

Ali Ansari<sup>†</sup>, Shanqing Lin<sup>†</sup>, Miguel Peón-Quirós<sup>‡</sup>, Arash Pourhabibi<sup>†</sup>, Mark Sutherland<sup>†</sup>,

Babak Falsafi<sup>†‡</sup>, Michael Ferdman<sup>‡</sup>

<sup>†</sup>PARSA, EPFL

<sup>‡</sup> EcoCloud, EPFL

<sup>‡</sup>Stony Brook University

## Cloud Server Efficiency



Diagram illustrating cloud server efficiency with logos for Amazon.com, YouTube, Microsoft Bing, and Twitter. Below the logos, a visual equation shows a server rack plus a building plus power lines equals a money bag, symbolizing the cost of hardware, space, and power.

- Constant demand for more servers
- Increasing costs of HW, space & power

## Modern Servers are Scale-Up

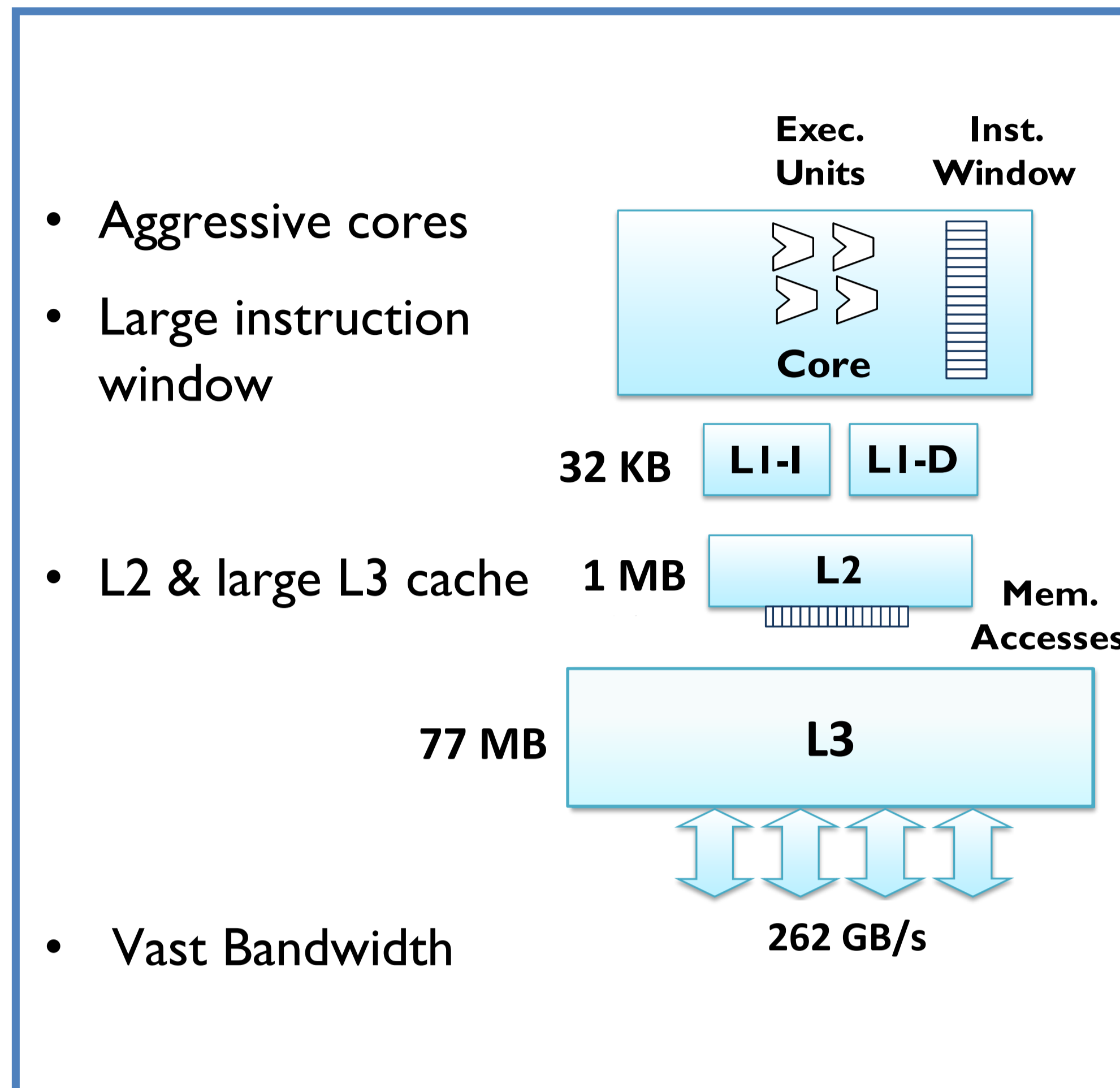


Diagram illustrating modern server architecture with a core containing execution units and an instruction window. It shows a 32 KB L1-I and L1-D cache, a 1 MB L2 cache, and a 77 MB L3 cache. The diagram also indicates a vast bandwidth of 262 GB/s.

- Aggressive cores
- Large instruction window
- L2 & large L3 cache
- Vast Bandwidth

