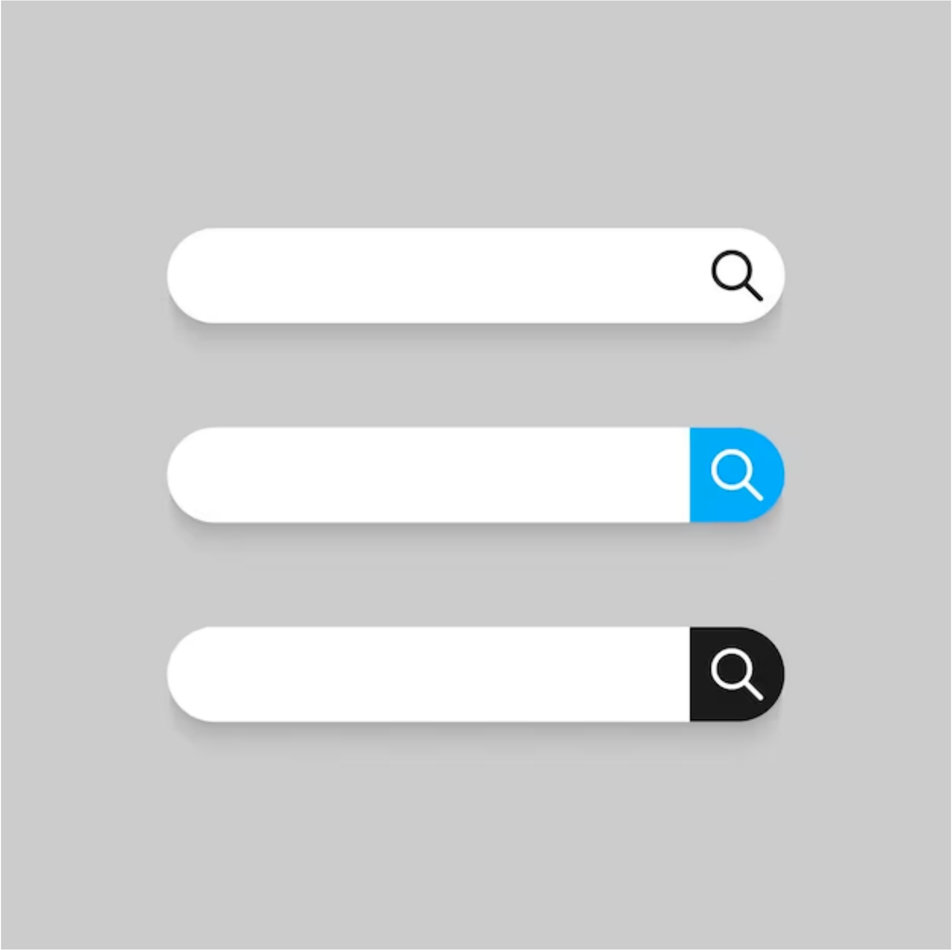
PARALLEL SYSTEMS ARCHITECTURE LAB

Data Analytics Machine learning	Graph Analytics Spark, GraphX	In-Memory Analytics Recommendation System
Data Caching Memcached	Data Serving Cassandra, NoSQL	
Media Streaming Nginx, HTTP Server	**Web Search** **Apache Solr & Nutch**	**Web Serving** **Nginx, PHP server**

WEB SEARCH

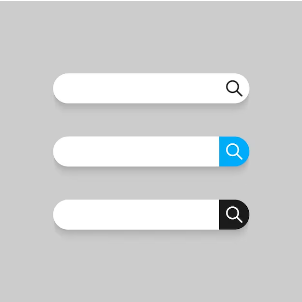
PARALLEL SYSTEMS ARCHITECTURE LAB

- Crucial feature for online services
  - High volume of queries
  - Quick response time (< 1s)
  - Accurate results
- Embedded search appears in almost all websites
  - Service's profit and users' experience

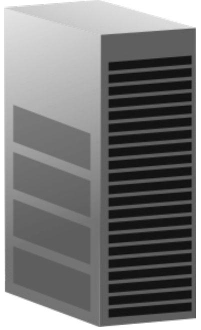


WEB SEARCH (cont.)

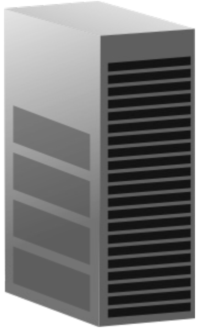
User



Frontend

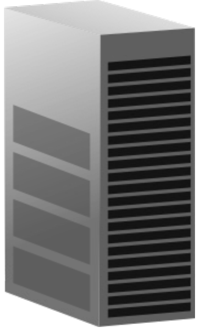


Index Serving Node (ISN)



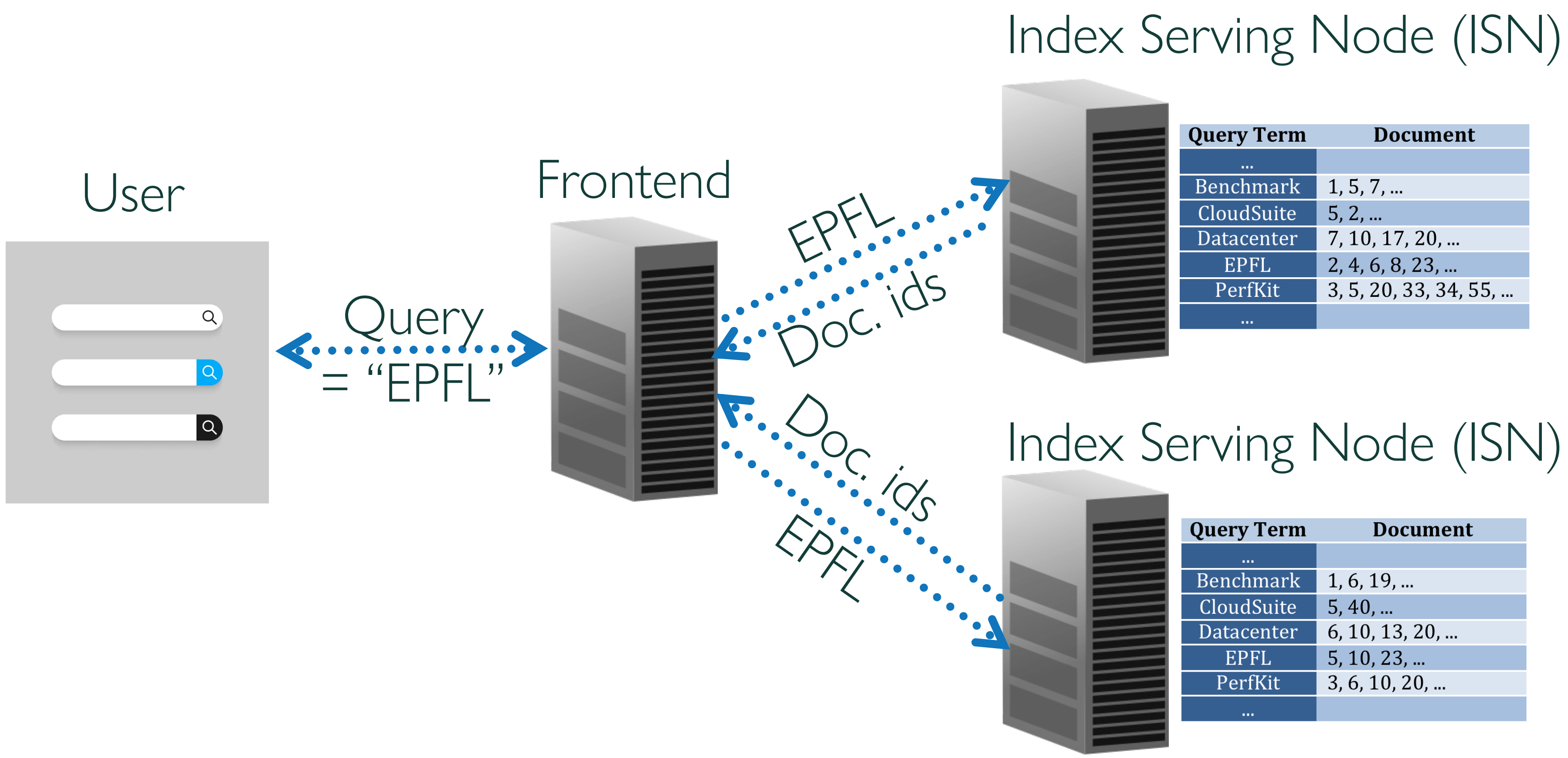
Query Term	Document
...	
Benchmark	1, 5, 7, ...
CloudSuite	5, 2, ...
Datacenter	7, 10, 17, 20, ...
EPFL	2, 4, 6, 8, 23, ...
PerfKit	3, 5, 20, 33, 34, 55, ...
...	

Index Serving Node (ISN)



Query Term	Document
...	
Benchmark	1, 6, 19, ...
CloudSuite	5, 40, ...
Datacenter	6, 10, 13, 20, ...
EPFL	5, 10, 23, ...
PerfKit	3, 6, 10, 20, ...
...	

