INFERENCE ENGINEERING REPORT

Catnip Portrait on Dual RTX 5090 Current K32 MXFP8 Engi-

模型 LTX-2.3 基座与 Catnip Streaming Model

硬件 2 × NVIDIA GeForce RTX 5090

当前合同 480×832，20 秒，489 帧，first 4 + latest 6 LF KV

报告日期 2026-07-28

状态 Portrait v2 engineering baseline ACTIVE；仅声明 warm-throughput，formal video quality acceptance 未完成

执行摘要

这项工作的核心不是“把所有算子都改成 FP8”，而是在固定 Streaming 语义、20 秒视频合同和 fail-closed 门禁下，逐步解决容量、显存生命周期、双卡调度、W8A8 GEMM、编译稳定性、provider 选择和精度泛化问题。报告不按模块堆砌结论，而按真实实验顺序叙述：每一步先说明动作，再给 profiling 或实验结果，随后解释判断，最后记录为什么进入下一步。

本版在 2026-07-21 首版之上追加了八批工作：576 层的 outlier 统计研究、自研 SM120 MXFP8 attention kernel、验收判据的证伪与重建、K32 MXFP8 工程基线 V2、该基线的设备分离 profiling、V-direct-BSH attention boundary A/B、Portrait v1，以及 2026-07-28 的 Portrait v2 基线晋升。VAE 优化已有独立报告，本报告仅记录其集成结果、当前 KPI 与证据边界，不重复实验过程。

2026-07-28 current state：应优先引用的结论

- **当前 engineering baseline** 是 catnip-k32-vdirect-single-tile-compile-portrait-480x832-v2：480×832、20 s、489 frames、11 chunks、55 forwards、first 4 + latest 6 LF KV，运行在 2× RTX 5090。
- 三对 Phase 3 candidate runs 的权威 warm 均值为 **37.557043 stable-DiT FPS** 与 **34.392314 generated FPS**；generated-FPS population CV 为 0.085863%。25 FPS 是播放速率，不是生成吞吐。
- GPU1 执行 complete Catnip DiT；GPU0 执行 INT8 text encoder 与 BF16 video/audio VAE。576 selected Linear 使用 native K32 MXFP8 scaled-MM，48 video-self attn1 使用 V-direct-BSH；v2 接纳 single-tile latent-window decode 和 video VAE decoder compile。
- 最终无争用 promotion smoke r03 以 34.408774 generated / 37.566968 stable-DiT FPS 通过；489-frame H.264 视频、AAC 音频、4+6 KV、576 K32、48 V-direct 与 no-new-graph gates 全部成立。
- 本次只声明 warm steady-state throughput。fresh-process wall paired median 增加 10.802722%，不声明 cold-start speedup；formal ground-truth quality acceptance 仍未建立。

历史工作链路中仍成立的结论

- 最终拓扑为 GPU1 上完整 mixed-precision Catnip DiT，GPU0 上 Gemma、embedding、BF16 VAE 与 vocoder；视频 VAE 按 11 个 chunk 异步逐段解码。
- 历史 K32 MXFP8 baseline V2 在 832×480 合同下确立了 576 Linear 的 native K32 MXFP8、BF16 output、fused bias、0 fallback/upcast 与 48 个自研 SM120 attn1；它是当前 Portrait stack 的前序工程证据，而非 current headline。
- 2026-07-26 的 Portrait v1 三次 fresh-cache mean 为 37.486982 stable-DiT / 33.070761 generated FPS；它现为历史前序基线，orientation comparison 仍只支持 throughput regime preserved。
- 该历史基线第一次给出了严格配对 A/B：三对交替顺序（B/K、K/B、B/K）的独立 fresh 进程，stable-DiT median gain **14.103773924659912%**、最小配对增益 13.995280960853007%、ratio CV 0.04551887470930984%。
- 正确 Streaming KV 为 first 4 LF + latest 6 LF。O0–O7 将 formal wall 从 25.629 s 降至 18.041 s，generated FPS 从 19.080 提升到 27.105；其后 16 MiB cuBLASLt workspace、自研 SM120 attention 与 K32 依次叠加到历史 832×480 的 **32.572402 generated FPS**。
- Week 3 的 16 MiB cuBLASLt workspace 已**固化**进 Week 8 的冻结启动环境，在 6 次 performance 与 8 次 calibration 运行中生效；它不再是待部署候选。
- 端到端 latent trajectory rel-L2 作为验收判据已被三**重证伪**：同一干预跨 cell 符号翻转、安慰剂优于处理组、效应量级与轨迹混沌一致。验收权威已改为 decoded 20 秒媒体的 reference-free 伪影面板。
- Week 6 的局部测量事实保留（exact-FP32 K32 local mean rel-L2 0.0175908 对比 tensorwise 0.0228185，改善 22.910%；K32 pow2/native E8M0 为 0.0230594，比 tensorwise 差 1.056%），但据此作出的部署判决已被推翻——native E8M0 K32 正是当前基线的 scale 编码。

不能横向比较的数字

本报告出现的 FPS 具有不同分子、合同与计时边界，任何两个之间都不能直接相除：当前 Portrait v2 37.557043 stable-DiT / 34.392314 generated；历史 v1 37.486982 / 33.070761；25 playback；O7 27.104924 generated；Week 3 27.370839 generated；历史 Week 8 tensorwise 28.344310 generated / 32.319421 stable-DiT；历史 Week 8 K32 32.572402 generated / 36.866330 stable-DiT；以及插桩 Nsys

表 1: 关键数字与正确解释

指标	结果	解释边界
当前 Portrait generated throughput	34.392314 generated FPS	v2 三对 Phase 3 candidate mean，CV 0.085863%；489 / (sampling + pipelined video decode)，warm steady-state 口径
当前 Portrait stable DiT	37.557043 stable-DiT FPS	v2 三对 candidate mean；288/ chunks 5–10 DiT wall，与 generated FPS 分子分母都不同
Portrait status	v2 engineering baseline ACTIVE	final r03 runtime/media/K32/V-direct/compile gates PASS；不声明 cold-start speedup 或 formal ground-truth quality acceptance
历史 landscape V-direct A/B	stable-DiT +1.289408%（中位）	三对 832×480 full-model A/B；仅为 engineering-performance pass，不是 Portrait comparison 或 quality acceptance
历史 K32 paired gain	stable-DiT +14.103774%（中位）	同栈 tensorwise control 的三对交替 A/B；解释 K32 采用，不替换 current KPI
历史 O7 交付	27.104924 generated FPS	三轮 run-level FPS 均值；已打包并通过稳定性验收
播放帧率	25 FPS	489 帧编码为 19.56 秒
GPU1 第一大 kernel 类	K32 scaled-mm GEMM 38.489%	稳态 chunk 5–10 envelope 的 kernel-sum share；加 support 共 40.966%；不是可直接消除的 E2E 比例
Attention useful throughput	168.677600 TFLOP/s	2026-07-20 全 PyTorch Flash SDPA 状态的测量，不描述当前自研 SM120 attn1 路径
权重 outlier 对 E4M3 误差	$\rho = -0.046$（不显著）	同一指标在 INT8 对照下 $\rho = 0.817$；INT8 outlier 直觉不能移植到 E4M3
解码媒体 calibration	8/8 运行 PASS，两 lane AUTO_PASS	仅覆盖四个自动伪影指标；两人盲评仍为 PENDING

31.907889 / 36.301144。Nsys 插桩本身会使历史 stable-DiT 与 generated 分别下降约 1.533% 与 2.040%，插桩运行不得用于对外性能口径。v2 数字只描述 warm steady state，fresh-process wall 不属于该 speedup claim。Portrait 与 landscape 的 orientation comparison 不是 paired causal A/B；历史 BF16 与任何 FP8 配置也不是同 commit、同合同、同 timer，本报告不拼接出虚假的 BF16 到 FP8 官方加速倍数。旧 NVML 100% 仅代表 busy，80.5144%/68.6384% 只表示 FA-style useful throughput；三类数字都不得写成 SM occupancy。Week 9 的 counter-free NCU 只给出 theoretical occupancy 与 wave facts，tensor-pipe、stall 与 achieved occupancy 仍未测得。

目录

插图

表格

第一部分

实验口径与证据边界

1

模型身份、统一合同与实验方法

1.1 先把研究对象说清楚

LTX 目录下有两个不同对象：LTX-2.3 是基座模型；Catnip 是基于它蒸馏得到的 Streaming model。本文后续的“Catnip”只指 Streaming model。LTX-2.3 作为基础 checkpoint、架构和 VAE 来源出现，但不与 Catnip 的推理结果混写。

1.2 最终 20 秒 Streaming 合同

表 1.1: 所有正式视频实验冻结的输入与状态合同

字段	冻结值
输出 geometry	832×480，landscape，489 帧，25 playback FPS，编码时长 19.56 s
请求时长	20.0 s；所有 GPU 视频实验以该长度为基准
Latent 与 chunk	62 LF；$[4, 4, 6, 6, 6, 6, 6, 6, 6, 6, 6]$，共 11 chunks
Denoise / commit	每 chunk 4 NFE，加 1 次 sigma-zero 完整 cache-commit forward
Transformer calls	$11 \times (4 + 1) = 55$
KV	first 4 LF + latest 6 LF；persistent history 最大 10 LF
Video VAE	11 次逐 chunk decode，8 LF decode history，FIFO ordered drain
Audio	视频队列完全 drain 后一次性 VAE + vocoder
Pipeline	GPU1 DiT，GPU0 copy/VAE worker，pending queue depth 3

历史合同曾使用 31 chunks、155 forwards 与 4+2 KV。4+2 会削减上下文并造成场景闪烁；因此历史数据只能用于追溯，不能与最终合同做严格 A/B。

1.3 三个 FPS 与四个时间边界

1. **Playback FPS**：MP4 播放速度，固定 25 FPS。

2. **Generated FPS**：489000/formal wall(ms)。Week 2/3 的“27 FPS”属于此定义。
3. **Stable DiT FPS**：288/chunks 5–10 DiT seconds，只描述稳定 DiT 段。

Formal timer 包含 11 chunk DiT、latent copy、11 次 video VAE、队列 drain 和尾部双卡同步；不包含 checkpoint load、Gemma、FP8 conversion、compile warmup、audio decode、mux 与冷进程启动。Nsys 的 kernel-duration sum、device busy union、CPU NVTX wall 和 shell E2E 是另外三种时间边界，禁止相加或互相替代。