## Cloud Applications are Scale-out

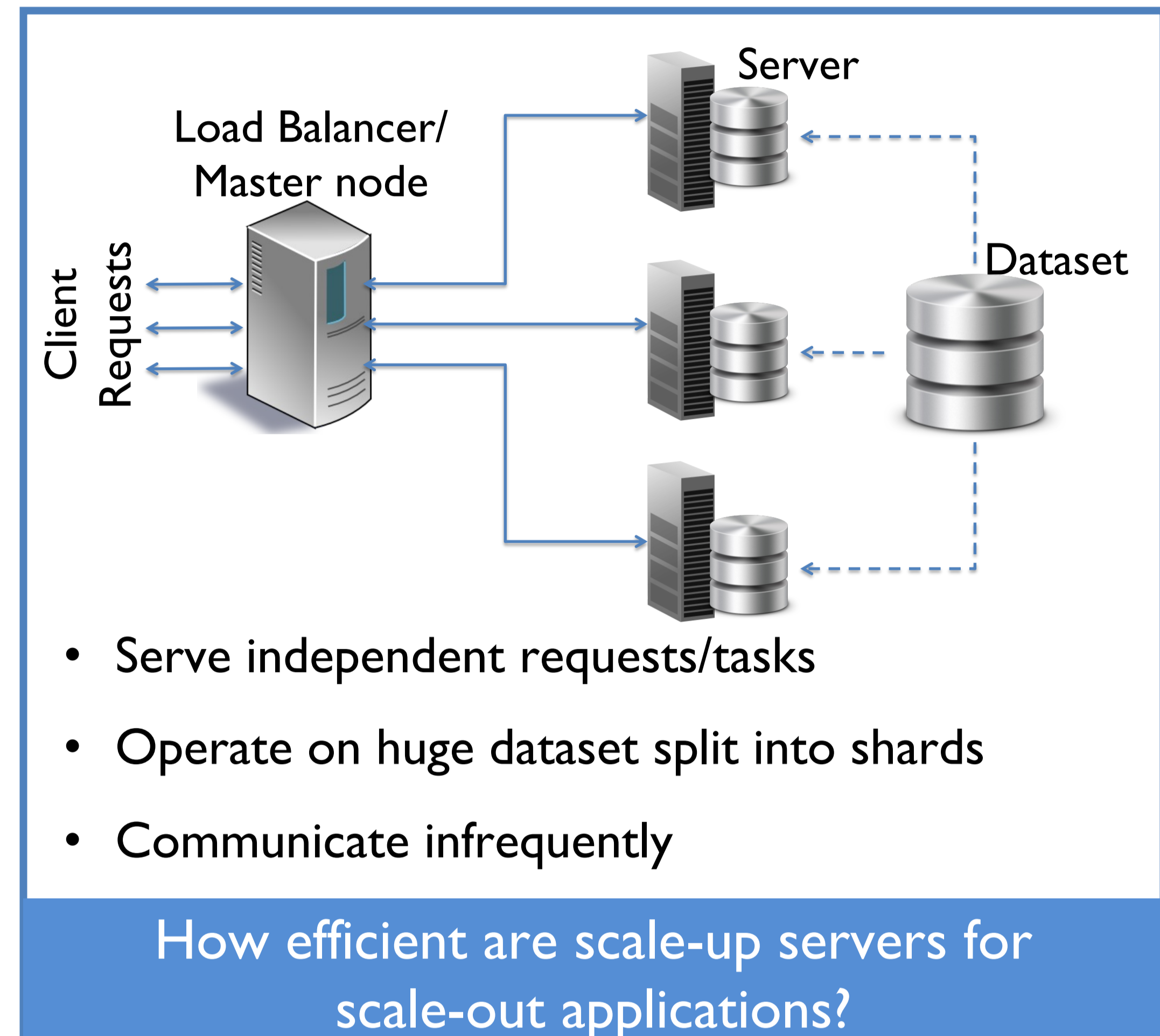


Diagram illustrating cloud application architecture showing a load balancer/master node distributing client requests to multiple servers, which access a shared dataset. The servers are shown as independent units that can serve requests and operate on a huge dataset split into shards.

- Serve independent requests/tasks
- Operate on huge dataset split into shards
- Communicate infrequently

How efficient are scale-up servers for scale-out applications?