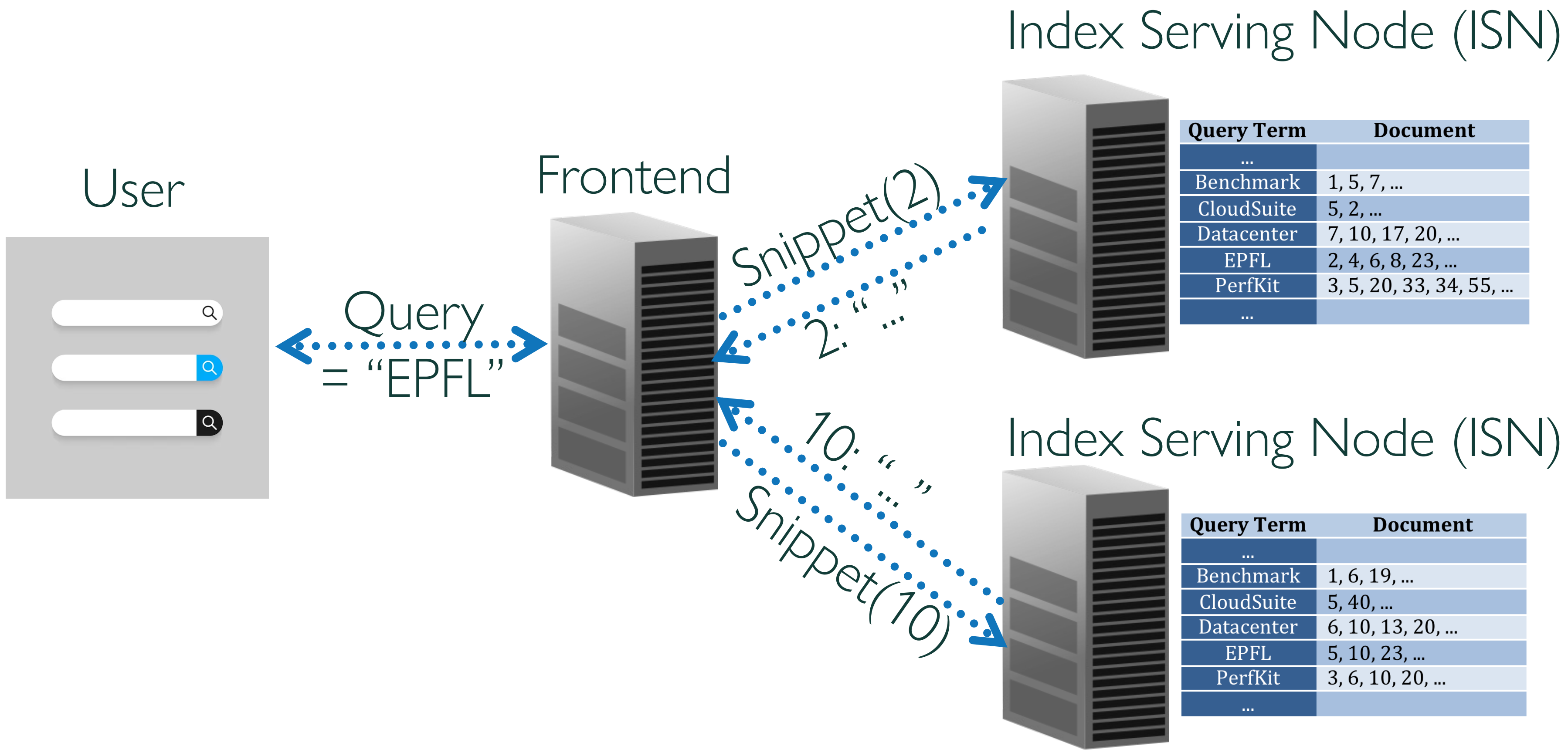
WEB SEARCH (cont.)



Query Term	Document
...	
Benchmark	1, 5, 7, ...
CloudSuite	5, 2, ...
Datacenter	7, 10, 17, 20, ...
EPFL	2, 4, 6, 8, 23, ...
PerfKit	3, 5, 20, 33, 34, 55, ...
...	

Query Term	Document
...	
Benchmark	1, 6, 19, ...
CloudSuite	5, 40, ...
Datacenter	6, 10, 13, 20, ...
EPFL	5, 10, 23, ...
PerfKit	3, 6, 10, 20, ...
...	

WEB SEARCH (cont.)



Query Term	Document
...	
Benchmark	1, 5, 7, ...
CloudSuite	5, 2, ...
Datacenter	7, 10, 17, 20, ...
EPFL	2, 4, 6, 8, 23, ...
PerfKit	3, 5, 20, 33, 34, 55, ...
...	

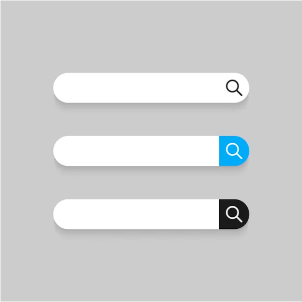
Query Term	Document
...	
Benchmark	1, 6, 19, ...
CloudSuite	5, 40, ...
Datacenter	6, 10, 13, 20, ...
EPFL	5, 10, 23, ...
PerfKit	3, 6, 10, 20, ...
...	

WEB SEARCH (cont.)

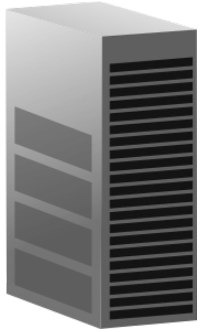
PAISA PARALLEL SYSTEMS ARCHITECTURE LAB

- Application: Apache Solr search engine for ISNs
- Dataset: Inverted index & snippets
  - Crawled with Apache Nutch
  - Indexed with Apache Lucene
- Performance metric: # queries/sec under SLO (e.g., 200ms)

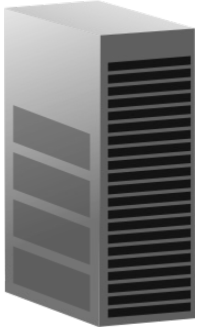
User



Frontend

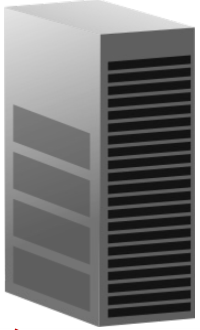


Index Serving Node (ISN)



Query Term	Document
...	
Benchmark	1, 5, 7, ...
CloudSuite	5, 2, ...
Datacenter	7, 10, 17, 20, ...
EPFL	2, 4, 6, 8, 23, ...
PerfKit	3, 5, 20, 33, 34, 55, ...
...	

Index Serving Node (ISN)

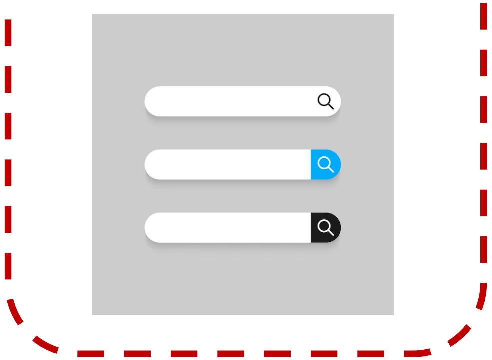


Query Term	Document
...	
Benchmark	1, 6, 19, ...
CloudSuite	5, 40, ...
Datacenter	6, 10, 13, 20, ...
EPFL	5, 10, 23, ...
PerfKit	3, 6, 10, 20, ...
...	

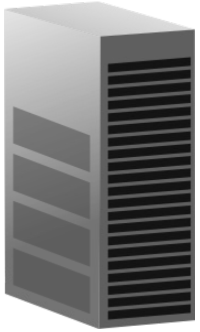
WEB SEARCH (cont.)