1.4 证据等级与门禁

表 1.2: 证据等级

等级	定义	本报告中的例子
A	原始 run/trace、机器汇总、分析器和哈希仍存在	Week 3 profiling；Week 4–6 gate artifacts；2026-07-20 Attention V2；outlier 统计（576 层全量权重与 63,360 条 activation call records）；Week 7 SM120 MXFP8 门禁与 r04；Week 8 因果 lane 与质量面板；Week 8 K32 三对配对与 8 次 calibration；Week 8 设备分离 profiling（45 分析窗口、12 stage gate、3 attention count gate、5 sequence-pair gate 全 PASS）
B	详细冻结报告仍在，但早期 raw 已清理	Week 1 runbook；Week 2 O0–O7 waterfall；历史 BF16 三轮
C	在当前代码重建等价合同，不是历史 commit 复现	Week 6 fresh BF16 20 秒 oracle

吞吐候选先跑 exact production shape microbenchmark；通过后才允许三次独立 warmed 20 秒进程。变化小于 0.25% 或三倍 baseline CV 视为 neutral；显存增量不得超过 128 MiB；576 modules、55 forwards、4+6 KV、31,680 logical scaled-MM 与 0 fallback/upcast 都是 fail-closed 不变量。精度候选按 operator-local、propagated latent/block 与 decoded perceptual 三层分开，heldout 只有在 calibration 和 validation 全通过后才能解锁。

判据边界的一次修订：2026-07-22 起生效

上述三层结构保留，但**中间层的验收资格已被撤销**。第 13 章给出的三重证伪表明，端到端 propagated latent/trajectory rel-L2 在这个 4-NFE 蒸馏流式系统上不能 accept/reject 任何候选：同一干预跨 cell 符号翻转，安慰剂在同一度量下优于处理组，全部效应落在与轨迹混沌同量级的 $\pm 3 \sim 20\%$ 区间。自该日起它只能

作为“轨迹分岔幅度”的描述性指标。
仍然有效的是**局部单-forward oracle-L2**：在固定输入的单次前向上它是单调的，Week 6 的 320-strata 协议因此继续成立。新的验收权威是 decoded 20 秒视频/音频的 reference-free 伪影面板：以 BF16 与生产 FP8 之间的自然差异带归一化，孤立有害偏差严格大于 2.0 报警，同一有害指标在两个及以上 cell 超过 1.0 亦报警。凡以旧判据作出的 accept/reject 结论，在本报告中一律标注为“判据失效、需重审”，而不是静默保留。

1.5 整份报告的阅读路径

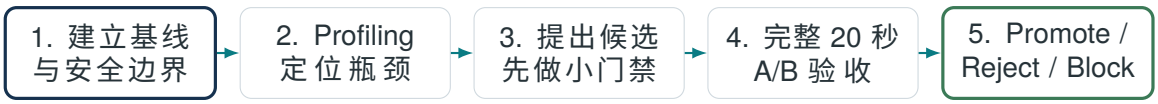

图 1.1: 后续每个阶段统一采用“动作–结果–分析–下一步”的叙事结构。

第二部分

从容量可行到 27 Generated FPS

2

第一轮 Runbook：先确认问题是什么

这一轮的价值不是形成最终性能数字，而是把“容量、算子、双卡拓扑、VAE 和 profiler”从一个模糊问题拆开。早期 raw 目录后来已经清理，因此本章属于等级 B 历史证据；凡无法从现存报告重建的细节都明确保留为证据缺口。

Step 1 | Phase A：仓库自动侦察与安全约束

动作 定位 BF16 推理入口、FP8-cast、KV cache、VAE、双卡 placement、计时器和 profiler；模型权重只读，所有实验产物写入独立结果目录。

Profiling / 结果 确认历史“24/24 pipeline”实为 block model-parallel：blocks 0–23 在 GPU0，24–47 在 GPU1；每个 forward 仍先后串行执行两半，并在 block 23/24 边界搬运完整 state。

分析 它解决的是 BF16 容量，不会让两张卡同时执行同一个 DiT forward。性能研究必须把 model-parallel 与真正的 DiT–VAE producer/consumer overlap 分开。

下一步 / 决策 先建立 BF16 sanity baseline，再审计实际 Linear shapes 与 native scaled-MM 能力。

Step 2 | Phase B：历史 BF16 基线

动作 运行旧 Streaming 合同下的 BF16 双卡推理，记录 sampling 与 shell E2E。

Profiling / 结果 sampling 为 42.947 s / 11.386 FPS；E2E 为 85.771 s / 5.701 FPS。

分析 该路径使用 31 chunks、155 forwards 与 4+2 KV，不能与最终 11 chunks、55 forwards、4+6 KV 的 generated FPS 相除。它只回答“历史 BF16 能否跑通”和数量级。

下一步 / 决策 Phase C 重新以 20 秒为基准做 profiling，但将 profiler 数据与吞吐数据分离。

Step 3 | Phase C：20 秒 Profiling

动作 把 20 秒请求中的稳定 chunk 分解为 attention、FFN、跨模态 attention 与 VAE。

Profiling / 结果 历史稳定 chunk 归因约为 attention 46.79%、FFN 25.59%、AV cross 17.61%；video VAE 完整解码约 16.7 s。Profiler 使目标 chunk 延迟增加约 69.6%。

分析 attention 与 FFN 是 DiT 主体，VAE 也足够大，具备跨卡重叠价值。由于 profiler 扰动很大，trace 只用于类别与 shape 归因，不能充当 FPS 基线。

下一步 / 决策 在 Phase D 枚举 Linear 调用和 GEMM shapes，再决定哪些算子值得量化。

Step 4 | Phase D：Linear 与 GEMM Shape 审计

动作 统计历史请求中的 Linear 调用、唯一 GEMM shape 与双卡 block CUDA work。

Profiling / 结果 约 294,500 次 Linear 调用、44 个唯一 shape；GPU0/GPU1 block CUDA work ratio 约 0.988。

分析 两半计算量接近，却仍被 24/24 串行拓扑限制。不同 shape 的 M 差异很大，小 audio shape 的动态量化成本可能超过 FP8 GEMM 收益。

下一步 / 决策 禁止“All Linear 一键 FP8”；先对真实 shape 做 capability 与收益 allowlist。

Step 5 | Phase E：审计旧 FP8-cast

动作 逐层检查旧 FP8 路径究竟执行了什么 GEMM。

Profiling / 结果 旧路径主要是 FP8 权重存储后上转 BF16，再执行 BF16 GEMM；top-shape native probes 的局部 speedup 约 1.015×–2.155×。

分析 FP8 storage 能降低静态权重显存，但不是 W8A8 Tensor Core 计算。局部 shape speedup 也不代表完整视频会等比例加速。

下一步 / 决策 下一周实现 E4M3FN weight + activation 与 aten._scaled_mm，同时保留 BF16 output 和零 fallback。

Step 6 | Phase F–H：缓存、自动化与证据边界

动作 继续审计 persistent cache、runbook 安全门与周报产物；按用户要求不执行 Phase I FP8 KV storage POC。

Profiling / 结果 历史 persistent cache 稳态约 5.210449 GiB。Phase 13/14 的汇总与图表流程完成，但 Phase F/G 的部分逐配置 raw 已删除。

分析 缓存已经是静态权重以外的主要容量项；没有 raw 的配置不能在新报告中补造精确结果。

下一步 / 决策 把“所有正式实验固定 20 秒”和“证据等级”写入后续协议；Phase I 保持未执行。

表 2.1: Runbook 第 13/14 部分的执行与当前可复算性

要求项	当时状态	当前仍可引用的结果	当前可复算
13.1 Chunk latency	历史流程标记完成	Phase C 类别/稳定 chunk 汇总仍在；逐配置 CSV 已清理	否，Grade B
13.2 Peak VRAM	历史流程标记完成	Persistent cache 稳态约 5.210449 GiB；逐配置 GPU0/GPU1 bars 的 raw 不在	否，Grade B
13.3 模块耗时	完成	Attention 46.79%、FFN 25.59%、AV cross 17.61% 的历史归因仍在	仅汇总
13.4 GEMM microbenchmark	完成	Top-shape native probes 约 1.015×–2.155×	仅汇总
13.5 量化误差	历史流程标记完成	早期逐 chunk BF16/FP8 曲线 raw 已清理；后续 Phase 4 有独立 backend 误差证据	早期图不可复算
13.6 KV cache 增长	历史流程标记完成	稳态总量仍在；逐 chunk video/audio/PE 拆分 CSV 已清理	否，Grade B
14 周报与 experiment table	完成	冻结结论已并入 runbook/综合报告；原输出树未保留	不能逐字节重建

表 2.1 的“完成”只复述现存 Grade B 冻结报告，不把已经删除的 raw 伪装成可复算证据。后续 Week 3–6 因此改为同时保留 JSON/CSV、runner、pre-run hash 与最终报告。

这一轮能证明什么

Week 1 证明了旧 BF16 能跑、24/24 是串行 model-parallel、attention/FFN/VAE 是主要方向、旧 FP8-cast 不是 native W8A8。它没有形成最终 4+6 合同，也没有提供可与 27 FPS 严格比较的 BF16 基线。[E-W1-RUNBOOK]

3

第二轮 Phase 0–4：建立可比较的 W8A8 实验台

Step 7 | Phase 0–1：独立工作区、人物 Prompt 与统一 Runner

动作 创建独立 weekly 目录；冻结人物相关 prompt、seed、initial noise、timer、telemetry、NVTX、结果 schema 和失败即停止规则。

Profiling / 结果 性能、quality 与 profiling 被拆成不同 run；Phase 2/3 共用人物 prompt，所有正式视频固定 832×480、20 秒。

分析 没有统一输入、随机性与 timer，任何“更快”都可能只是合同变化。

下一步 / 决策 先跑 BF16 三轮容量参考，再以同一组 production shapes 建 native backend。

Step 8 | Phase 2–3：BF16 三轮与 Shape 冻结

动作 2026-07-13 对 24/24 BF16 sampling-only 运行三轮，并复核历史 44 shapes 与热点。

Profiling / 结果 三轮 wall 为 43.229055 / 43.249080 / 43.245502 s；sampling FPS 为 11.311837 / 11.306599 / 11.307534，中位 11.307534。

分析 三轮很稳定，但仍是旧 31-chunk、4+2 路径，因此只作为容量/采样参考；不能作为最终 FP8 formal timer 的分母。

下一步 / 决策 进入 Phase 4，逐 shape 验证真正的 W8A8 backend。

Step 9 | Phase 4：第一版 Native W8A8 Backend

动作 实现静态 tensorwise E4M3FN weight、逐调用动态 tensorwise activation、原生 aten._scaled_mm、BF16 output、独立 BF16 bias、零 fallback/upcast。

