Open Source 西门子

Oct. 29
Industrial Open Source

Oct. 30
Open Source Robotics

Agenda

Beijing · 2026

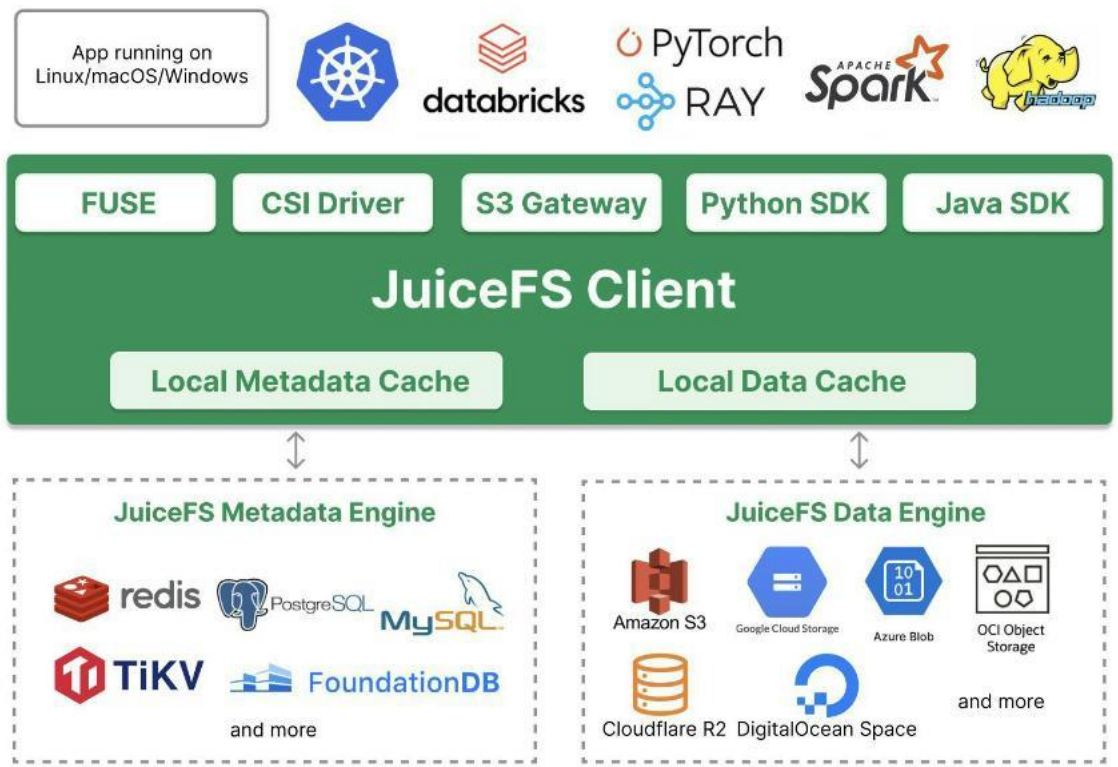
Open Storage for Autonomous Driving and Physical AI

Production Lessons from JuiceFS

Rui Su, co-founder of JuiceFS

Where does a training sample spend its time?

JuiceFS at a glance



2017
Founded

2021
Open source

Built for the cloud

Multi-cloud and hybrid deployments

JuiceFS scale

500B files

EB scale

1 million clients

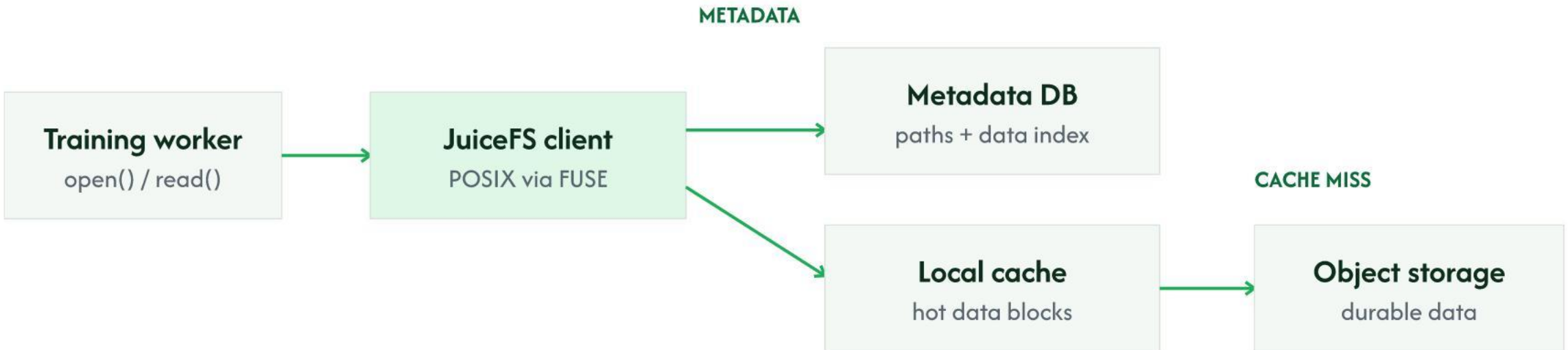
Pragmatic design

Performance balanced with cost

Autonomous Driving / Physical AI / LLM & Multi-Modal Models / Biotech / Quant Investment

Community Edition architecture: Open Source Summit Japan 2025, p.13

The GPU is waiting. Which part needs capacity?