- Faban traffic generator
- Flexible request mixes
  - # terms per request from published surveys
  - Terms extracted from the crawled dataset

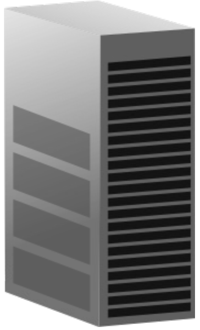
User



Frontend

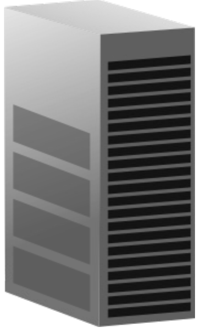


Index Serving Node (ISN)



Query Term	Document
…	
Benchmark	1, 5, 7, …
CloudSuite	5, 2, …
Datacenter	7, 10, 17, 20, …
EPFL	2, 4, 6, 8, 23, …
PerfKit	3, 5, 20, 33, 34, 55, …
…	

Index Serving Node (ISN)



Query Term	Document
…	
Benchmark	1, 6, 19, …
CloudSuite	5, 40, …
Datacenter	6, 10, 13, 20, …
EPFL	5, 10, 23, …
PerfKit	3, 6, 10, 20, …
…	

CloudSuite 4.0

PARALLEL SYSTEMS ARCHITECTURE LAB

Data Analytics
Machine learning
WIKIPEDIA MAHOUT

Graph Analytics
Spark, GraphX
GraphX

In-Memory Analytics
Recommendation System
movielens Spark

Data Caching
Memcached

Data Serving
Cassandra, NoSQL
Cassandra

Media Streaming
Nginx, HTTP Server

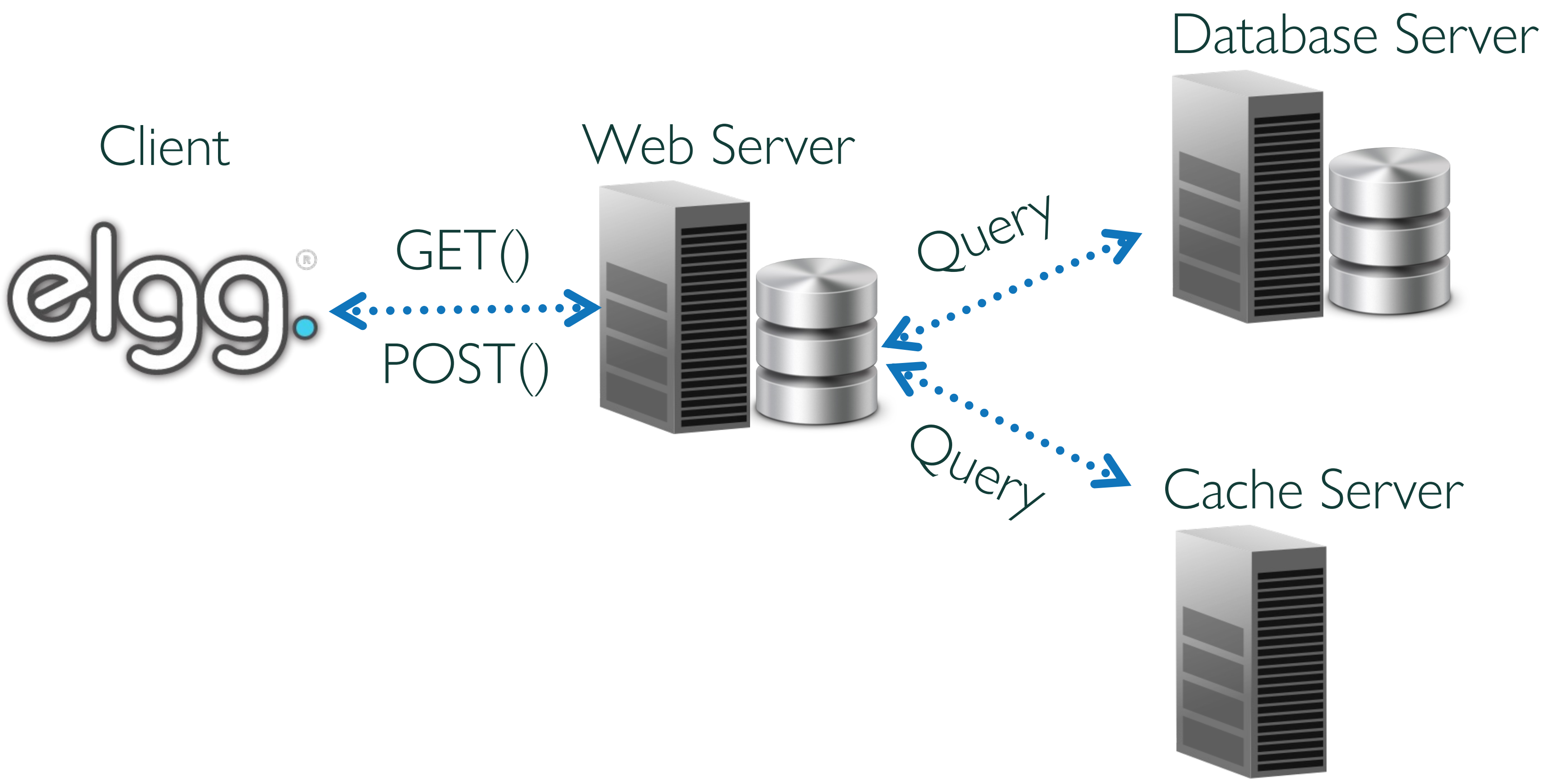
Web Search
Apache Solr & Nutch
Solr nutch

Web Serving
Nginx, PHP server
php MariaDB

WEB SERVING

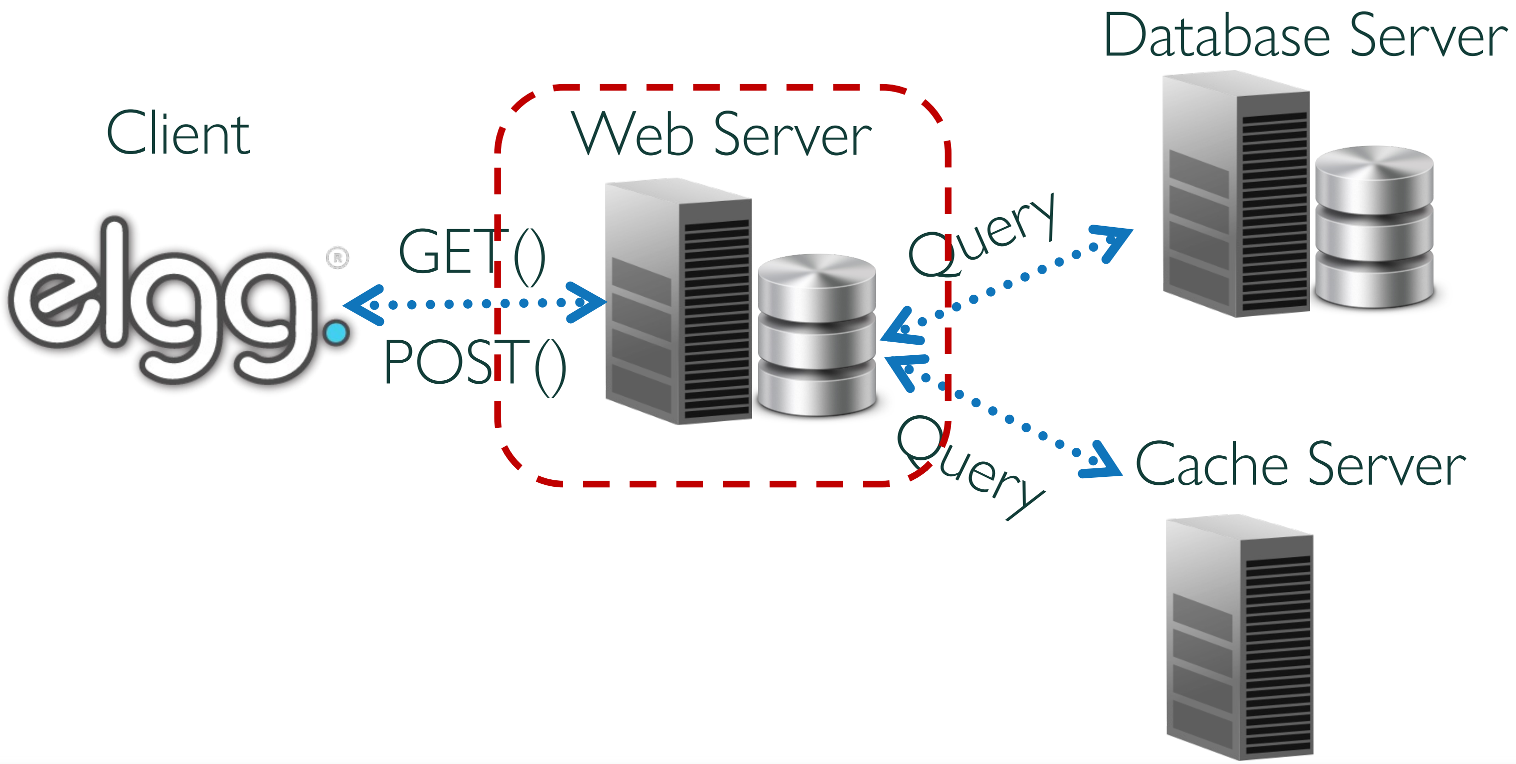
- Key to all internet-based services
- All services are accessed through web servers
- Various technologies construct web content

WEB SERVING (cont.)



