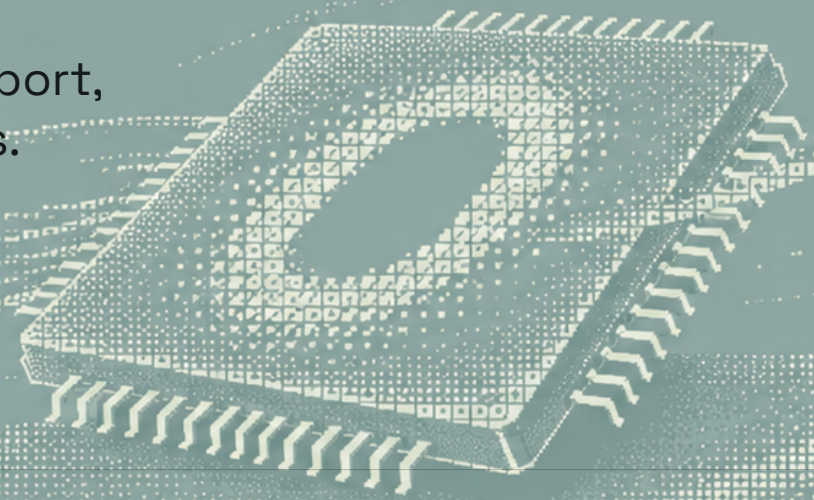
zerolat

Low-level AI optimization
Measured business value

Hardware-specific kernels and deployment support,
built around your workload and operating costs.



OPTIMIZE DEPLOY MAINTAIN MEASURE

Inference cost sits across the stack

A model's speed depends on its workload, memory movement, runtime and target hardware. The business pays for the complete deployment.

WORKLOAD

Shapes and traffic

Input sizes, batch size, concurrency and latency targets.

EXECUTION

Kernel execution

Compute, data transfers, precision and scheduling.

OPERATIONS

Releases and cost

Rollouts, regressions, monitoring and recurring spend.

One workload. One hardware target

CUSTOMER INPUTS

Deployment profile

Representative inputs and traffic
Hardware and software versions
Quality and reliability constraints
Latency, throughput and cost targets

ZEROLAT DELIVERABLES

A tested execution path

Tuned kernels and runtime settings
Reproducible before-and-after report
Deployment package and rollback plan
Maintenance scope and cost model

Optimization starts with a measured bottleneck

Change the parts that matter, then verify the whole path under the same workload and quality constraints.

01

Profile

Find time spent in compute, transfer and waiting.

02

Tune

Adjust tiling, fusion, memory use and scheduling.

03

Validate

Check outputs, tolerances and failure cases.

04

Compare

Measure latency, throughput and cost per inference.

Performance has to survive the next release

The engagement continues across model updates, runtime changes and production traffic.

DEPLOYMENT

Controlled production rollout

Versioned build, staged release, monitoring and a tested rollback.

ONGOING MAINTENANCE

Update and revalidate

Compatibility updates, regression checks and periodic cost/performance reviews.

Project work, followed by recurring support

01	Paid assessment	Measure the baseline. Agree the target workload, constraints and acceptance criteria.
02	Optimization project	Deliver engineering, validation and production rollout against the agreed scope.
03	Maintenance plan	Support updates, deployments and ongoing cost analysis through a recurring agreement.

The decision spans build cost and run cost

ONE-TIME INVESTMENT

Engineering + rollout

Profiling, kernel development, testing, integration and production launch.

ONGOING COST

Compute and ops

Inference usage, hosting, monitoring, support and maintenance.

BUSINESS RETURN

Revenue + margin

Track revenue, modeled surplus, payback and cumulative benefit.

Lifecycle cost = one-time investment + recurring cost over the chosen period.

A transparent example of the business case

Assume the same traffic and €40,000 monthly revenue. All values below are illustrative.

MONTHLY CUSTOMER VIEW	BASELINE	OPTIMIZED
Inference compute	€20,000	€12,000
Other operating costs	€5,000	€5,000
zerolat maintenance	€0	€2,000
Total recurring cost	€25,000	€19,000
Monthly surplus*	€15,000	€21,000

€6,000

net monthly savings

8.3 months

simple payback on €50k

*Before other overhead and tax. €50k incremental upfront: €40k engineering + €10k rollout. Payback starts at full savings after rollout. Quality and reliability must be validated.

A Tenstorrent workload is the first testbed

The supplied YOLOv4 / Wormhole N150 code gives us a reproducible starting point for a focused optimization pilot.

AVAILABLE FOUNDATION

Profile, trace and evaluate

The shared reference includes stage timing, trace replay, command queues and an accuracy harness.

PILOT ACCEPTANCE

One comparable result

Match workload and quality criteria. Report latency, throughput, cost and versioned artifacts.

This is a proposed zerolat pilot. No zerolat accelerator improvement or customer saving has been measured in this deck.

A focused workload
A measurable decision

Begin with a baseline and an agreed constraint.
Continue when the measured value supports it.

Hardware-specific engineering
Production deployment and maintenance
Lifecycle cost and contribution analysis