## Why not Conventional Scale-Up Processors?

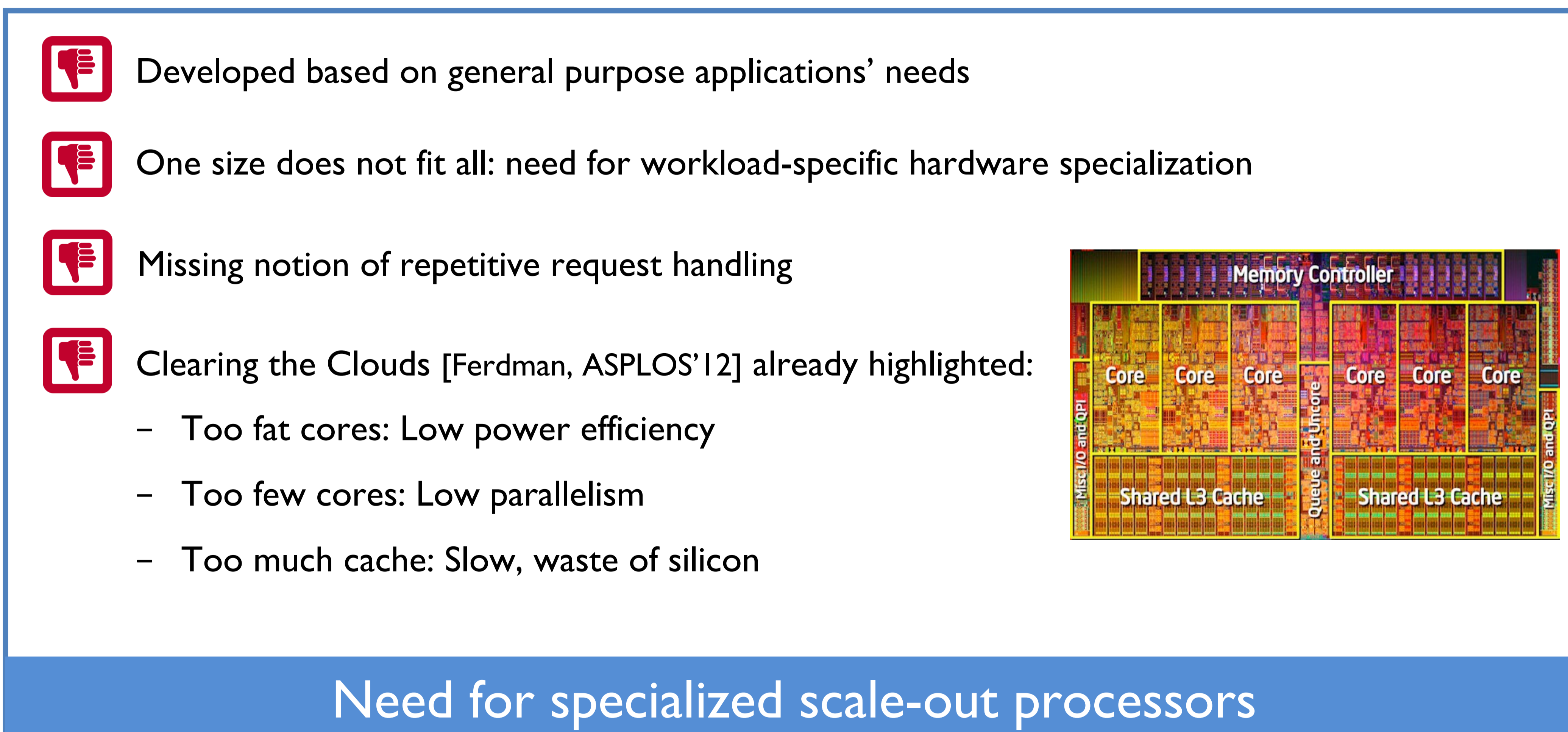


Diagram illustrating why conventional scale-up processors are not ideal for scale-out workloads. It lists several drawbacks: developed based on general purpose applications' needs, one size does not fit all, missing notion of repetitive request handling, and clearing the clouds already highlighted. A micrograph of a processor die is shown to illustrate the physical layout.

- Developed based on general purpose applications' needs
- One size does not fit all: need for workload-specific hardware specialization
- Missing notion of repetitive request handling
- Clearing the Clouds [Ferdman, ASPLOS'12] already highlighted:
  - Too fat cores: Low power efficiency
  - Too few cores: Low parallelism
  - Too much cache: Slow, waste of silicon