Profiling / 结果 36 shapes × 2 GPUs，共 72/72 capability probes 通过；relative-L2 0.028111–0.039523，cosine 0.999227–0.999605。第一版选择 240 modules，all-Linear 参数覆盖约 46.6742%，估算 FLOP 覆盖约 54.8653%，静态节省约 8.25 GiB。

分析 这证明 native W8A8 可执行，但小 M audio shapes 不值得量化。240 modules 仍未证明完整 DiT 能单卡运行，也没有完整 FP8 视频或 FPS。

下一步 / 决策 拿 240-module manifest 做完整 mixed DiT 单卡 placement；下一步只回答容量可行性。

Phase 4 的关键纠偏

“FP8 权重占显存”与“W8A8 GEMM 在算”是两件事。最终 backend 的核心调用是

$$Y_{\text{BF16}} = \text{scaled\_mm}(X_{\text{E4M3}}, W^{T}_{\text{E4M3}}, s_x, s_w, b_{\text{BF16}}),$$

而不是先把 W_{FP8} 转回 BF16 再做普通 GEMM。[E-W2-HOW]

4

完整 DiT 单卡：从三次 OOM 到正确 Streaming

Step 10 | 240 Modules：静态权重省了，但状态下界仍超容量

动作 将完整 mixed Catnip DiT 放到 GPU1，Gemma/VAE 放 GPU0，连续三次运行 240-module manifest。

Profiling / 结果 mixed DiT 为 29,119,307,200 B；KV/PE floor 约 5,594,677,248 B；合计比 GPU1 物理容量多 994.898 MiB。三次均在 chunk 0 首个 forward OOM。

分析 报错落在下一次约 14 MiB bias allocation，并不意味着 bias 是根因；模型参数加必需状态的下界已经超卡。

下一步 / 决策 扩大只对大 M 高收益路径的 FP8 coverage，而不是通过 allocator 参数掩盖结构性超量。

Step 11 | 480 Modules：静态容量通过，生命周期峰值仍 OOM

动作 扩到 480 modules，将 mixed DiT 降到 25,898,082,688 B。

Profiling / 结果 Linear weight coverage 达 63.651684%；chunk 0 能完成，但 chunk 1 NFE0 OOM。OOM 时 allocated/reserved 为 31,185,477,632 / 32,933,675,008 B；KV/PE 解释 chunk 1 增量的 86.783%。

分析 根因从“静态参数下界”转成 append-before-evict 的 KV/PE lifetime peak。碎片整理或移动一个 bias 不能解决持续增长状态。

下一步 / 决策 实现 bounded KV、共享 PE 和 fused bias，先关闭显存生命周期问题。

Step 12 | Bounded KV、Shared PE 与 Fused Bias

动作 把 persistent KV 改为容量受控的 sink+recent，PE 只保留共享实例，bias 融入 scaled-MM epilogue。

Profiling / 结果 pre-fusion control 虽跑完，但 reserved 达 95.0268%，超过 95% memory gate；fused treatment 为 94.0863%，11 chunks 完成。单个 FFN-down bias fusion 将局部增量 peak 从 25,559,040 B 降至 12,779,520 B。

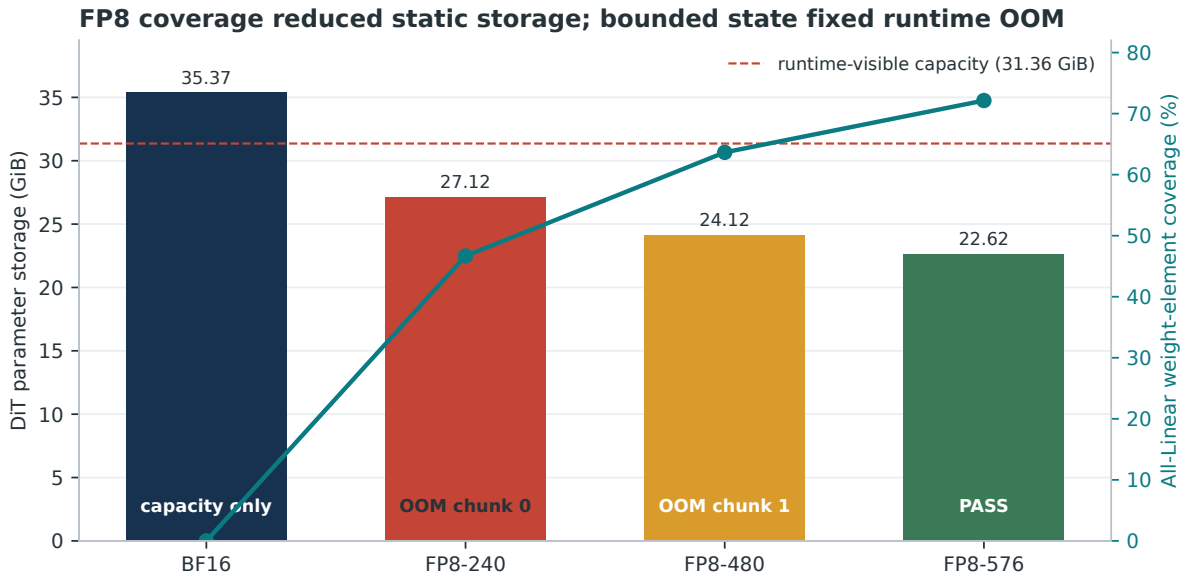

图 4.1: FP8 coverage 降低了静态 storage；真正让 20 秒请求通过的是随后对 KV/PE 生命周期的约束。

分析 这一组改动关闭了容量风险，但当时仍是旧 4+2、serial decode；35.568 s 不能作为最终性能数字。

下一步 / 决策 容量问题暂时关闭，下一步审计生产 MaineCoon 的真正调度与 VAE 语义。

Step 13 | 审计生产推理：VAE 是逐 chunk，不是最后统一 decode

动作 阅读 MaineCoon 生产脚本的 DiT、latent copy、VAE 与 audio drain 逻辑。

Profiling / 结果 视频 VAE 每个生成 chunk 解码一次，携带最多 8 LF history；audio latent 则在全部视频队列 drain 后一次性 VAE/vocoder。生产 BF16 的双卡并发是完整 DiT 与前一 chunk VAE 的功能流水线。

分析 最终 FP8 不应继续 24/24 搬完整 Transformer state；完整 DiT 放 GPU1、GPU0 解码前一 chunk，跨卡只传 latent，才有真正 overlap。

下一步 / 决策 修正 KV 为 first 4 + latest 6，并实现 GPU0 单 worker 异步 VAE。

4.1 最终 bounded KV 语义

Listing 4.1: 最终 KV 语义，不是历史 4+2

```
persistent sink   = first 4 latent frames
persistent recent = latest 6 latent frames
max history       = 10 latent frames

for each chunk:
```

```
run 4 denoise forwards without persistent commit
run 1 full sigma-zero forward
write bounded pending cache
commit first-4 + latest-6
```

当前 chunk 与 history 同时可见时窗口可达 16 LF，但 persistent history 上限始终是 10 LF。曾经误用 4+2 会减少长期上下文并造成闪烁，因此所有正式 wrapper 都对 sink=4、recent=6、capacity=10 fail-closed。

Step 14 ｜ False Overlap 修正与 O0 冻结基线

动作　先尝试在主线程用不同 CUDA stream 提交 VAE；发现 Python 侧 vae_submit_ms 约 1.25 s，仍阻塞下一 chunk DiT。随后把 VAE launch 移入单独 host worker，GPU0 copy/VAE 与 GPU1 DiT 用 event 串联，pending depth 固定 3。

Profiling / 结果　第一版 structured prompt 同时喂给 video/audio，conditioning r01 作废；修正 audio prompt 后得到 O0：25,628.835600 ms / 19.080070887 generated FPS。DiT CUDA sum 24.041768 s，VAE CUDA sum 12.418855 s，两者已大量重叠。

分析　“用了两个 CUDA stream”不等于真正异步；host launch 也必须离开关键线程。O0 首次同时满足正确 4+6、逐 chunk VAE、固定 20 秒和明确 timer。