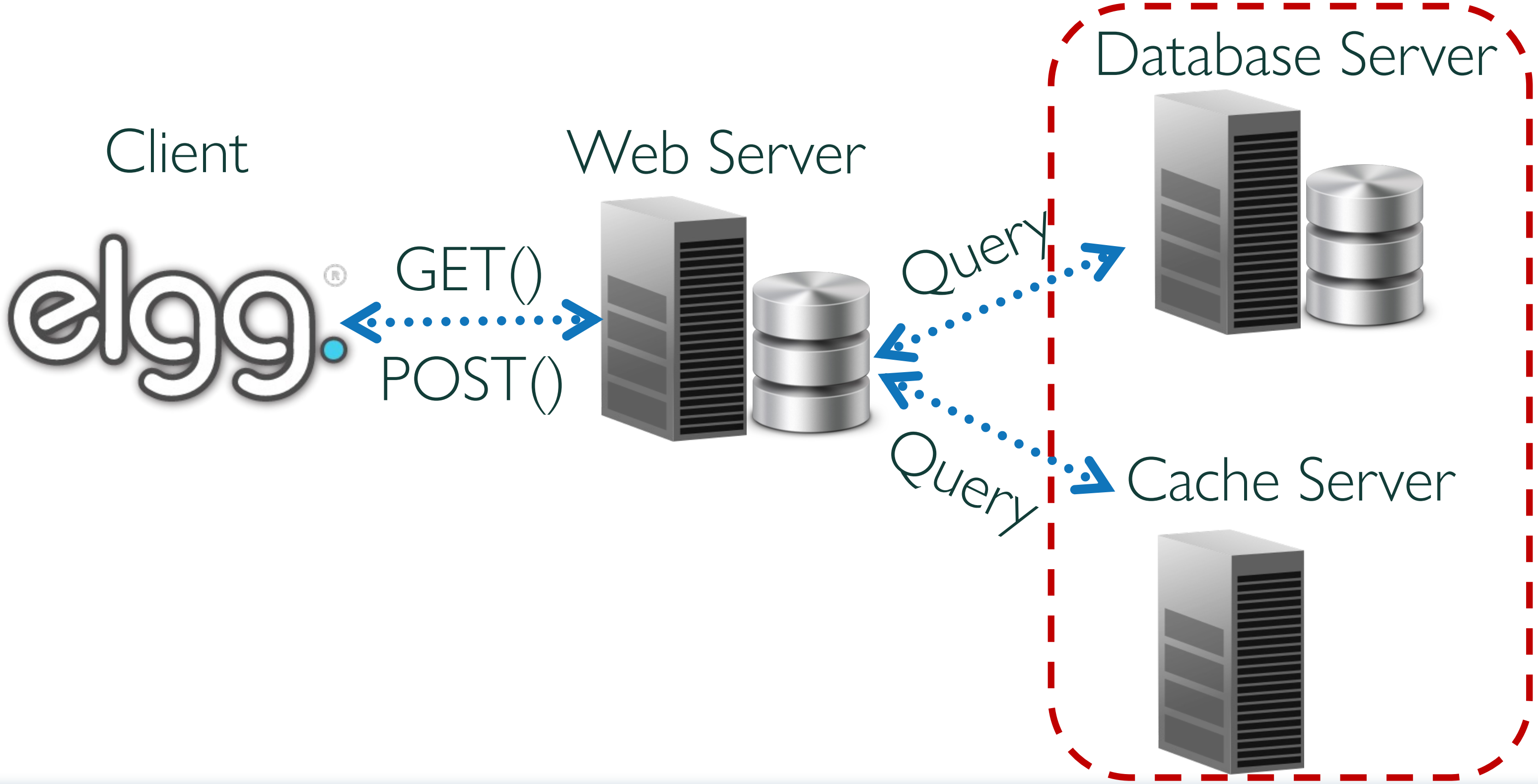
WEB SERVING (cont.)

- Application: Elgg, a social network engine running with PHP
- Web server: Nginx with TLSv1.3



WEB SERVING (cont.)

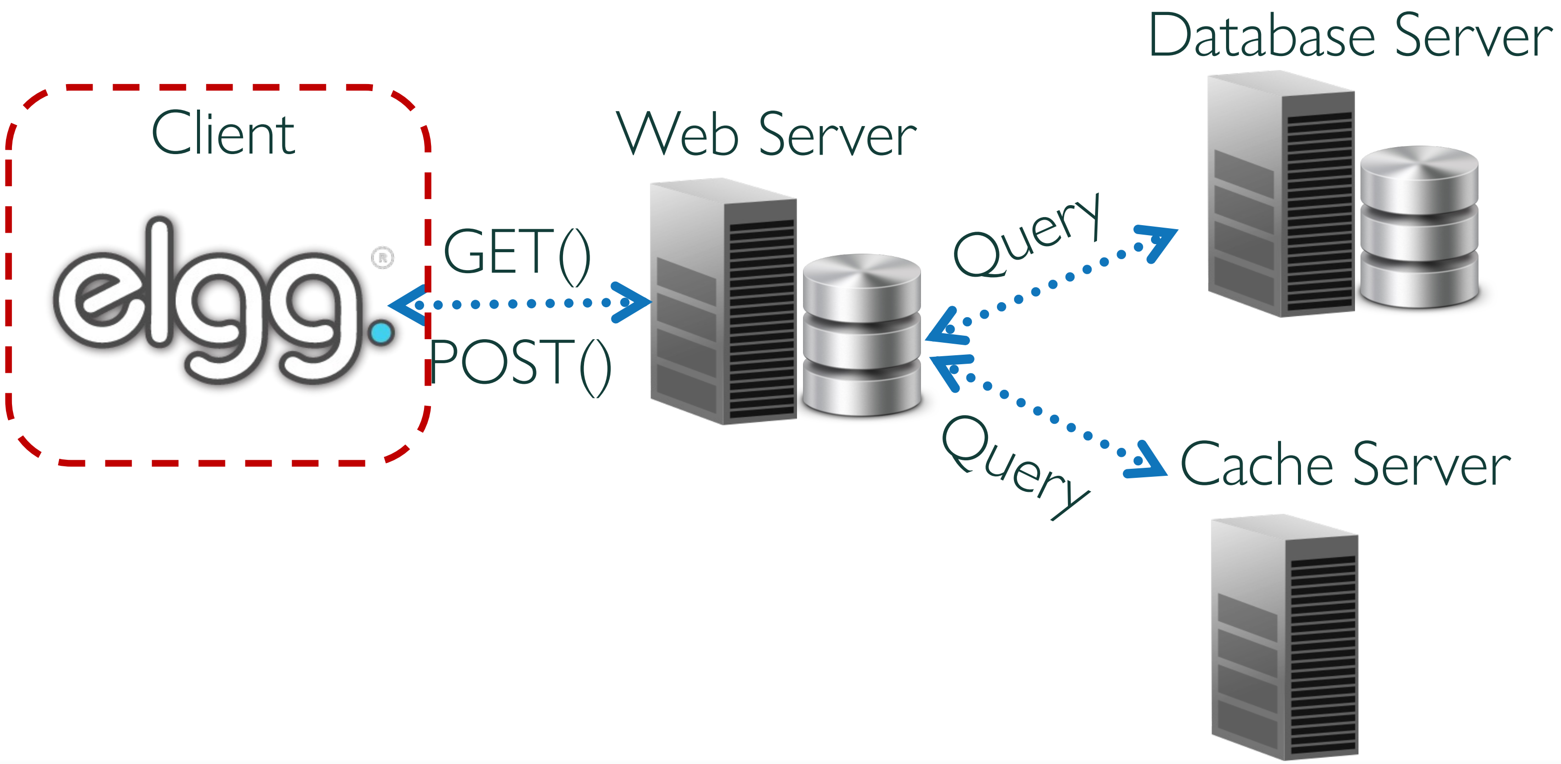
- Database: MariaDB
  - 100 K users with friends, messages, posts, etc.
- Cache server: Memcached
- Performance metric: # pages/s under SLO (e.g., 1s)



WEB SERVING (cont.)

PARSA PARALLEL SYSTEMS ARCHITECTURE LAB

- Faban traffic generator
- Pre-configured page transition matrix for around 30 request types
  - Friend request, posting a blog, status update, sending private messages, etc.



OUTLINE

Part 1: Why CloudSuite?

- Server workloads' benchmarking
- Introducing CloudSuite 4.0

Part 2: Hands-on experience

- CloudSuite on a real machine
  - Tuning the workload
  - Extracting µArch characteristics
- CloudSuite in a full-system emulator (QEMU)
  - Cache hierarchy simulation

AWS EC2 NODES

- Download the `CloudSuite-ASPLOS2023.pem` from cloudsuite.ch/asplos23-tutorial/
- Connect to the entry point:
  - chmod 0400 CloudSuite-ASPLOS2023.pem
  - ssh -i CloudSuite-ASPLOS2023.pem ubuntu@<ip address given in tutorial>
- Connect to your personal node:
  - ./connect.sh <my_number>
  - Run `tmux`
- Check that both folders part01 and part02 are available

COFFEE BREAK

PARALLEL SYSTEMS ARCHITECTURE LAB

We have provided AWS EC2 nodes for Part 2.
Please make sure everyone is connected to their node.

See you in a bit...

CloudSuite

OUTLINE

PARALLEL SYSTEMS ARCHITECTURE LAB

Part 1: Why CloudSuite?