Need for specialized scale-out processors

## Processors for Scale-Out Workloads

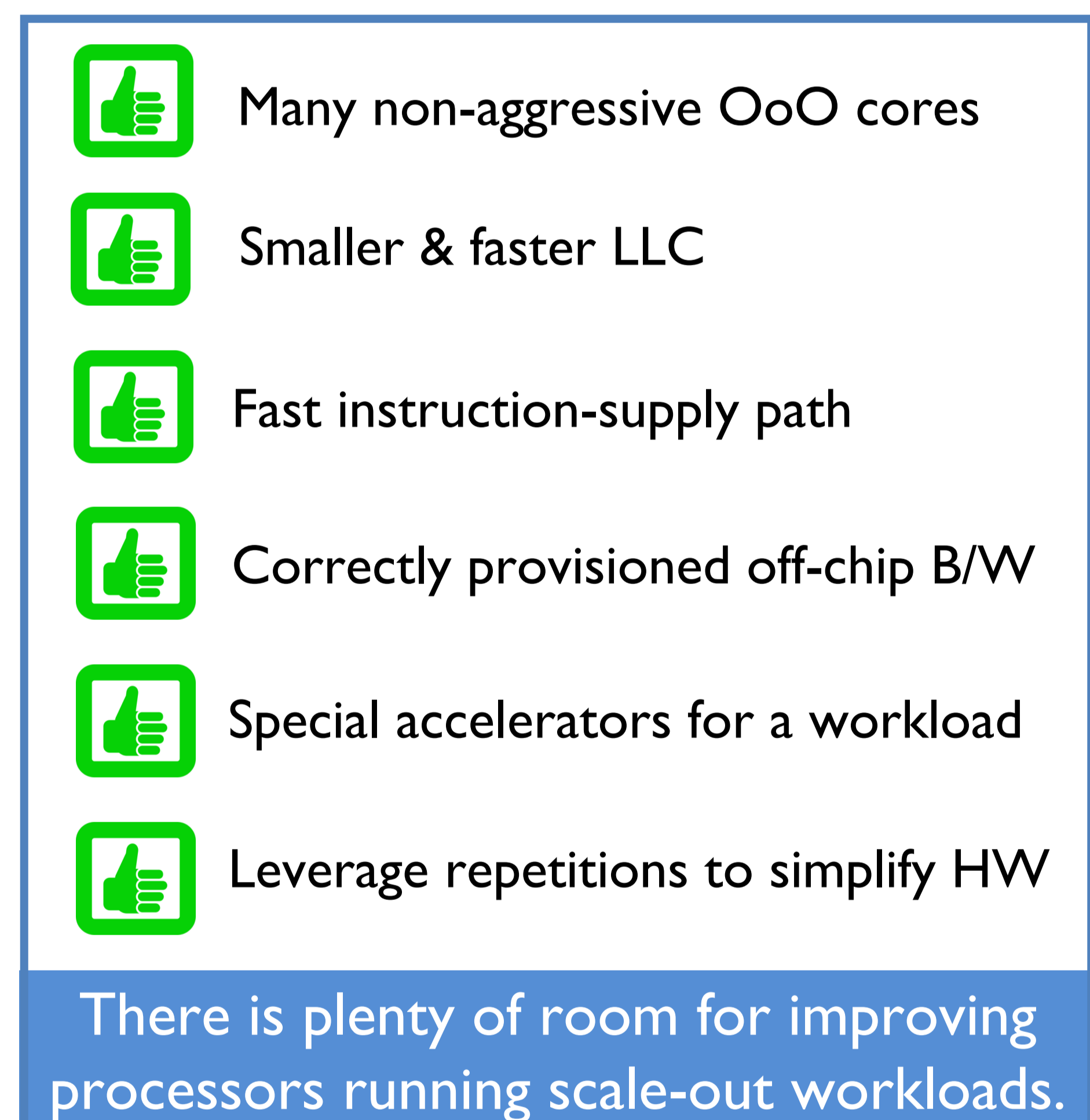


Diagram illustrating the requirements for processors for scale-out workloads. It lists several key features: many non-aggressive OoO cores, smaller & faster LLC, fast instruction-supply path, correctly provisioned off-chip B/W, special accelerators for a workload, and leverage repetitions to simplify HW. A concluding statement notes that there is plenty of room for improving processors running scale-out workloads.

- Many non-aggressive OoO cores
- Smaller & faster LLC
- Fast instruction-supply path
- Correctly provisioned off-chip B/W
- Special accelerators for a workload
- Leverage repetitions to simplify HW

There is plenty of room for improving processors running scale-out workloads.