下一步 / 决策　以 O0 为唯一后续 O1–O7 起点，优化过程中冻结 Streaming 合同。

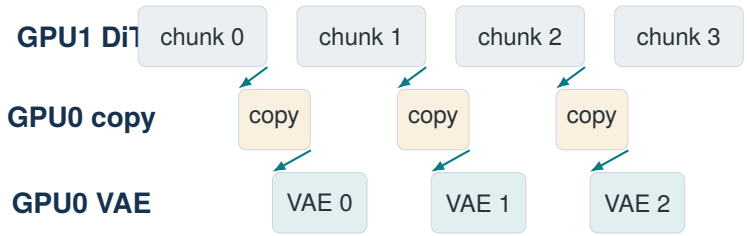

图 4.2: 最终双卡不是“两张卡各算一半 DiT”，而是 GPU1 生成下一 chunk 时 GPU0 解码前一 chunk。

5

O0–O7：从 19.08 到 27.10 Generated FPS

5.1 先看完整瀑布，而不是只挑成功项

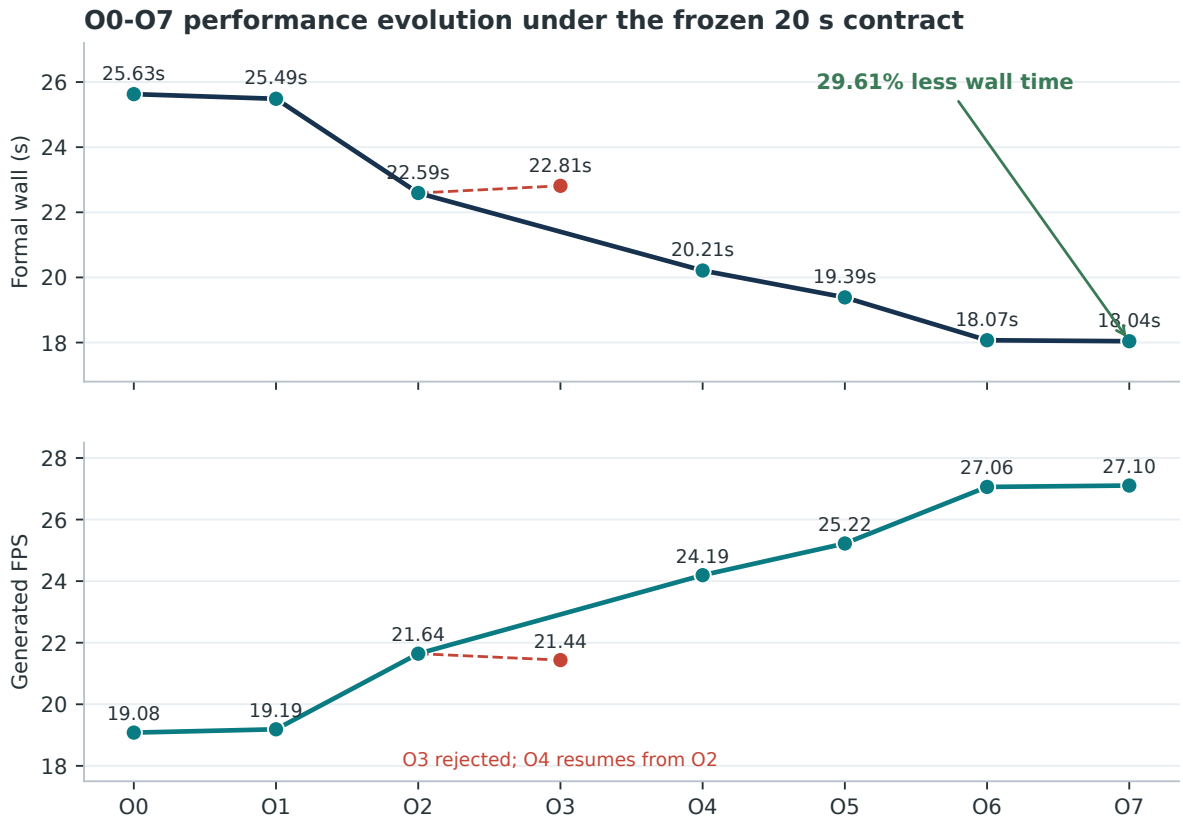

图 5.1: O0–O7 的 formal wall 与 generated FPS。O3 的红点表示单 shape 有收益、完整请求却回退。

O0 到 O7 formal wall 减少 7,587.829 ms / 29.6066%, generated FPS 增加 42.0588%。这是冻结 workload 与 timer 下的 observed waterfall，不是严格 factorial 因果分解；尤其 O5 同时改变了 FP8 coverage。

Step 15 | O1：移除观测开销

动作 关闭 performance hooks、细粒度 NVTX 和 Python audit counter。

Profiling / 结果 wall 减少 143.264 ms，FPS 19.0801→19.1873。

分析 历史命令没有完整记录 allocator 配置，不能把全部 143 ms 断言为 hooks

表 5.1: O0–O7 历史实验瀑布

阶段	主要变化	Wall (ms)	Generated FPS
O0	正确 4+6 + async VAE	25,628.836	19.080071
O1	关闭 hooks / NVTX	25,485.571	19.187327
O2	48 blocks 分别 compile	22,592.952	21.643918
O3	selective cuDNN SDPA，拒绝	22,813.005	21.435142
O4	折叠重复 V2A attention	20,212.044	24.193496
O5	480→576 FP8 Linear	19,387.098	25.222960
O6	PE metadata key + 5-shape warmup	18,070.993	27.059941
O7	strict entry 与稳定化	18,041.006	27.104924

的独立因果收益。

下一步 / 决策 保留工程清理，进入 block-level compile。

Step 16 | O2：48 Blocks 分别 torch.compile

动作 每个 Transformer block 单独 compile，warmup 与 formal timer 分离。首次 r01 在 warmup chunk 2 因约 2 GiB reserved-but-unallocated OOM；启用 expandable segments、冷 cache 后重跑。

Profiling / 结果 r02 为 22,592.952 ms / 21.643918 FPS，比 O1 少 2,892.619 ms；formal graph delta 为空。

分析 收益来自 normalization、pointwise、reshape 与量化 support graph 的融合和 launch 减少；外部 aten._scaled_mm GEMM 并没有被 48 层融合成一个 kernel。

下一步 / 决策 保留 compile，下一步测试 attention provider。

Step 17 | O3：Selective cuDNN SDPA

动作 只对 microbenchmark 中 D128、$Q \geq 512$ 的候选 shape 选择 cuDNN SDPA。

Profiling / 结果 单 shape 约快 19.7%，完整 20 秒请求却变成 22,813.005 ms，比 O2 慢 220.053 ms。

分析 shape mix、workspace、后续 kernel 与调度交互抵消了局部收益；microbench 没错，但问题边界太小。

下一步 / 决策 **REJECT**；恢复 PyTorch Flash，以完整请求门禁为准。

Step 18 | O4：折叠重复 V2A Attention

动作 跳过第一次不写 cache 的重复 V2A 调用，复用第二次有 cache 副作用的 output，并保持两次 residual add 顺序。

Profiling / 结果 20,212.044 ms / 24.193496 FPS，比 O2 少 2,380.908 ms。局部 output/cache exact-equal；最终相对 O2 的 PSNR 21.233598 dB、SSIM 0.721615、audio correlation 0.351360。

分析 这是重复计算消除，不是 kernel fusion。性能显著，但最终轨迹并非 bit-identical，不能据此宣称 BF16 感知等价。

下一步 / 决策 保留性能路径，继续扩大对高收益 video K/V 的 FP8 覆盖。

Step 19 | O5：480→576 FP8 Linears

动作 每层新增 attn1.to_k/to_v 两个 Linear，共 96 modules。

Profiling / 结果 19,387.098 ms / 25.222960 FPS，首次超过 25 generated FPS；最终 selected weight elements 为 13,690,208,256，占全部 Linear weight 72.138575%。

分析 这些 K/V 具有足够大的 video M，scaled-MM 收益覆盖动态 activation quant；但 O5 不是“量化策略完全不变”的单变量 A/B。

下一步 / 决策 保留 576-module manifest，下一步消除 graph key 与 warmup 漏洞。

Step 20 | O6：PE Metadata Key 与五 Shape Warmup

动作 将 PE memo key 从 storage identity 改成稳定 metadata key，并让 warmup 覆盖五种正式 shape。

Profiling / 结果 只 warmup 四种状态的 r01 在 formal chunk 4 新编译 6 graphs，chunk DiT 51.927 s、总 wall 68.435 s，整轮作废。五 shape r02 为 18,070.993 ms / 27.059941 FPS，formal graph delta 为空。

分析 O5→O6 的 1.316 s 包含 key、完整 warmup 与交互，不能全归因于一个 metadata key。

下一步 / 决策 把所有 formal shapes 编译完再计时，进入交付稳定化。

Step 21 | O7：Strict Entry 与三轮验收

动作 固化 safe helper、strict entry、allocator-aware serial warmup。首次 cold run 在 warmup chunk 3 RoPE buffer OOM，改 serial warmup 与清理后重跑。

Profiling / 结果 三轮 wall 为 18,031.774 / 18,052.090 / 18,039.155 ms；FPS 为 27.118797 / 27.088276 / 27.107700。mean 18,041.006 ms / 27.104924 FPS，FPS CV 0.046536%。三轮 decoded-video hash 相同。GPU1 peak allocated/reserved 为

29,968,928,768 / 30,624,710,656 B。

分析 固定输入下结果稳定，且运行余量比早期 feasibility 更可靠；hash 一致只证明候选自身确定性，不证明 BF16 等价。

下一步 / 决策 形成 `week2-improved-fp8_scaled` 交付包；下一周以三轮 frozen baseline 开始 profiling-driven 优化。

27 FPS 的准确含义

$$\overline{\text{generated FPS}} = \frac{1}{3}\sum_{i=1}^{3}\frac{489000}{t_i(\text{ms})} = 27.104924.$$

发布值是三轮 run-level FPS 的算术平均；三轮 wall 的平均为 18.041006 s，而 $489/18.041006 = 27.104918$，两种聚合不可混写。它表示 formal timer 内生成 489 个输出帧的速度略快于实时；MP4 仍按 25 FPS 播放。模型加载、Gemma、FP8 conversion、约 145 s compile warmup、audio decode 和 mux 不在该计时器内。

[E-W2-HOW]

第三部分

Profiling 驱动的吞吐优化

6

Week 3：冻结基线、筛选候选，再做完整 Profiling

6.1 冻结三轮基线与 promotion gate

Step 22 | 建立 Week 3 Frozen Baseline

动作 用新的独立进程跑三次 warmed 20 秒请求；冻结 576 modules、4+6 KV、55 forwards、31,680 logical scaled-MM、media contract 与显存门。

Profiling / 结果 median formal wall 18,021.708801 ms，median generated FPS 27.133941925，wall population CV 0.028937%；GPU1 peak allocated/reserved 27.910740 / 28.521484 GiB。

分析 小于 0.25% 或三倍 baseline CV 的变化无法覆盖运行噪声与工程复杂度；这成为 promotion gate。

下一步 / 决策 候选先做 exact-shape micro/feasibility，达到理论收益门才允许三次 full request。

Step 23 | 六类候选：完整请求决定去留

动作 依次测试 external M padding、shape-specific fast-accum、shared activation quant、text K/V cache、简单 QKV/KV concat 和 cuBLASLt workspace。

Profiling / 结果 只有 16 MiB provider workspace 达到 full-request promotion gate。Padding 与简单 fusion 在 micro/ceiling 阶段早停；fast-accum、shared quant 和 workspace 进入三轮 full request。

分析 局部 kernel 加速不等于请求加速；容量、轨迹变化和工程复杂度与 wall 同时作为门禁。

下一步 / 决策 固化 provider-only 候选；其余进入失败尝试账本，不重复烧 20 秒视频。

表 6.1: Week 3 候选的最终状态

候选	关键结果	Full runs	决定
External M padding	三大 shape 均慢；预计每请求 +52.517 ms，尚未计 pad/crop	0	**REJECT**
Fast-accum	square micro 1.0237×；full median 慢 2.402 ms / 0.0133%	3	**REJECT**
Shared activation quant	physical quant calls -25%；full wall -0.1664%，低于 gate 且改轨迹	3	**REJECT**
Text K/V cache	48 层需 3,840 MiB；9 层 720 MiB 也超过 memory gate	0	**REJECT**
Simple QKV/KV concat	未扣后处理的乐观 ceiling 仅 0.173474%	0	**REJECT**
Provider workspace 16 MiB	wall -0.865511%，FPS +0.873067%，GPU1 allocated +30 MiB	3	**PROMOTE**