- Server workloads' benchmarking
- Introducing CloudSuite 4.0

Part 2: Hands-on experience

- CloudSuite on a real machine
  - Tuning the workload
  - Extracting µArch characteristics
- CloudSuite in a full-system emulator (QEMU)
  - Cache hierarchy simulation

METHODOLOGY

PARSA PARALLEL SYSTEMS ARCHITECTURE LAB

AWS Machine

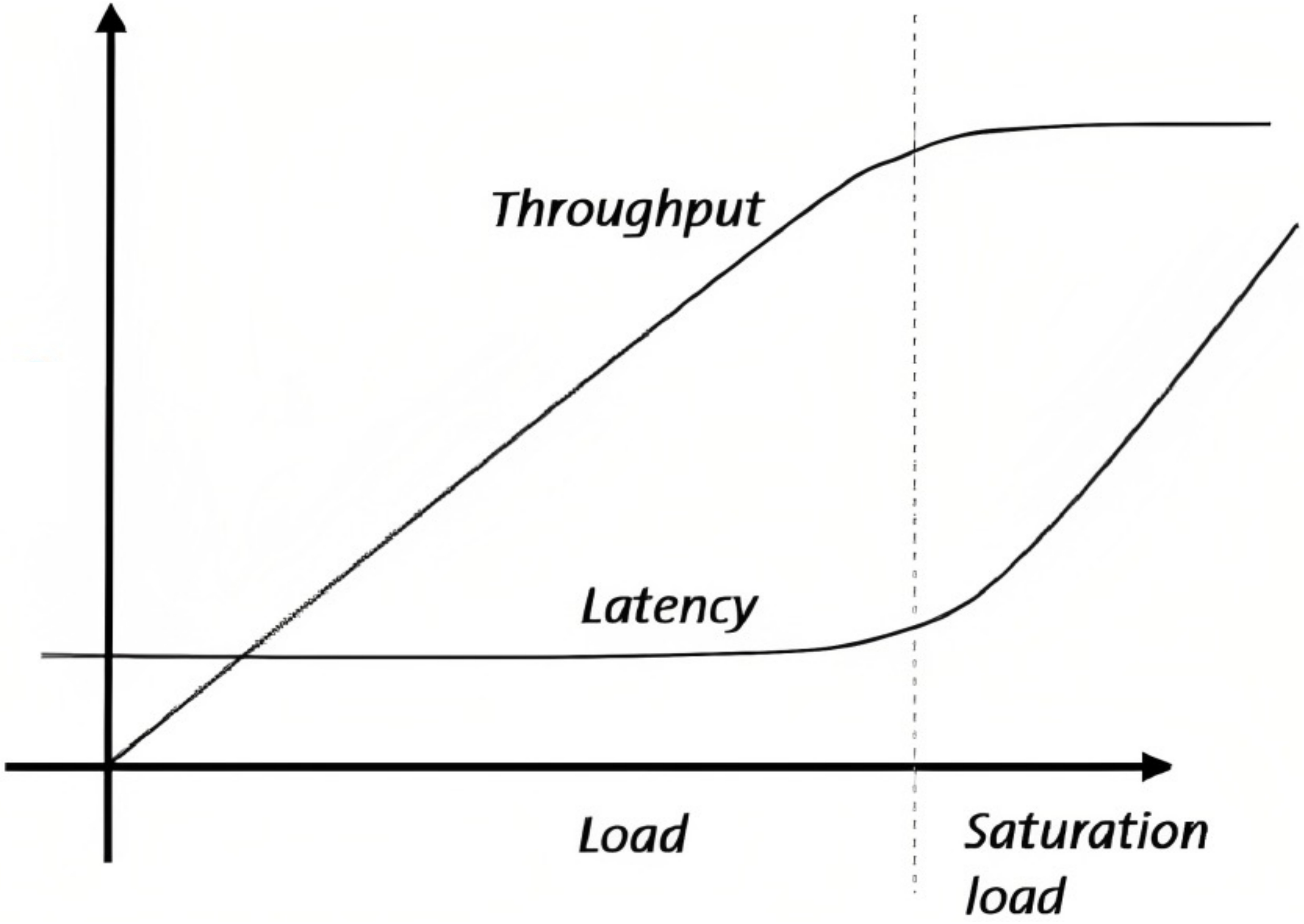
- Xeon Platinum 8124M processor
- **Missing PMUs**
- Release year: **2017**
- Skylake µArch – 36 cores, 1-socket
  - 32 KB L1-I, 48 KB L1-D
  - 1 MB L2
  - 24.8 MB LLC
- Ubuntu 22.04
- Perf Tool 5.15.0-1031-aws

PARSA Machine

- Xeon Gold 6338N processor
- **All PMUs**
- Release year: **2021**
- Ice Lake µArch – 64-cores, 2-sockets
  - 32 KB L1-I, 48 KB L1-D
  - 1.25 MB L2
  - 48 MB LLC
- Ubuntu 22.04
- Perf Tool 5.19.0-35-generic

THROUGHPUT vs. LATENCY (recap)

PARALLEL SYSTEMS ARCHITECTURE LAB



Online services target finding the maximum throughput under SLO

RUNNING THE DATA CACHING WORKLOAD

- Pull server image and start server container
- Pull client image and start client container
- Scale dataset from base Twitter dataset
  - Consider a 10 GB dataset in memory [Shao, CASCON'05]
- Warm up server
  - The client brings the dataset to server's memory
- Load server
  - Set *request per second (RPS)* parameter and measure throughput and latency

- Run `./01-data-caching-launch.sh`

WARMED UP

PARALLEL SYSTEMS ARCHITECTURE LAB

Warmed up

10 GB dataset

```
Outstanding requests per worker:
6963 6196 7074 7543 7363 6154 7152 7454
You are warmed up, sir

--------------------------------      DONE      ------------------------------
ubuntu@ip-172-31-30-253:~/part01-realsystem$
ubuntu@ip-172-31-30-253:~/part01-realsystem$ free -h
              total        used        free      shared  buff/cache   available
Mem:           68Gi        10Gi        48Gi       1.0Mi       9.5Gi        57Gi
Swap:            0B          0B          0B
```

TUNING (cont.)

- AWS Skylake same socket

Load (RPS)	Throughput (RPS)	99th %tile (ms)	CPU util.	Queueing (req. per worker)
20K				
40K				
100K				
142K				
170K				
372K				
1'000K				

TUNING

- Run `ctrl-b` then `>` then `v`; then open `htop`
- SLO ~= 1 ms
- Run `./02-data-caching-tune.sh`

Queuing   Throughput   Tail Latency

```
   unix_ts,  timeDiff,      rps,     requests,     gets,     sets,     hits,    misses,  avg_lat,     90th,     95th,     99th,      std,
1679063339,  5.000005, 171983.2,      859917,   687987,   171930,   687987,        0, 0.398698, 0.679100, 0.777100, 1.100000, 0.217527,
Outstanding requests per worker:
16 22 16 9 10 13 5 14
   unix_ts,  timeDiff,      rps,     requests,     gets,     sets,     hits,    misses,  avg_lat,     90th,     95th,     99th,      std,
1679063344,  5.000005, 171970.6,      859854,   687887,   171967,   687887,        0, 0.387853, 0.659100, 0.753000, 0.963100, 0.208296,
Outstanding requests per worker:
7 5 8 16 10 5 16 4
   unix_ts,  timeDiff,      rps,     requests,     gets,     sets,     hits,    misses,  avg_lat,     90th,     95th,     99th,      std,
1679063349,  5.000005, 172275.6,      861379,   689222,   172157,   689222,        0, 0.395049, 0.677000, 0.775100, 0.986100, 0.214206,
Outstanding requests per worker:
12 2 3 2 5 3 10 4
```