## CloudSuite 4.0

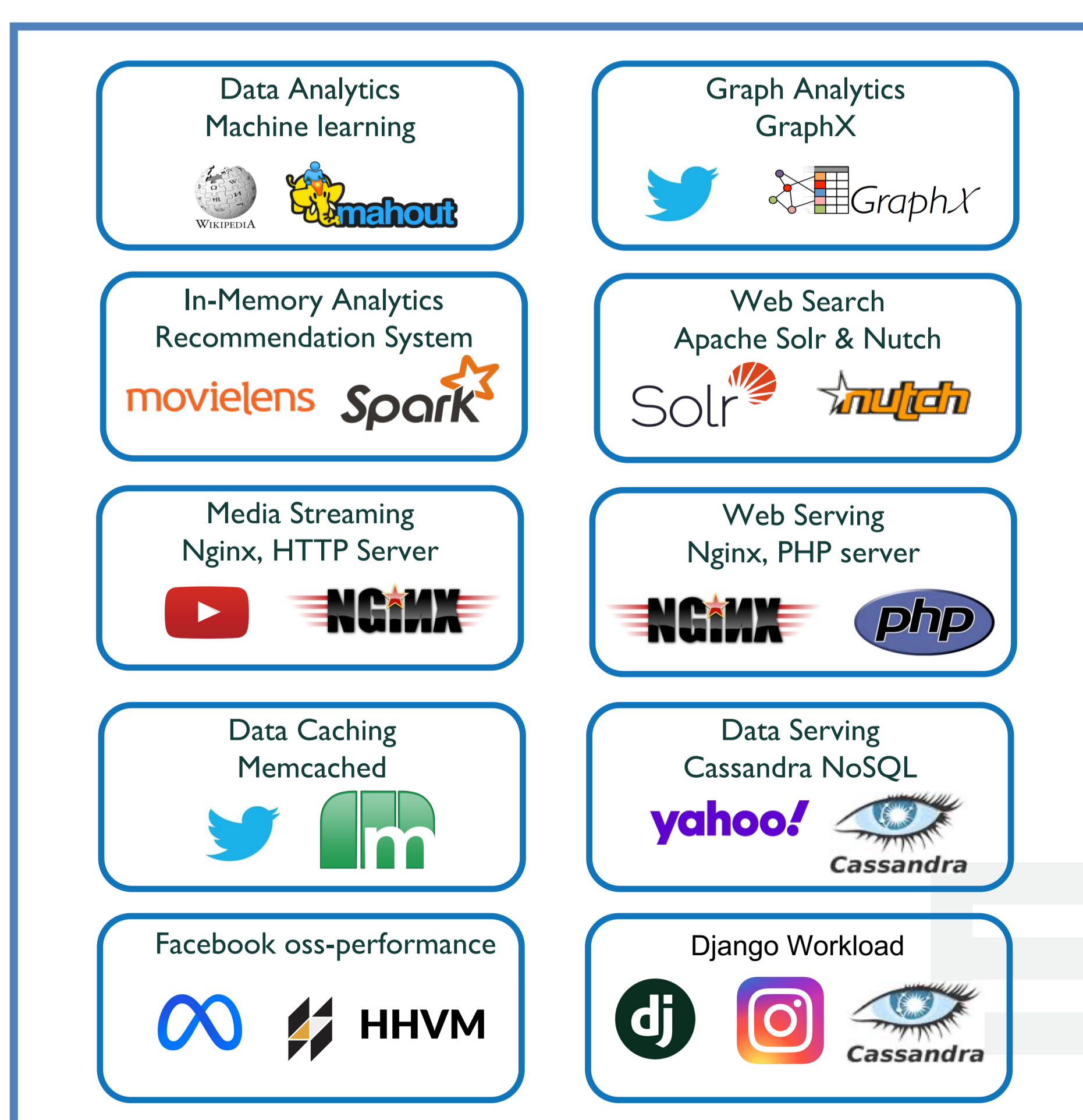


Diagram illustrating the various workloads supported by CloudSuite 4.0, categorized into Data Analytics, Graph Analytics, In-Memory Analytics, Web Search, Media Streaming, Web Serving, Data Caching, Data Serving, Facebook oss-performance, and Django Workload.