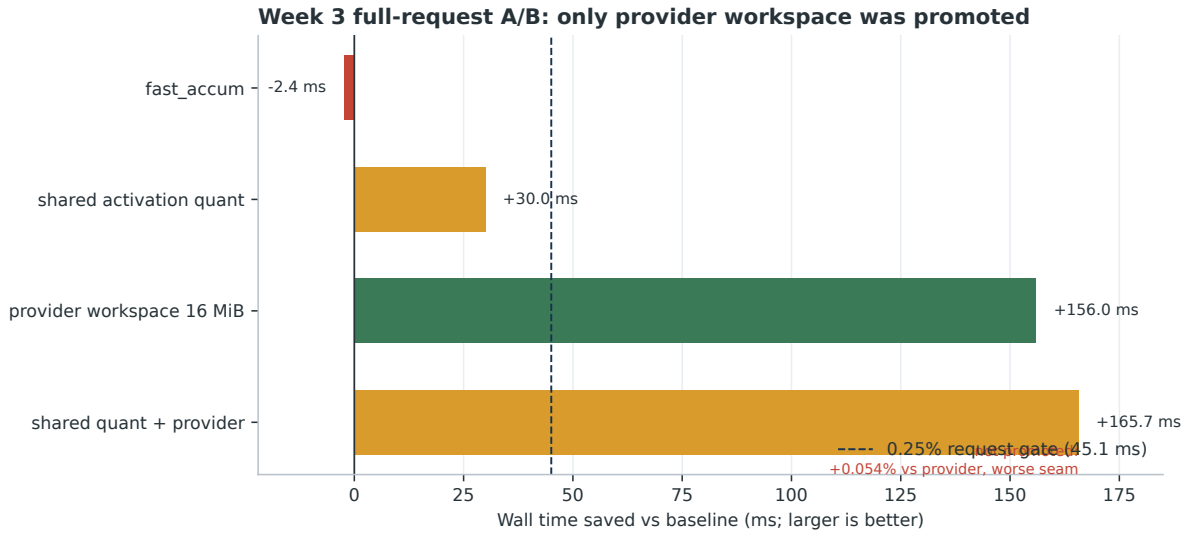

图 6.1: 相对 frozen baseline 节省的 full-request wall。Shared+provider 总收益更大，但相对 provider-only 仅多 9.716 ms / 0.0544%，且 seam 诊断更差，因此没有上线。

Step 24 ｜唯一 Promote：16 MiB cuBLASLt Workspace

动作　在首次 CUDA 初始化前设置 CUBLASLT_WORKSPACE_SIZE=16384，不改变 scale、weight、coverage、KV、scheduler、VAE 或 output dtype。

Profiling / 结果　median wall 17,865.729011 ms，27.370839427 generated FPS；相对 baseline 为 -155.979790 ms / -0.865511%，FPS +0.873067%。Stable chunks 8–10 DiT CUDA -1.104%；GPU1 allocated +30 MiB、reserved +16 MiB。FFN-down micro 从 785.254 μs 降至 725.942 μs，约 1.0817×。

分析　额外 provider 搜索空间改变 cuBLASLt algorithm 选择，主要收益落在 FFN-down。三轮均保持 0 fallback/upcast 与完整合同；固定 seed 得到稳定的新 trajectory，但不与 baseline bit-identical。

下一步 / 决策　作为 Week 3 推荐配置；部署脚本仍需内置环境检查并重新做三轮 fresh acceptance。

6.2　完整 Nsys/Kineto 归因

Step 25 ｜对推荐配置采完整 Nsys

动作　对 provider-only build 采 full request Nsys；随后从原始 SQLite 只读离线复算。

Profiling / 结果　profiled wall 17,962.078 ms，比 unprofiled median 慢 96.349 ms / 0.5393%。GPU1 busy union 89.298%，GPU0 为 66.825%；GPU1 是关键路径。

分析　Profiler 扰动小于 1%，足以做 kernel 类别归因，但最终 FPS 仍必须来自 unprofiled runs。

下一步 / 决策　按 GPU1 kernel-duration share 排序下一批方向，而不是继续凭直觉改代码。

Step 26 ｜ Kineto 只回答 Exact Shape，不回答生产 FPS

动作　对 uncompiled chunk 10 采 Kineto，建立 scaled-MM shape/call attribution。

Profiling / 结果　精确得到 $2,880 = 576 \times 5$ calls；主要包括 $[2340, 4096] \times [4096, 4096]$ square、text K/V、FFN up/down 与 A2V q/out。完整请求合同为 $576 \times 55 = 31,680$，Nsys 独立观察一致。

分析　该 trace 故意 uncompiled，适合识别 shape 与 operator，绝不能替换 compiled Nsys 时间。

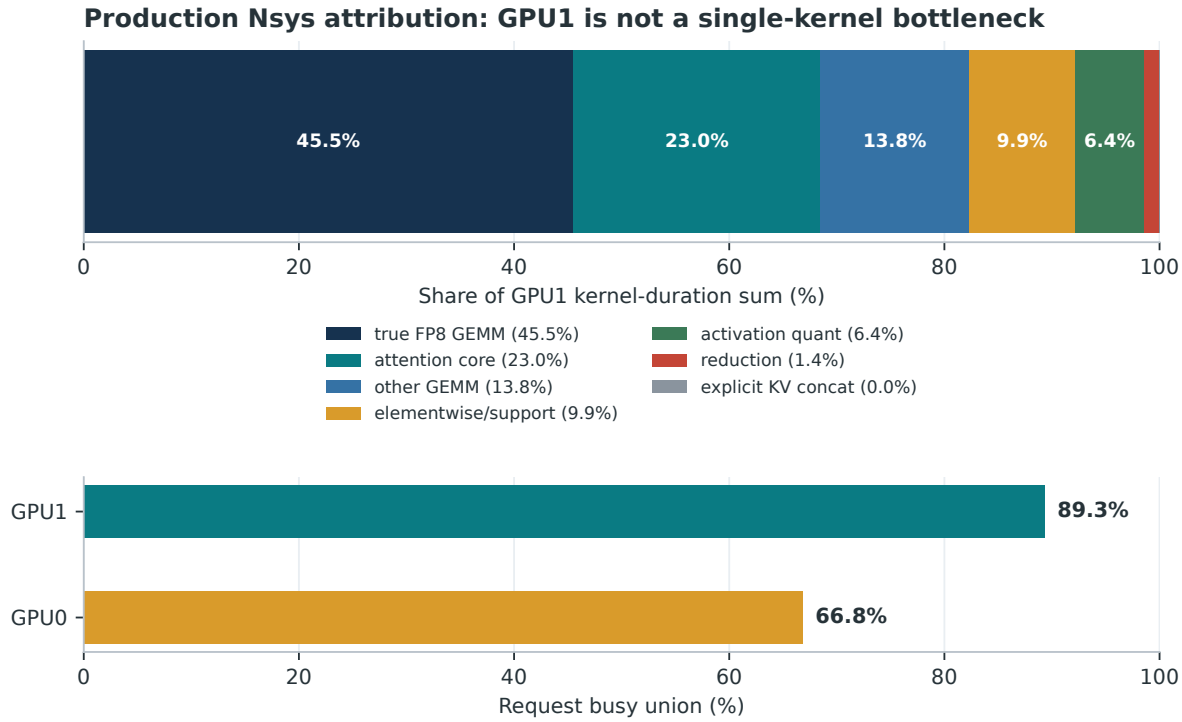

图 6.2: 推荐 provider 配置的 GPU1 kernel-duration 构成与双卡 busy union。这里的 share 不是 SM 利用率，也不是可直接消除的 E2E 比例。

表 6.2: GPU1 完整请求 kernel 分类

类别	Calls	GPU time (ms)	Share
True FP8 GEMM	31,680	7,295.072	45.4866%
Attention core	21,120	3,693.002	23.0268%
Other GEMM	83,020	2,207.063	13.7616%
Elementwise/support	83,168	1,588.672	9.9058%
Activation quant	79,200	1,019.437	6.3564%
Reduction	42,248	232.412	1.4491%
Explicit KV concat	1,045	2.180	0.0136%

下一步 / 决策　吞吐方向排序为 provider/FP8 GEMM、D128 attention、activation-quant support 和完整 shared-A grouped GEMM。

Profiling 后的实际判断

FP8 GEMM 是最大单项，但只占 GPU1 kernel-duration 的 45.49%。Attention、other GEMM、support 和 activation quant 合计超过一半。因此“换一个 FP8 GEMM kernel 就能整体快 45%”是错误推断。[E-W3-NSYS]

这个推断在 Week 8 的 K32 栈上被重新量化，结论同向但更严格：稳态 K32 scaled-mm GEMM 占 GPU1 kernel sum 38.489%、加 support 共 40.966%，而把全部稳态 scaled-mm GEMM 加快 10% 只对应 steady envelope 3.73% 的串行上限；三个 P0 shape 各加快 20% 约减少 89.4 ms/steady chunk 的设备归因。原因是两卡流水已接近均衡，任一卡 busy 达 99.640%，device-level share 与端到端收益之间隔着 GPU0 的并行执行。详见第 15 章。

第四部分

向 BF16 对齐的精度研究

7

Week 4：更细 Scale 能否稳定向 BF16 对齐

7.1 先把“精度”拆成三层

表 7.1: 精度问题的三层边界

层级	问题	典型指标
Operator local	单次 quantized Linear 离 BF16 Linear 多远	relative-L2、cosine、saturation
Propagated latent/block	误差经过 attention、residual、KV 与 55 forwards 后如何变化	video/audio p50/p95、velocity/latent rel-L2
Perceptual output	最终视频、音频与 AV sync 是否更好	decoded metrics、seam、盲评、业务指标

局部误差下降是好信号，却不是完整视频改善的充分条件。Streaming 系统会反复把当前输出写回后续状态与 KV；误差方向、残差抵消、跨模态耦合和 early-chunk tail 都可能反转单层收益。

Week 4–5 所称的“overall improvement”统一指**相对同 cell tensorwise FP8 的 BF16-relative trajectory-error 改善**：

$$I = 100 \times \frac{E_{\text{tensorwise}} - E_{\text{candidate}}}{|E_{\text{tensorwise}}|},$$

其中 E 是 gate artifact 中 matched overall trajectory relative-L2。正值只表示该 trajectory 指标更接近 BF16；它不等于 decoded perceptual quality。Median/worst 只在各 gate 自己冻结的 cells 上聚合，不同样本数的 gate 不能当作同一 population 排名。