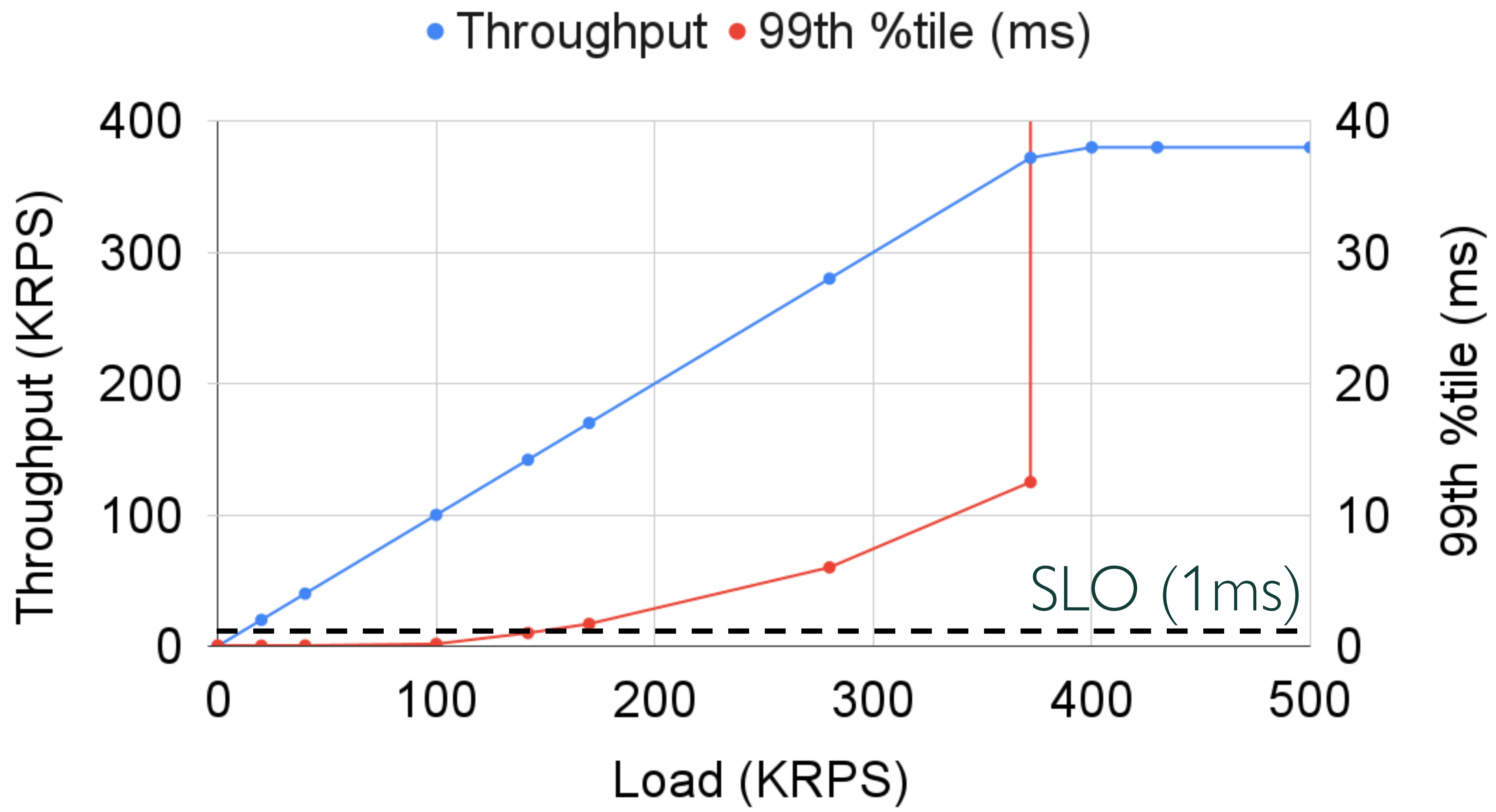
TUNING AWS EC2 RESULTS

- AWS Skylake same socket

Load (RPS)	Throughput (RPS)	99th %tile (ms)	CPU util.	Queueing (req. per worker)
20K	20K	0.04	20%	0
40K	40K	0.05	40%	0
100K	100K	0.17	92%	0
142K	142K	~1	100%	10
170K	170K	1.7	100%	20
372K	372K	13	100%	450
1'000K	380K	infinite	100%	infinite

THROUGHPUT vs. LATENCY RESULTS

PARALLEL SYSTEMS ARCHITECTURE LAB



Online services target finding the maximum throughput under SLO

OUTLINE

PARALLEL SYSTEMS ARCHITECTURE LAB

Part 1: Why CloudSuite?

- Server workloads' benchmarking
- Introducing CloudSuite 4.0

Part 2: Hands-on experience

- CloudSuite on a real machine
  - Tuning the workload
  - Extracting µArch characteristics
- CloudSuite in a full-system emulator (QEMU)
  - Cache hierarchy simulation

µArch CHARACTERISTICS

- Load the machine at maximum throughput under SLO
- Run `./03-data-caching-loadrps.sh`
- Run `Ctrl-b` then `>` then `v` to open a new vertical tmux panel
  - Change panels with `ctrl-b` then arrow keys

```
   unix_ts,  timeDiff,     rps,        requests,     gets,       sets,       hits,       misses,  avg_lat,     90th,      95th,      99th,      std,       min,       max,    avgGetSize
1679737991,   5.000005,  144055.9,      720280,     576470,     143810,     576470,          0,  0.389770,  0.644100,  0.724100,  0.908100,  0.196837,  0.011000,  1.578000, 1081.638873
Outstanding requests per worker:
6 4 8 2 4 8 1 3
   unix_ts,  timeDiff,     rps,        requests,     gets,       sets,       hits,       misses,  avg_lat,     90th,      95th,      99th,      std,       min,       max,    avgGetSize
1679737996,   5.000005,  144171.7,      720859,     577088,     143771,     577088,          0,  0.389451,  0.643100,  0.726100,  0.919100,  0.198711,  0.011000,  2.187000, 1081.577193
Outstanding requests per worker:
10 9 9 6 4 6 5 3
   unix_ts,  timeDiff,     rps,        requests,     gets,       sets,       hits,       misses,  avg_lat,     90th,      95th,      99th,      std,       min,       max,    avgGetSize
1679738001,   5.000005,  144099.7,      720499,     576301,     144198,     576301,          0,  0.393499,  0.653100,  0.735100,  0.928100,  0.201349,  0.012000,  1.745000, 1081.573364
Outstanding requests per worker:
2 6 5 4 5 1 5 6

ubuntu@ip-172-31-39-192:~/part01-realsystem$
```

µArch CHARACTERISTICS (cont.)

PARALLEL SYSTEMS ARCHITECTURE LAB

- Out-of-Order processors are designed to maximize IPC
- IPC: Instructions Per Cycle
- Run `./04-perf-uarch-ipc.sh`

```
[ubuntu@ip-172-31-39-192:~/part01-realsystem$ \
[> cat 04-perf-uarch-ipc.sh
perf stat -C 2 -M IPC -- sleep 5
ubuntu@ip-172-31-39-192:~/part01-realsystem$
```

µArch CHARACTERISTICS (cont.)

PARALLEL SYSTEMS ARCHITECTURE LAB

- Perf shows that Data Caching has low IPC

```
ubuntu@ip-172-31-39-192:~/part01-realsystem$ perf stat -C 2 -M IPC -- sleep 5

 Performance counter stats for 'CPU(s) 2':

       14010328857      inst_retired.any          #     0.83 IPC
       16839319312      cpu_clk_unhalted.thread

       5.000979643 seconds time elapsed
```

Why can the CPU not execute more instructions?

µArch CHARACTERISTICS (cont.)

PARALLEL SYSTEMS ARCHITECTURE LAB

- Cloud workloads have huge datasets
- MPKI = Misses Per Kilo Instructions
- LnMPKI measures only data misses
- Run `./05-perf-uarch-dcache.sh`

```
[ubuntu@ip-172-31-39-192:~/part01-realsystem$ \
[> cat 05-perf-uarch-dcache.sh
perf stat -C 2 -M L1MPKI,L2MPKI -- sleep 10
perf stat -C 2 -M L3MPKI -- sleep 10
ubuntu@ip-172-31-39-192:~/part01-realsystem$
```

µArch CHARACTERISTICS (cont.)

PARALLEL SYSTEMS ARCHITECTURE LAB

- L1-D MPKI is high, 9.2
- ~25% of L1-D misses also miss in the L2
- ~80% of L2 data misses also miss in the LLC

```
ecparsa@ecocloud-exp02:~/asplos_tutorial$ perf stat -C 2 -M L1MPKI,L2MPKI,L3MPKI -- sleep 5

 Performance counter stats for 'CPU(s) 2':

    13,269,444,240      INST_RETIRED.ANY            #     1.95 L3MPKI
        25,858,737      MEM_LOAD_RETIRED.L3_MISS
    13,269,444,132      INST_RETIRED.ANY            #     2.66 L2MPKI
        35,355,458      MEM_LOAD_RETIRED.L2_MISS
    13,269,444,024      INST_RETIRED.ANY            #     9.32 L1MPKI
       123,733,994      MEM_LOAD_RETIRED.L1_MISS

       5.002211089 seconds time elapsed
```

µArch CHARACTERISTICS (cont.)