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

## What is New in CloudSuite 4.0?

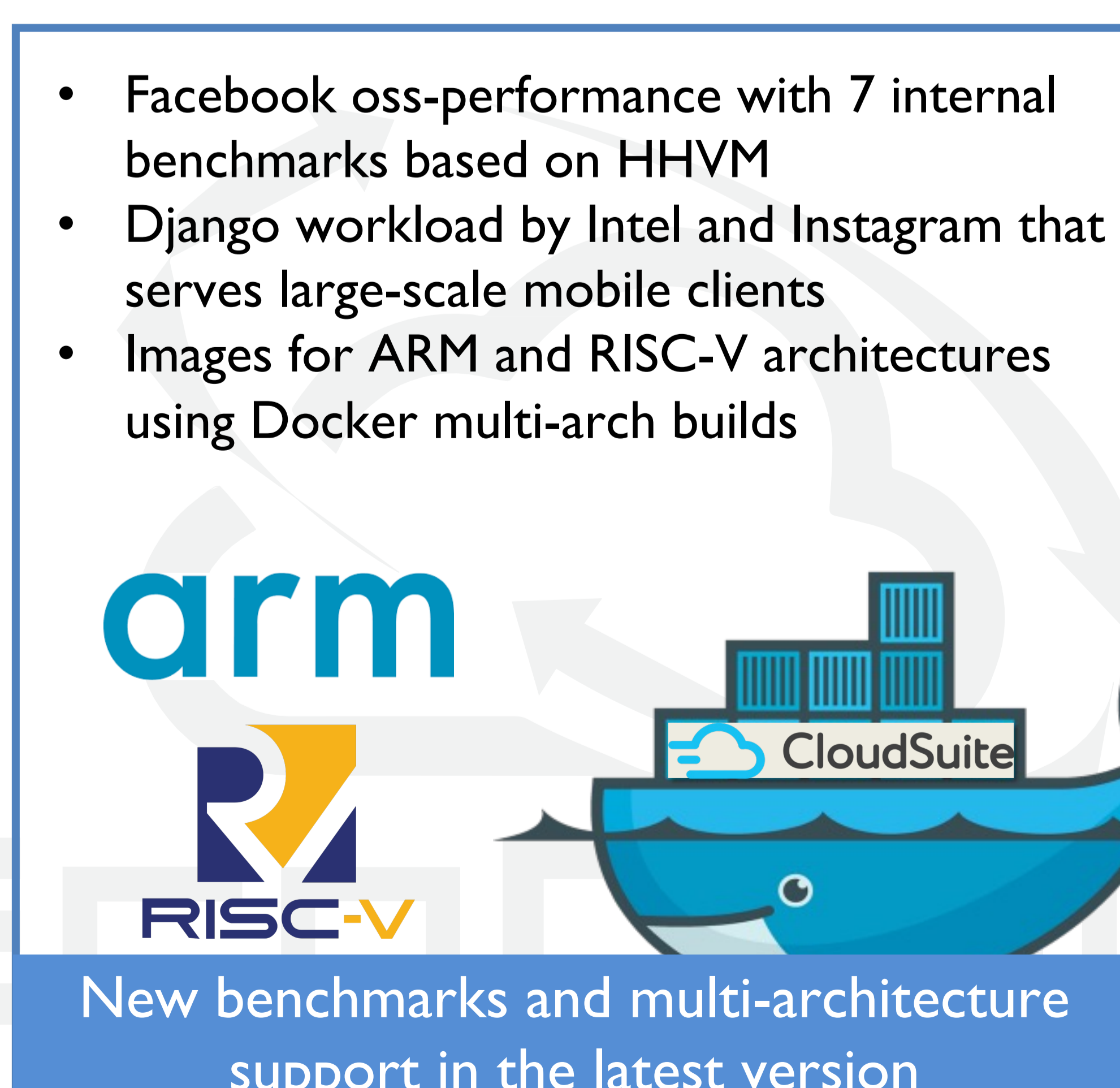


Diagram illustrating the new benchmarks and multi-architecture support in the latest version of CloudSuite 4.0. It highlights the addition of Facebook oss-performance, Django workload, and support for ARM and RISC-V architectures using Docker multi-arch builds.

- Facebook oss-performance with 7 internal benchmarks based on HHVM
- Django workload by Intel and Instagram that serves large-scale mobile clients
- Images for ARM and RISC-V architectures using Docker multi-arch builds

New benchmarks and multi-architecture support in the latest version

## Research Directions



Diagram illustrating the research directions for scale-out server workloads. It lists several key areas of research: identifying mismatches between workloads' characteristics and processors' implementation, deployment of ARM and RISC-V as emerging server architectures, power and energy consumption characteristics of scale-out server workloads, and industry's response to scale-out workloads' requirements over the past decade.

- Identifying mismatches between workloads' characteristics and processors' implementation to propose workload-specific processor design
- Deployment of ARM and RISC-V as emerging server architectures
- Power and energy consumption characteristics of scale-out server workloads
- Industry's response to scale-out workloads' requirements over the past decade

Interesting opportunities for research on scale-out server workloads