Step 27 | Weight Output-Channel Scan

动作 对全部 576 modules 将 weight scale 从 tensorwise 细化到 output-channel，先只看权重重建。

Profiling / 结果 weight relative-L2 从 0.0264973 降到 0.0264341，仅改善 0.2385%；nonzero-to-zero 量化错误降低 95.16%。

分析 更细 weight scale 能救回大量小权重，但整体 relative-L2 变化很小；weight-only 指标不能推断 Linear output 或视频。

下一步 / 决策 进入 activation rowwise 单 cell probe，再决定是否扩大矩阵。

Step 28 | 单 Cell Rowwise 信号与音频反转

动作 在 person-car / seed 12345 上比较 tensorwise 与 rowwise activation。
Profiling / 结果 overall trajectory error 0.516014→0.440042,改善 14.723%;video trajectory MSE 改善 36.056%；audio rel-L2 却从 1.11933 变为 1.26212。K-only、V-only 与后续成功的 K+V BF16 rescue 均未改善 aggregate；K+V 的前两次 OOM 单独记为 Invalid。
分析 单格 video trajectory 有强信号，但同一改动已伤害音频，说明必须用多 prompt/seed 与 modality-aware gate；这里没有得出 decoded-video 感知质量结论。
下一步 / 决策 冻结六个 cells，按 Gate D/E/F 顺序测试全局策略。

Step 29 | Gate D：Hybrid-K

动作 528 个 activation rowwise,48 个 attn1.to_k 保持 tensorwise;weight output-channelwise。
Profiling / 结果 overall trajectory-error improvement 为 4 胜 2 负，median +0.506%，worst -20.320%；audio velocity/latent median 为 -7.679% / -8.666%。
分析 局部胜率不足以覆盖最差 cell，且音频系统性退化。
下一步 / 决策 **REJECT**；测试更保守的 tensor activation + channel weight。

Step 30 | Gate E：Tensor Activation + Channel Weight

动作 activation 保持生产 tensorwise，只细化 weight。
Profiling / 结果 overall trajectory-error improvement median +3.221%，worst -22.632%; stable DiT FPS worst -2.323%。Ceramics/12345 +6.703%, 换 seed 67890 反转为 -22.632%。
分析 中位数为正仍无法控制 seed 泛化，速度门也失败。
下一步 / 决策 **REJECT**；测试 full rowwise，确认是否由更一致的 activation/weight 粒度改善。

Step 31 | Gate F：Full Rowwise

动作 对允许模块使用全局 rowwise activation 策略。
Profiling / 结果 overall trajectory-error improvement median +0.385%，worst -7.563%；速度 worst -1.542% 通过，但 audio velocity/latent worst 达 -61.777% /

-65.655%。

分析 平均速度可接受，音频 tail 却不可部署；更细 scale 不是单调改善。

下一步 / 决策 **REJECT**；转向带 calibration 的 SmoothQuant。

Step 32 | SmoothQuant：Calibration 赢，Heldout 反转

动作 以 car/12345、ceramics/12345 calibration 搜索 alpha，另外四格只做 held-out 判断。首次 observer 使用超过 2^{24} 的 FP32 linspace index，触发 CUDA assert；改纯 INT64 后完成 63,360 records。

Profiling / 结果 选择 $\alpha = 0.0$；calibration overall trajectory-error improvement median +22.782%、worst +19.932%。Heldout median -2.991%、worst -18.750%；stable FPS worst -2.933%。

分析 这是典型 calibration overfit。不能拿 calibration 的 +22.8% 当最终精度提高，也不能从 heldout 重新选 alpha。

下一步 / 决策 **REJECT**；Week 5 改为只给敏感 family/top-N 路径更细 scale。

Week 4 收获

更细 scale 能在特定 cell 明显改善 video trajectory 指标，但没有任何全局 row/channel/SmoothQuant 策略同时通过相应 cell population 的最差值、音频与速度门。生产 tensorwise 继续保留，不是因为“细 scale 没用”，而是因为 trajectory 收益没有泛化；本轮也没有得到 decoded perceptual quality 的替代结论。[E-W4-FINAL]

8

Week 5：敏感路径、K32/K64 与时序路由

8.1 预注册数据切分与候选矩阵

表 8.1: Week 5 冻结的 cell split

Split	Cells
Calibration	chef-market / seed 24680；drummer-rooftop / seed 13579
Validation	tailor-workshop / seed 31415；skateboarder-plaza / seed 27182
Final-heldout	scientist-greenhouse / seed 42424；violinist-station / seed 77777

18 个候选在运行前注册：family、sensitive top-N、K32、logical K64 与五个 temporal policies。只有 calibration 和 validation 全通过的 survivor 才能组合；只有组合通过才允许访问 final-heldout。

该切分的后续状态：两格 final-heldout 已被消耗

截至 2026-07-21，本表与其下的 survivor 结论都成立。此后 Week 8 的第一版 reference-free 面板在构造自然差异带时使用了 scientist-greenhouse/42424 与 violinist-station/77777 两格，构成**泄漏**。V2 协议已记录该泄漏，把这两格重分类为 calibration（现 calibration 共四格），并另行保留两个不透明的 final-heldout-v2 占位符，需 hash-bound 书面解锁才能访问。

此外，本章 13 个 reject 中 K32 与 logical K64 的拒绝依据是端到端 overall trajectory-error improvement 跨 prompt 反转；该判据已于 2026-07-22 被证伪（第 13 章）。K32 随后在同栈配对实验中通过性能门禁并进入解码媒体 calibration，成为工程基线 V2 候选。

Step 33 ｜ 先验证 K-group Provider 能力

动作 区分 native K32、compact K64、K64-via-K32、naive split-K 与 fake quant。

Profiling / 结果 Native K32 单 GEMM 可执行；compact native K64 不支持，只能把一个 K64 scale 复制到两个 K32 provider slots。Naive split-K 慢 68×–135×；

fake quant 根本不是 FP8 GEMM。

分析 “每 64 个元素一个 scale”不代表 provider 真有 compact K64；实现形式会直接决定速度与可比较性。

下一步 / 决策 只允许 native K32 和明确标注的 logical K64-via-K32 进入视频 gate。

Step 34 | Family 与 Sensitive Top-N

动作 先在 calibration 上测试 A2V row/channel、sensitive top-48/96/192、video FFN 等静态模块集合。

Profiling / 结果 A-row A2V 在 chef/drummer 为 +2.221% / +9.174%，进入 validation 后 tailor -12.150%、audio latent -49.179%。W-channel A2V tailor -10.925%；top-48 在 drummer audio 为 -2.299% / -2.456%；top-96、top-192 与 video FFN 在首个门禁即负。

分析 local activation probe 的 mean rel-L2 从 0.0263162 降到 0.0245079，但单 record 变化范围 -31.515% 到 +70.263%，无法预测端到端传播。

下一步 / 决策 全部静态 family/top-N 候选 reject；继续检查 K32/K64 是否兼具速度与 BF16-relative trajectory error。

Step 35 | K32/K64：速度稳定，Trajectory Error 跨 Prompt 反转

动作 在 chef 与 drummer 上运行 tensorwise、K32、logical K64-via-K32。

Profiling / 结果 K32 overall trajectory-error improvement 为 +0.827% / -7.064%，stable-DiT FPS 约 +14.3%；K64 为 +17.009% / -9.357%，stable-DiT FPS 约 +14.0%。

分析 Block-scaled provider 确实有性能潜力，但 scale 选择让误差传播方向随 prompt 反转。这里的约 34 stable-DiT FPS 不能与 Week 3 的 27.37 generated FPS 横比。

下一步 / 决策 K32/K64 均 **REJECT**；性能信号留给下一轮因果拆分。

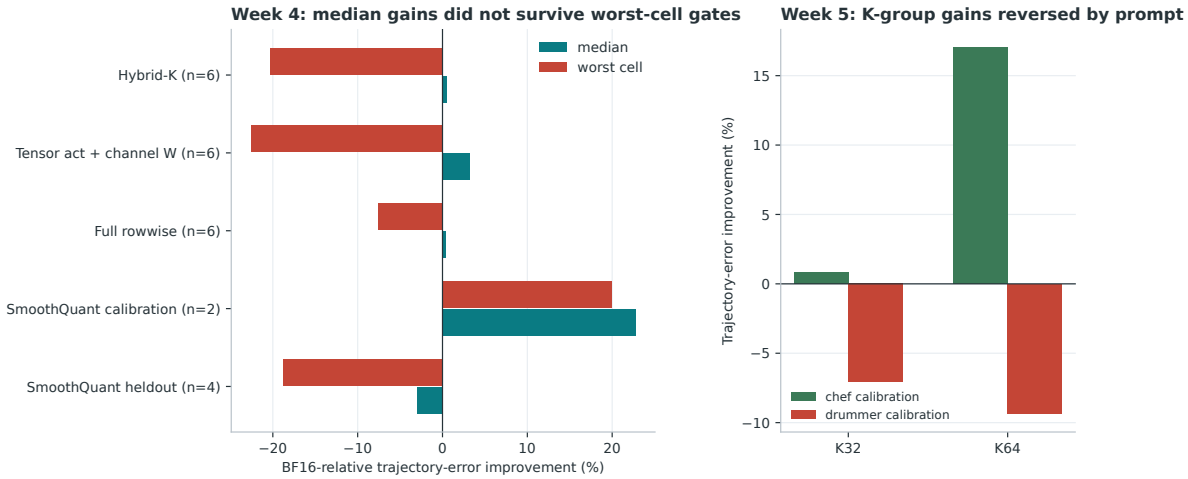

图 8.1: BF16-relative trajectory-error improvement。左侧 Week 4 的 Hybrid-K、channel-weight 与 rowwise 各为 6 cells，SmoothQuant calibration 为 2 cells、heldout 为 4 cells，因此只展示各自 gate 的泛化失败，不能横向排名；右侧 K32/K64 使用同一 chef/drummer 两个 calibration prompts，方向发生反转。

Step 36 | Dynamic torch.cond Temporal Routing

动作 用 bit-exact、无 graph-break 的 dynamic branch，在不同 timestep/chunk 选择更细 scale。

Profiling / 结果 Attention overhead 10.1885%–13.1055%，FFN-up 15.9881%–20.9805%，FFN-down 17.1468%–18.8054%。