- Server workloads have large instruction working sets
- Frontend supplies instructions
- L1-I misses are on the critical path
- Run `./06-perf-uarch-icache`

```
ubuntu@ip-172-31-39-192:~/part01-realsystem$ \
> cat 06-perf-uarch-icache.sh
perf stat -C 2 -e frontend_retired.l1i_miss,instructions -- sleep 5
ubuntu@ip-172-31-39-192:~/part01-realsystem$
```

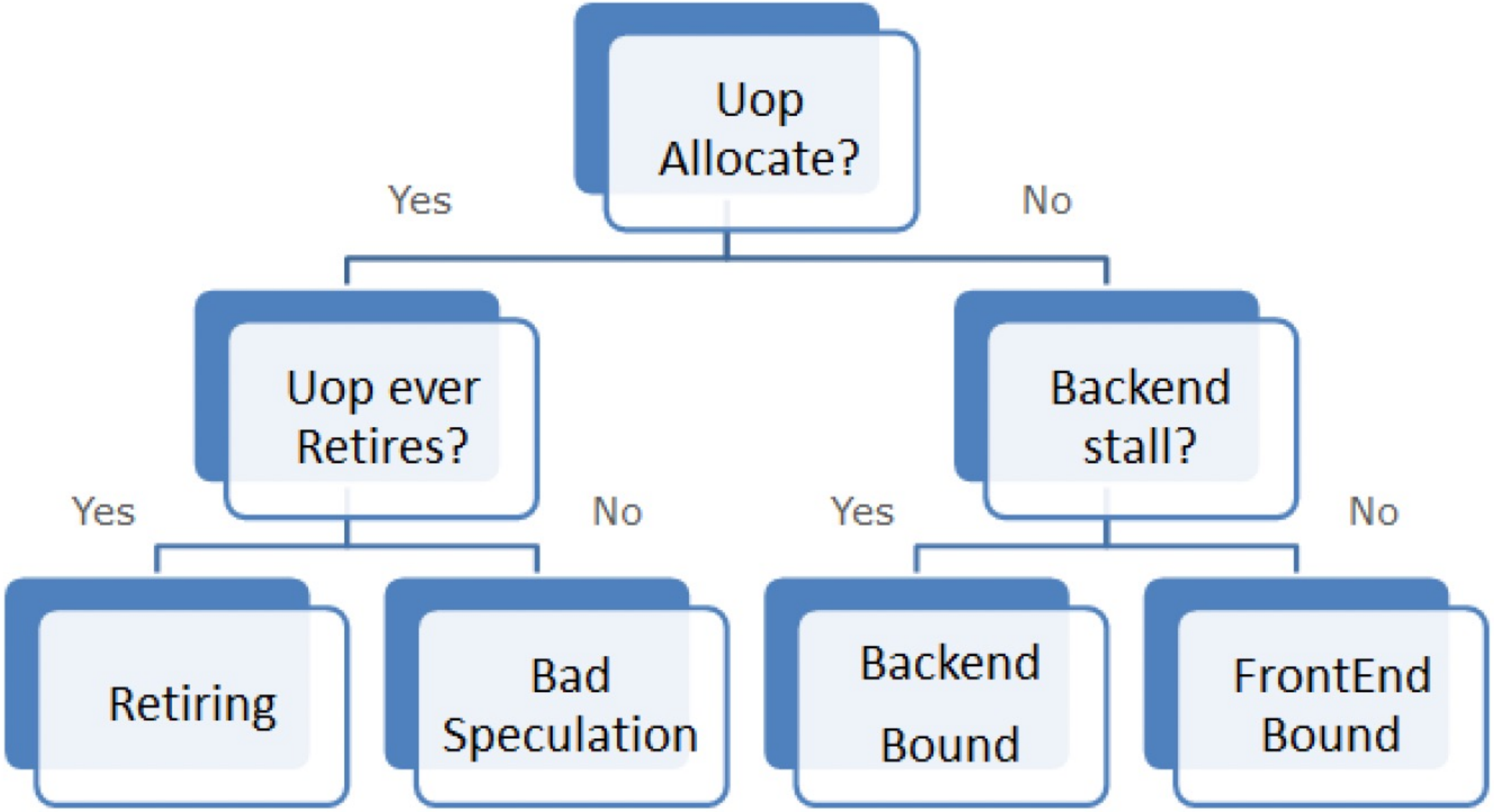
µArch CHARACTERISTICS (cont.)

- L1-I MPKI is around 20
- Back of the envelope calculation
  - (250M/13.3B)*1000 ~= 20 MPKI
  - L2 access latency ~= 10 cycles
  - 20*10 stall cycles = 200 stall cycles PKI
  - 200 stall cycles out of 1000 total cycles
  - 20% performance drop due to L1-I misses

```
ecparsa@ecocloud-exp02:~/asplos_tutorial$ perf stat -C 2 -e frontend_retired.l1i_miss,instructions -- sleep 5

 Performance counter stats for 'CPU(s) 2':

       251,857,093      frontend_retired.l1i_miss
    13,289,567,102      instructions

       5.002147902 seconds time elapsed
```

TOP-DOWN METHODOLOGY [Yasin, ISPASS'14]

PARALLEL SYSTEMS ARCHITECTURE LAB



Top-Down is a powerful technique to identify the bottlenecks

µArch CHARACTERISTICS (cont.)

PARALLEL SYSTEMS ARCHITECTURE LAB

- Top-Down indicates Data Caching is mostly backend bound
- Frontend bound is also noticeable
  - Instruction supply is a bottleneck
- Only 18% of pipeline slots are useful

```
ecparsa@ecocloud-exp02:~/asplos_tutorial$ perf stat -C 2 --topdown -- sleep 5

 Performance counter stats for 'CPU(s) 2':

           retiring      bad speculation       frontend bound        backend bound
              18.0%                 2.7%                20.0%                59.2%

       5.002252470 seconds time elapsed
```

Fix the backend problem to improve performance of Data Caching

OUTLINE

Part 1: Why CloudSuite?

- Server workloads' benchmarking
- Introducing CloudSuite 4.0

Part 2: Hands-on experience

- CloudSuite on a real machine
  - Tuning the workload
  - Extracting μArch characteristics
- CloudSuite in a full-system emulator (QEMU)
  - Cache hierarchy simulation

QEMU

PARALLEL SYSTEMS ARCHITECTURE LAB

- QEMU is a free and open-source full-system emulator
  - Supports various hardware platforms including x86, ARM, and RISC-V

- Running CloudSuite requires full-system support
  - QEMU can emulate network, disk, memory allocation, etc.

- Emulating CloudSuite requires emulating billions of instructions
  - QEMU emulates 400 Millions of Instructions Per Second (MIPS) per core

QEMU

SIMULATION SLOWDOWN

PARALLEL SYSTEMS ARCHITECTURE LAB

- Simulation slowdown per core

Real machine:	~ 2 GIPS	1 s
QEMU:	~ 400 MIPS	5 s
Functional simulation:	~ 1 MIPS	30 min
Simple CPU Model:	~ 100 KIPS	5 h
OoO CPU Model:	~ 10 KIPS	2.3 days

For more details about simulation challenges and techniques, please check [SMARTS, ISCA'03]

QEMU LIMITATIONS

- QEMU does not support multi-node execution
  - Clients and servers must be run on the same QEMU machine

- Time distortion
  - QEMU emulation is 10x slower than the real machine
  - Slowdown affects system behavior like timeouts and tail latencies
  - Solutions like enabling *icount* mitigate the problem

METHODOLOGY

- QEMU 7.0 (2022)
  - AARCH64
  - Ubuntu 22.04 LTS server
  - 4 cores, 8 GB of memory
  - *icount* enabled
- Data Caching
  - Snapshot `running`
  - Dataset ~= 1 GB dataset
  - RPS = 500; 99th %tile ~80 ms
  - Server on core 2
  - Clients on cores 1,3
- Cache Sim plugin
  - L1-D
    - 32 KB, 8-way set associative
  - L1-I
    - 32 KB, 8-way set associative
  - L2 private, inclusive
    - 1 MB, 16-way set associative

QEMU HANDS-ON EXERCISE

PARALLEL SYSTEMS ARCHITECTURE LAB

- Run `03-run-sim-qemu.sh`

```
#!/bin/bash
QEMU_DIR=$PWD/qemu
IMAGE_DIR=$PWD/images

$QEMU_DIR/build/aarch64-softmmu/qemu-system-aarch64 \
        --cpu max -machine virt,gic-version=3 -smp 4 -m 16G \
        -rtc clock=vm \
        -drive file=$IMAGE_DIR/qemu-efi.img,format=raw,if=pflash,readonly=on \
        -drive file=$IMAGE_DIR/varstore.qcow2,format=qcow2,if=pflash,readonly=on \
        -drive file=$IMAGE_DIR/jammy-server-cloudimg-arm64.qcow2,format=qcow2,if=virtio \
        -object rng-random,filename=/dev/urandom,id=rng0                       \
        -device virtio-rng-pci,rng=rng0                                        \
        -device virtio-net,netdev=net0                                         \
        -netdev user,id=net0,hostfwd=tcp::8022-:22                             \
        -icount shift=0,sleep=on,align=off                                     \
        -D ./cache-sim-log -d plugin
        -plugin $QEMU_DIR/contrib/plugins/libcache-inclusive.so                \
        -loadvm running                                                        \
        -nographic
```

icount for time distortion

plugin for cache sim.

snapshot with running bench.