Path lookup **Block location** **Cache / origin** **Decode**

A cache hit can still leave metadata calls and CPU work on the critical path.

Bytes, files and reuse grow independently

Road logs
continuous writes

→

Frames + point clouds
many file creates

→

Labels + derivatives
more objects

→

Train / simulate
repeated reads

48,000

illustrative file count

8 cameras × 10 frames/s × 600 s

image files from 10 minutes, if each frame is a file

Capacity

bytes retained

Metadata

files × operations

Read demand

epochs × consumers

Source talk, pp. 5–6, 15–17

A sample can be a time window inside a shard

dataset[i] with relative timestamps



Episode metadata

shard path + start/end offsets

Parquet shards

state / action / timestamps

MP4 shards per camera

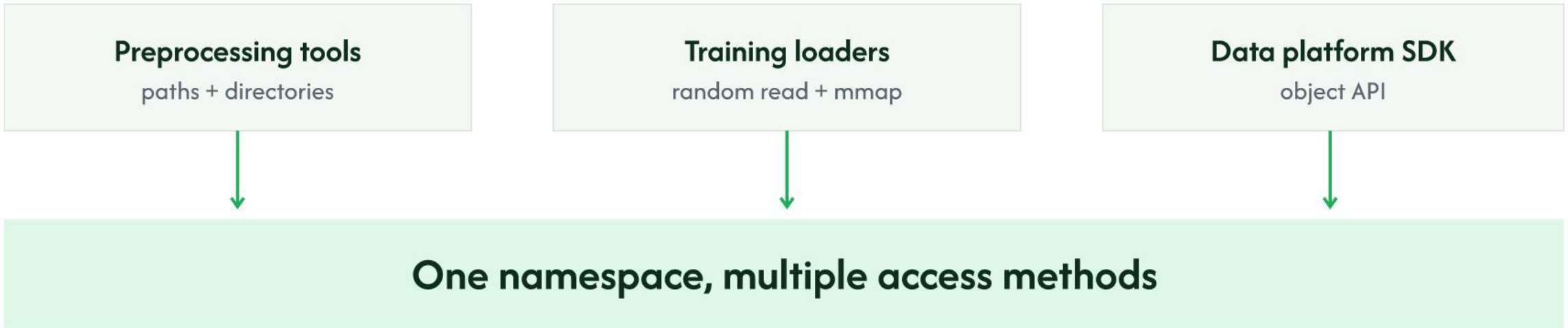
many episodes per file

Fewer filesystem entries

More work in indexing and decode

LeRobot v3 packs multiple episodes into each Parquet / MP4 file.

File semantics keep existing tools working



POSIX reduces integration work

Access latency still depends on file count, cold data and client concurrency.

One read crosses two independent services

open("/train/scene/frame.jpg")

read(offset, length)