分析 数值语义正确，但 CUDA overhead 远超门槛。五个 temporal candidates 的视频 launch 为 0，是协议早停，不是遗漏。

下一步 / 决策 **BLOCKED**；未来若重启只能用静态预绑定 callable，而不是原 dynamic branch。

Week 5 的严格终态

13 个候选实际执行后 reject，5 个 temporal candidates 被 overhead gate 前置阻断；survivor set 为 0，因此 combination runs 严格为 0，candidate final-heldout launches/comparisons 也为 0。未访问 heldout 是协议正确行为。[E-W5-FINAL]

9

Week 6：把 Granularity 与 Scale Encoding 拆开

9.1 Phase 0–3：先修实验台，再谈结论

Step 37 | Phase 0：Provider 与完整调用合同审计

动作 重新核验 576 modules、31,680 calls、11 shapes 与 native K32 eligibility。
Profiling / 结果 首轮 finalizer 因证据记录不完整给出 ambiguous/fail；补全记录后全部通过。
分析 “代码看起来支持”不是证据；调用数、shape 与 fallback 必须在 artifact 中闭合。
下一步 / 决策 解锁 Phase 1 输入生成。

Step 38 | Phase 1：冻结新 Conditioning 输入

动作 为新的 calibration/validation/heldout 生成 conditioning、noise 与 sampler identity。
Profiling / 结果 Attempt 001 因 CUDA _CUuuid 无法 JSON serialize，只留下 12 个 partial tensors，禁止复用。修复后两次 publication 的 48/48 tensor pairs bit-identical；whole-file SHA 0/12 相同仅因 safetensors metadata 排序。
分析 Tensor identity 才是 gate；容器 metadata 差异不能被误判成 conditioning 不确定。
下一步 / 决策 解锁 Phase 2 fresh BF16/tensorwise reference。

Step 39 | Phase 2：30 次 Fresh 20 秒 References

动作 生成 8 个 BF16 oracle、8 个 tensorwise warmup、8 个 paired quality、3 个 timing warmup 与 3 个 timing formal。
Profiling / 结果 Attempt 001 的 venv symlink 解析到系统 Python, torch import 失败，0 GPU requests。修正后 30/30 完成。Tensorwise formal generated FPS 为 20.4769 / 20.3788 / 20.3853，mean 20.4137；stable DiT mean 30.7383 FPS。

分析　该 full-request timer 比 Week 3 formal timer 更大，所以 20.41 不代表性能回退。8 个 tensorwise/BF16 pair 的 overall trajectory rel-L2 约 0.3985–0.6374，说明生产 FP8 可用但数值并不接近 BF16。

下一步 / 决策　解锁 Phase 3 metric qualification。

Step 40 | Phase 3：Metric Qualification 没有全部通过

动作　对受控 audio/video corruption 评估候选指标的方向与单调性。

Profiling / 结果　232 manifest records、880 metric rows、约 12.58 GB。MR-STFT/log-mel 对三类 audio corruption 通过；Farneback 对 flicker/seam 只有 0.6667 direction success、Spearman 0.8，共 8 个 blockers。

分析　音频指标可用于局部研究，视频 metric registry 尚不能授权 production/final-heldout 决策。

下一步 / 决策　Phase 3 **BLOCKED**；允许 calibration factorization，但锁住 production metric registry 与 final-heldout。

9.2 Phase 4：Local Factor Grid

Step 41 | Exact K32、Pow2 与 Native E8M0 的因果拆分

动作　先做 20 秒 BF16 capture，再对 320 strata × 8 构成 2,560 paired local samples；分别比较 tensor FP32、K32 exact FP32、K32 ceil-pow2 与 native K32 E8M0。

Profiling / 结果　Capture attempt 01 在完整请求后对超大 tensor 调 torch.quantile 失败，未发布；attempt 02 成功。Tensorwise mean rel-L2 0.0228185；K32 exact 0.0175908，改善 22.910%；K32 pow2/native E8M0 都为 0.0230594，反而比 tensorwise 差 1.056%。

分析　更细 K32 granularity 本身有明确局部收益；简单 amax→ceil-pow2 的 E8M0 scale encoding 抹掉了收益。Native provider 与 pow2 simulator 高度一致，kernel 不是主要误差源。

下一步 / 决策　只让 BF16-reconstruct exact-K32 simulator 进入 paired propagation；native K32 不获得部署资格。

Step 42 | 四条 Paired Block Propagation Lanes

动作　在 blocks 11/23/35/47 对相同 BF16 capture 比较 tensorwise 与 exact-K32 propagated video/audio error。

Profiling / 结果　四条 video block-probe lanes 全部通过；四条 audio p95 lanes

全部失败。Audio p95 分别回退 +6.786%、+6.292%、+5.239%、+6.541%；正式 gate 为 4/8。这里的 4/4 是 block-probe 计数，不是四个 decoded full videos。

分析　Local Linear reconstruction 改善可以在视频传播中保持，却在少量关键 audio forwards 形成 tail。单一 overall mean 会掩盖该问题。

下一步 / 决策　Phase 4 **BLOCKED**，Phase 5 保持锁定；先定位 audio tail。

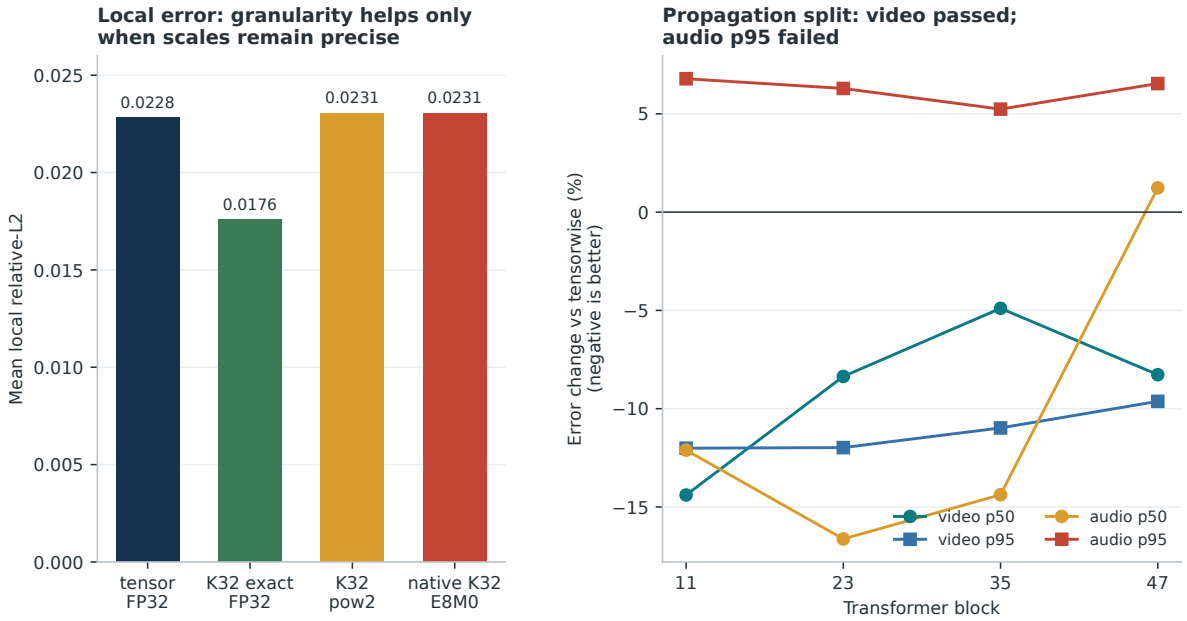

图 9.1: 左：只有 exact FP32 scale 保留 K32 局部收益；右：四条 paired block-probe lanes 中 video p50/p95 改善，audio p95 全部回退。负值表示误差降低；该图不代表 decoded full-video perceptual quality。

Step 43 | Audio Tail Retrospective

动作　按 chunk、forward、denoise stage 和 block 回溯 exact-K32 相对 tensorwise 的 audio delta。

Profiling / 结果　最大一致回退位于 forward 8、chunk 1、denoise_3；warmup audio p50/p95 回退约 72.952% / 20.866%，steady p95 回退约 5.187%，terminal 明显改善。denoise_3 与 cache-commit p95 变差，denoise 0–2 改善。

分析　问题不是“全部 K32 都伤害音频”，而是 early chunk 的 late-denoise/cache-commit 跨模态传播。

下一步 / 决策　尝试静态 audio-path rescue，不再使用高开销 dynamic torch.cond。

Step 44 ｜两个静态 Rescue 仍失败

动作　方案 A：A2V q/out tensorwise，其余 480 modules exact K32。方案 B：480 attention tensorwise，仅 96 video FFN exact K32。

Profiling / 结果　方案 A 仍为 video block-probe 4/4、audio p95 0/4，总计 4/8；方案 B 八组 block/modality gates 全部失败，0/8。

分析　只保护 96 个 A2V I/O 不足以隔离 audio error；只给 video FFN K32 也不能保留传播收益。误差通过更广的 attention/residual graph 耦合。

下一步 / 决策　两者 **REJECT**；下一假设必须覆盖更完整的静态 audio path 或 early-tail callable，并重新预注册数据切分。

到这里真正知道了什么

Scale 更细并非“没有用”：exact-FP32 K32 在 2,547/2,560 local samples 更好，并使四条 video block-probe lanes 全部改善。当时把阻塞归因于 E8M0 representable scale、audio p95 tail 和跨 prompt/seed 泛化，而不是 provider GEMM 正确性；这些 probe 结果不等价于 decoded perceptual quality。[E-W6-LOCAL][E-W6-PAIRED]

本章判决的后续修订：局部结论保留，部署判决被推翻

本章的**局部测量事实全部保留**：exact-FP32 K32 local mean rel-L2 0.0175908 对比 tensorwise 0.0228185（改善 22.910%），K32 ceil-pow2 与 native E8M0 同为 0.0230594（比 tensorwise 差 1.056%）。

但据此作出的两条部署判决已在 2026-07-22/23 被推翻。第一，“native K32 不获得部署资格”的依据是 local rel-L2 更差与 propagation 4/8；后者使用的端到端 trajectory 度量已被证伪，而 native E8M0 K32 现在正是工程基线 V2 的 scale 编码，其同栈配对 stable-DiT median gain 为 14.103773924659912%，两条 lane 的解码媒体面板均为 AUTO_PASS。第二，“E8M0 representable scale、audio p95 tail、跨 prompt/seed 泛化”这三个阻塞项中，前两项都是在被证伪的度量上观测到的，不再是有效的部署判决依据。