CACHE SIMULATION RESULTS

PARALLEL SYSTEMS ARCHITECTURE LAB

- L1-I MPKI is very high
  - No prefetchers
- L1-D and L2 data MPKI corroborates previous results
  - Dataset is only 1 GB

core #	Instructions	L1-D MPKI	L2 dMPKI	L1-I MPKI	L2 iMPKI	L2 MPKI
0	0	-nan	-nan	-nan	-nan	-nan
kernel	0	-nan	-nan	-nan	-nan	-nan
user	0	-nan	-nan	-nan	-nan	-nan
1	0	-nan	-nan	-nan	-nan	-nan
kernel	0	-nan	-nan	-nan	-nan	-nan
user	0	-nan	-nan	-nan	-nan	-nan
2	10000000	12.5	4.0	51.5	1.9	5.9
kernel	7076409	14.4	4.5	64.1	1.9	6.4
user	2923591	7.9	3.0	20.9	1.8	4.7
3	0	-nan	-nan	-nan	-nan	-nan
kernel	0	-nan	-nan	-nan	-nan	-nan
user	0	-nan	-nan	-nan	-nan	-nan
sum	10000000	12.5	4.0	51.5	1.9	5.9
kernel	7076409	14.4	4.5	64.1	1.9	6.4
user	2923591	7.9	3.0	20.9	1.8	4.7

FUTURE WORK

- New benchmarks
  - Database workloads
  - Django and Node.js web applications
- Better support for RISC-V
  - Now, only 4 workloads run on RISC-V
- Updating QFlex
  - Full-system simulation by integrating CloudSuite, QEMU, and QFlex

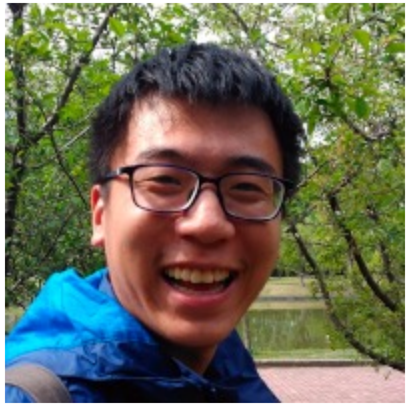
RISC-V QFlex node JS django

CloudSuite TEAM

PARALLEL SYSTEMS ARCHITECTURE LAB

Ali

Shanqing



Rafael

Ayan

Bugra

Babak

Mike

Thank You!

CloudSuite

For more information, please visit us at

cloudsuite.ch

PARALLEL SYSTEMS ARCHITECTURE LAB

EcoCloud

Stony Brook University

EPFL

Backup Slides

SUMMARY OF CHANGES

	CloudSuite 3.0	CloudSuite 4.0
Platforms	x86	x86, ARM, (partial support for RISC-V)
OS	Debian buster	Ubuntu 22.04
Data Analytics	Hadoop 2.7.4 (2017)	Hadoop 2.10.2 (2022)
Graph Analytics	Spark 2.1.0 (2016) Scala 2.10.4 (2017)	Spark 3.3.2 (2023) Scala 2.13.10 (2022)
In-memory Analytics	Spark 2.1.0 (2016) Scala 2.10.4 (2017)	Spark 3.3.2 (2023) Scala 2.13.10 (2022)

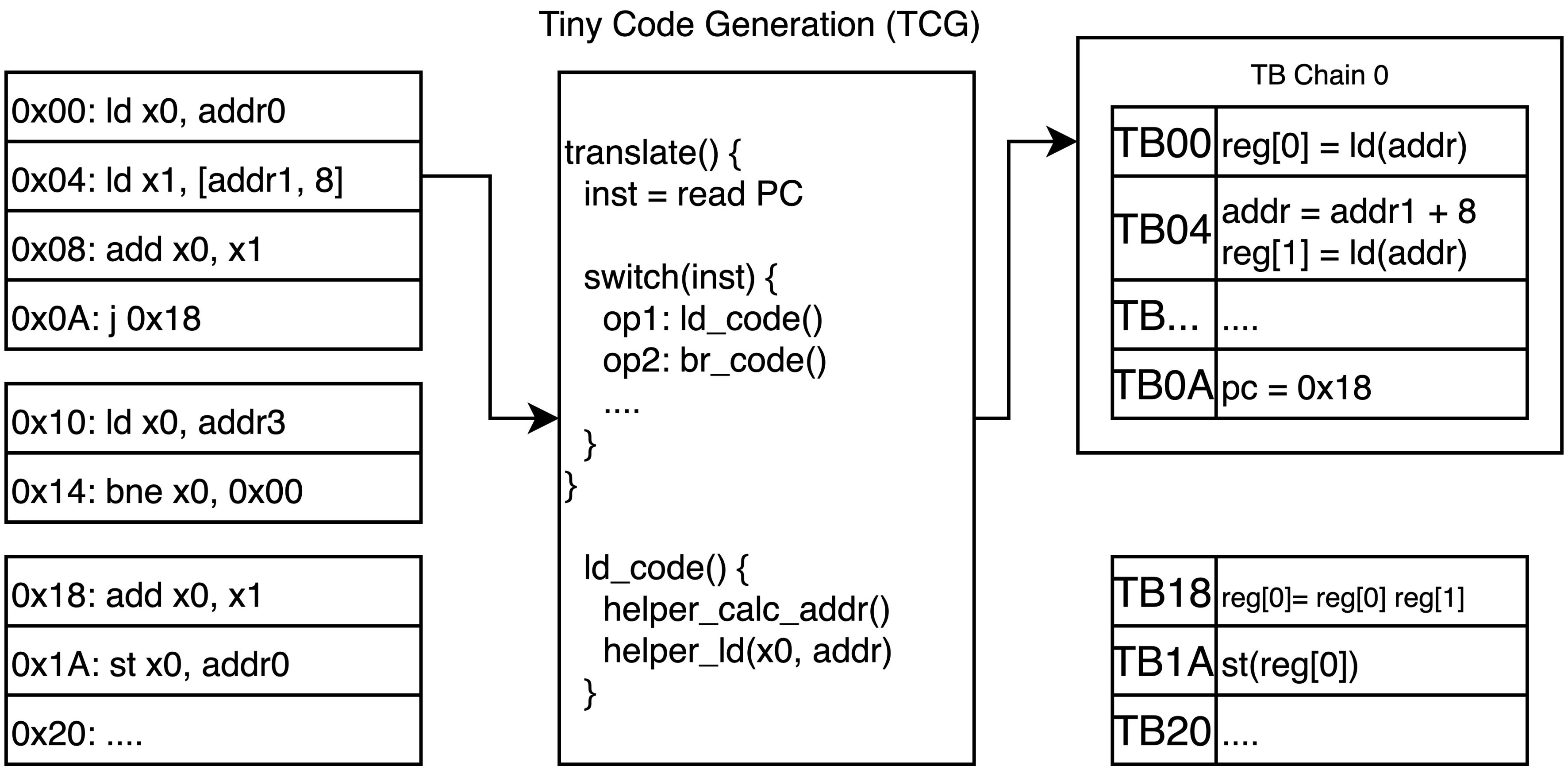
SUMMARY OF CHANGES (cont.)

PARALLEL SYSTEMS ARCHITECTURE LAB

	CloudSuite 3.0	CloudSuite 4.0
Data Caching	Memcached 1.4.24 (2015)	Memcached 1.6.15 (2022)
Data Serving	Cassandra 2.1.12 (2015) YCSB 0.3.0 (2015)	Cassandra 4.1 (2022) YCSB 0.14.0 (2018)
Media Streaming	raw plain text files no encryption	real videos TLSv1.3 encryption
Web Search	Solr 5.2.1 (2015)	Solr 9.1 (2022)
Web Serving	Elgg 1.9.3 (2014) PHP 5 (2016) MySQL 5.5.62 (2018) Memcached 1.4.14 (2012)	Elgg 4.3 (2022) PHP 8.1 (2022) MariaDB 10.6 (2021) Memcached 1.6.15 (2022)

QEMU EMULATION ENGINE

- Translation phase from guest code to Translation Blocks (TB)
- Execution phase of the Translation Blocks



QEMU INSTRUMENTATION

PARALLEL SYSTEMS ARCHITECTURE LAB

- Plugin system since QEMU 5.0 release (2020)
  - Allows for easy hooks to key phases of QEMU execution
  - Inserts callbacks on instruction and data access

```
static void vcpu_tb_trans(qemu_plugin_id_t id, struct qemu_plugin_tb *tb)
{
    size_t n_insns; size_t i; InsnData *data;
    n_insns = qemu_plugin_tb_n_insns(tb);
    for (i = 0; i < n_insns; i++) {
        struct qemu_plugin_insn *insn = qemu_plugin_tb_get_insn(tb, i);

        qemu_plugin_register_vcpu_mem_cb(insn, vcpu_mem_access,
                                         QEMU_PLUGIN_CB_NO_REGS,
                                         rw, data);

        qemu_plugin_register_vcpu_insn_exec_cb(insn, vcpu_insn_exec,
                                               QEMU_PLUGIN_CB_NO_REGS, data);
    }
}
```

QEMU EXECUTION ENGINE

- Insert callbacks for instruction execution
- Insert callback for memory accesses