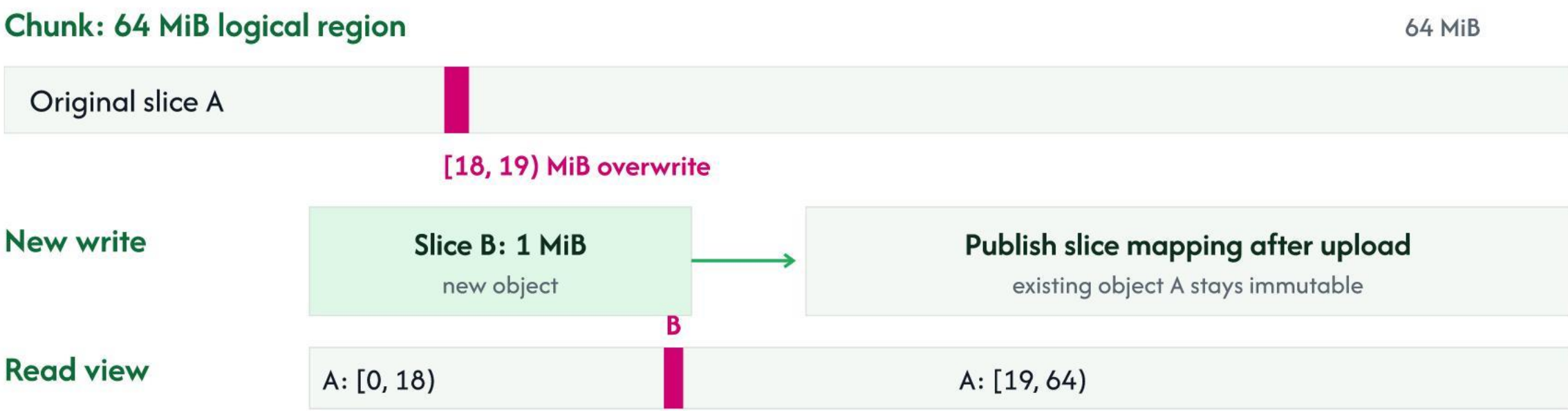
Metadata database

Data path bypasses the metadata DB

Rename / create require atomic namespace updates.

Client and kernel caches can skip work. This is the conceptual cold path.

A partial overwrite creates a new slice

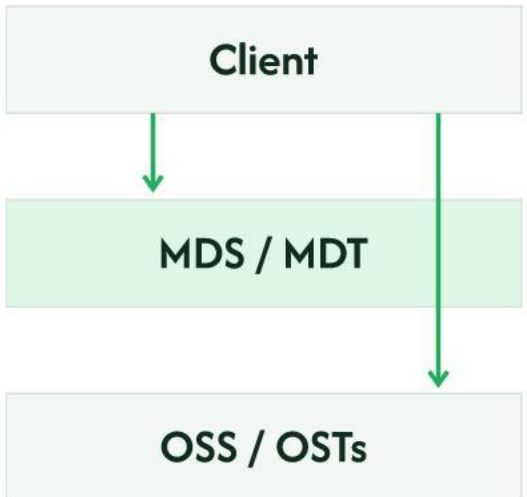


Blocks default to 4 MiB. A short slice can produce a smaller object.

Repeated overwrites add fragments. Compaction and GC consume background I/O.

Four designs place responsibility in different layers

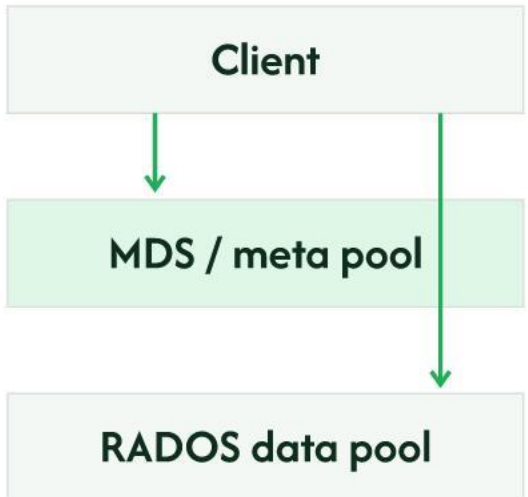
Lustre



Stripe layout

targets + HA

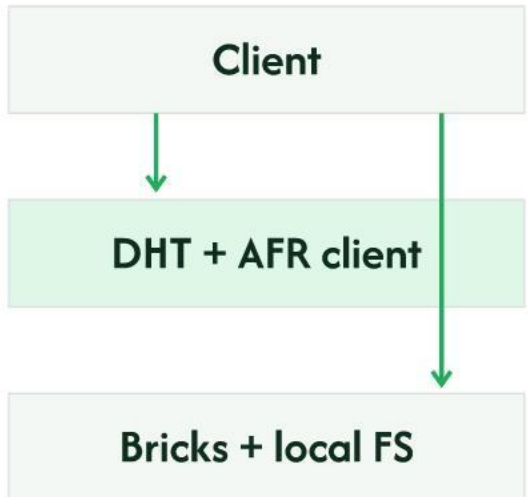
CephFS



MDS + RADOS

placement + recovery

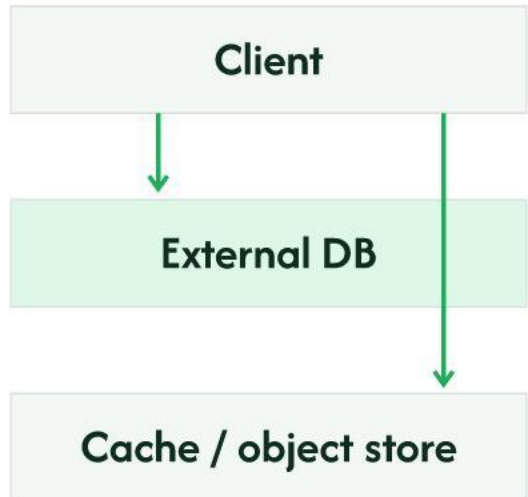
GlusterFS



Client distribution

rebalance + heal

JuiceFS CE



DB + origin + cache

cold read + DB limits

Zelos copies data before training across clouds

700 TB

one reported bucket

600 million files

Self-built data center
JuiceFS + Redis + MinIO

tool built on juicefs sync

dataset copy over dedicated links

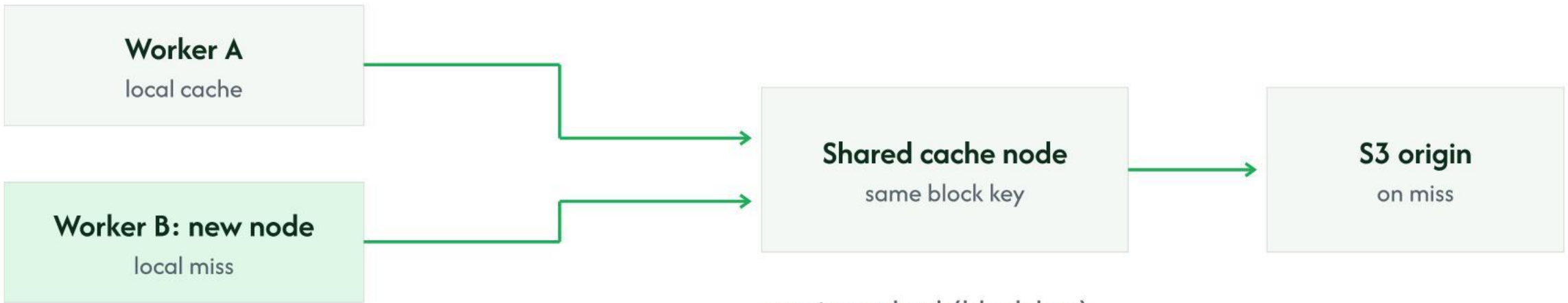
Public cloud
JuiceFS + Redis + OSS

Kubernetes CSI + PVC

Node-local NVMe cache

Enterprise Multi-Region is a separate capability from the sync tool in this case.

A new worker can reuse another node's hot data



Local cache

reuse within a node

Shared cache

reuse across nodes

Zhuoyu's distributed cache is a customer extension of Community Edition.

Data preparation becomes a task dependency

≈80%

used within 8 days of upload

Upload ⟶ **Build cache**

TTL follows the observed use window

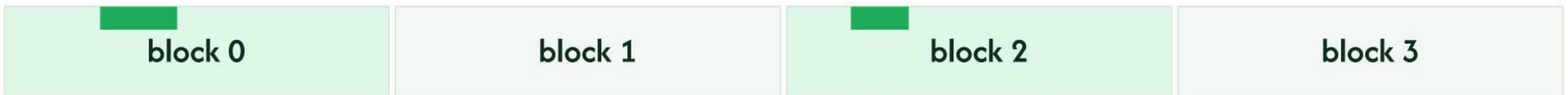


Fluid coordinates preload and lifecycle. Other jobs use Zhuoyu's cache API.

Column selection reduces the bytes to prepare



Illustrative 16 MiB file



Two 4 MiB blocks warmed for selected ranges

Reported I/O time: −20–90% across pipelines

Range warmup · PR #7398

Lance parser · PR #7399

Near saturation, a small gain raises the tail

1M sequential reads, aggregate server links: 1,200 Gbit/s

≈135 GiB/s	≈138 GiB/s
P99 ≈ 2 ms	P99 ≈ 20–30 ms

≈2.2% more throughput

10–15× P99

Measured cache path. A cold object-store read or training job has different limits.

Distributed metadata still has hot access paths

EB scale **10^{11} files** **10^{5}-scale clients**

Observed bottleneck		Customer change
Leader CPU full, followers spare	→	Follower read routing
Timestamp request to PD / TSO	→	One less RTT for frequent point reads
Repeated config / capacity queries	→	Cache hot metadata

Horizon's Community Edition engineering. Application consistency still needs protection.

The next measurement should locate the wait

Path / inode → Slice index → Cache / origin → Load / decode

01 Metadata or bytes?

Trace lookup / open, read and decode separately.

02 Cold or warm?

Compare empty-cache and prepared reads.

03 Same node or more workers?

Move tasks. Scale workers. Inspect P99 and misses.

Keep in touch

JuiceFS WeChat

Rui Su · WeChat

rsu@juicedata.io

x.com/suavesu

linkedin.com/in/suave/

Open Source 西门子

Oct. 29
Industrial Open Source

Oct. 30
Open Source Robotics

Agenda

Beijing · 2026