这是本报告中“局部指标不能替代解码质量”第一次以**相反方向**生效：一个局部重建更差的 scale 编码，在解码媒体面板上没有可证的伤害。本章唯一在两轮之间保持成立的判断，正是最后那句——local/propagated probe 不等价于 decoded perceptual quality。

第五部分

2026-07-20 后验 Attention 补测

10

2026-07-20 补测：Attention 有效吞吐与占用率辨析

本章发生在 Week 6 之后，是为回答“为什么未另行移植 FA3/FA4，当前 Attention 却看起来利用率很高”而追加的后验补测。结论先行：本章测到的是 FA-style useful-throughput efficiency，不是 SM occupancy；本章测量的对象是 2026-07-20 的状态：所有 attention 都走 PyTorch 自动选择的 fused Flash kernel。它不参与此前 precision candidates 的选择，也不改变 Week 3–6 的先后顺序。

该状态此后已改变。自 Week 7 起，48 个 video-self attn1 已替换为自研 SM120 MXFP8 persistent kernel（详见第 12 章），因此本章的 168.677600 TFLOP/s、80.5144% 与 68.6384% 只描述历史配置，不描述当前 attn1 路径。本章其余关于计时边界与“useful throughput 不是 SM occupancy”的方法论结论仍然成立。

三个百分比口径不能互换

指标	本次结果	它真正回答的问题
NVML GPU-Util	200/200 polls 为 100%	采样窗内 GPU 是否持续有 kernel；不说明 Tensor Core 效率
NCU SM occupancy	未采到	活跃 warp slots 占硬件上限的比例；被 ERR_NVGPUCTRPERM 阻断，当前值未知
FA-style useful throughput	80.51% fixed-spec；68.64% clock-normalized	两个 Attention matmul 的理论 useful FLOPs，相对 complete-op 时间与 dense BF16 capacity 的比例

低 SM occupancy 与高 useful throughput 并不矛盾：occupancy 只数同一时刻可驻留的 warp slots，少量 warp 若能通过流水持续发射 MMA，Tensor Core 仍可能保持较高吞吐。反过来，高 occupancy 也不能证明 Tensor Core 忙。本次没有 NCU counters，因此报告既不猜测当前 occupancy，也不拿 FA-style 百分比替代它。

10.1 第一次回答为什么不成立

Step 45 | NCU Permission Probe 与 NVML Busy 反例

动作 先对 dominant shape 运行 20 秒独立 Attention loop，并同时尝试 basic matmul 与真实 dominant attention 的 NCU counters。

Profiling / 结果 两个 NCU probes 都在 counter collection 前返回 ERR_NVGPUCTRPERM[5]。独立 contiguous-BHSD loop 运行 20.0089 s、15,290 calls、1.30863 ms/call；200 个 NVML polls 都返回 GPU-Util 100%。

分析 NVML 100% 只说明采样窗口内持续有 kernel；它不是 Tensor Core utilization、SM occupancy 或有效 FLOPs。该 loop 的 layout 也与生产 BSHD→BHSD transpose view 不同。

下一步 / 决策 放弃 NVML headline，改为 FA3/FA4 风格的 useful FLOPs / complete-op CUDA-event elapsed，并复现生产 layout。

权限边界

对话层面的授权不能给已创建容器补 Linux capability。当前驱动仍限制 performance counters；因此报告可以给出有效 TFLOP/s，却不能声称已经测得 Tensor/MUFU/SMEM/L2 active 或 compute-vs-memory roofline。[E-ATTN-NCU]

10.2 V1 六 Shape 为什么被降级

Step 46 | V1：六个 steady shapes 各 20 秒

动作 对六个 steady attention shapes 做第一轮 formal timing 与 provider capture。

Profiling / 结果 全部命中 PyTorch Flash，但初版分析错误地把 Kineto kernel-self sum 当作主 latency。审计后发现 FA helper 的核心边界是完整 operation elapsed。

分析 kernel-only duration 可用于归因，却会漏掉 main/combine kernel 之间和同 stream 的 gap；与 useful FLOPs 相除会高估完整算子效率。

下一步 / 决策 V1 superseded，只保留 provider/name 与 secondary kernel diagnosis；预注册 V2 的 25-shape complete-op 口径。

10.3 V2：25 个 Production Shapes 的正式结果

Step 47 ｜运行前冻结源码、Shape Inventory 与口径

动作　从 production Nsys SQLite 重建 25 shape rows 与 11 exact chunks；在首个 GPU interval 前快照计划、helper、benchmark、inventory 和 launcher，并写入 pre-run SHA256。

Profiling / 结果　25/25 rows 每个 formal window 至少 20 秒，输入为 contiguous BSHD 经 transpose 得到 non-contiguous BHSD，BF16、non-causal；全部命中 PyTorch Flash，输出 finite。

分析　当前路径并不是“朴素 attention”，而是 PyTorch SDPA 自动 dispatch 的 fused FlashAttention kernel[4]。长 D128 video shapes 很适合持续发射 Tensor Core work。

下一步 / 决策　用 production logical-call count 对 25 个 complete-op timing 精确加权。

对当前 non-causal、$D_q = D_v = D$ 的 forward，报告只把两个矩阵乘计入 useful work：

$$F_{\text{useful}} = 4BHS_qS_kD, \qquad U = \frac{F_{\text{useful}}/t_{\text{complete-op}}}{P_{\text{dense BF16}}}.$$

Softmax、exp、scale、寻址、padding、tail 和 launch gap 不记有效 FLOPs，但都留在 complete-op 分母中。RTX 5090 官方 dense BF16 Tensor + FP32 accumulate 分母取 209.5 TFLOP/s，而不是带结构化稀疏的 419 TFLOP/s[3]。

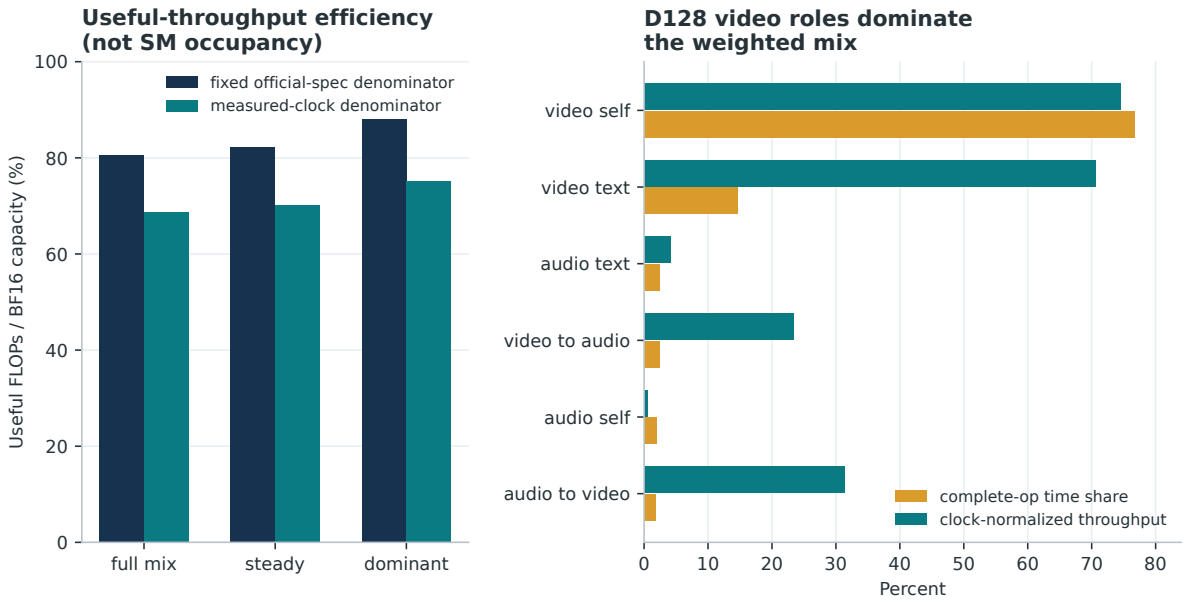

图 10.1: FA-style complete-op useful-throughput efficiency 与 role 构成；它不是 SM occupancy。固定官方规格分母结果与按 formal-window 实测时钟归一化结果必须并列解释。

表 10.1: V2 Attention useful throughput：protocol-primary 与 clock-normalized interpretation

Scope	TFLOP/s	Official peak	Clock-normalized
Full production-call mix	168.677600	80.514368%	68.638389%
Exact steady chunks 8–10	172.223771	82.207051%	70.098352%
Dominant $(1, 32, 2340, 6240, 128)$	184.346137	87.993383%	75.106409%

Step 48 ｜解释 80.51% Official 与 68.64% Clock-Normalized

动作　把分母、硬件、kernel 和 shape mix 分开复核。

Profiling / 结果　full mix event-time-weighted SM clock 为 2823.5 MHz，而 RTX 5090 官方 209.5 TFLOP/s 规格按 2407 MHz 给出；按实测频率缩放后的容量约 245.7 TFLOP/s，所以 168.68/245.7=68.64%。D128 video self/text 占 complete-op time 91.3157%，其 clock-normalized utilization 分别为 74.485% 与 70.620%。四个短 D64 roles 只有约 0.60%–31.37%。

分析　高 aggregate 来自“已经是成熟 PyTorch Flash kernel + 主导 shape 很大 + 5090 的 BF16 峰值分母低于 H100/B200”，不代表当前 kernel 超过 FA3/FA4。即使另一个 kernel 的 NCU occupancy 低于 30%，也不能与本表的 useful-throughput efficiency 直接比较。FA3 论文中的旧 baseline 是 H100 上 FA2 约 35%，不是 FA3 本身 [1]；FA4 的 71% headline 来自 B200 的 BF16 最佳 shape[2]。

下一步 / 决策　公平比较只能在同一 RTX 5090、同一 25-shape matrix、同一完整-op 计时下做 provider A/B。

表 10.2: 论文数字与 Catnip 数字的上下文；本表只防止误读，不构成跨硬件排名

对象	硬件	论文/实验结果	比较边界
FA2 旧 baseline	H100	约 35%	FA3 论文引用的旧实现，不是 FA3 最终值
FA3 FP16 final	H100	最高约 75%	论文最佳 shape；Hopper 专用实现
FA4 BF16 final	B200	最高约 71%	论文最佳 shape；数据中心 Blackwell
Catnip PyTorch Flash	RTX 5090	68.6384%	25-shape production-call weighted、实测时钟归一化；不是最佳 shape headline

10.4 审计修正与下一步

生产 Nsys 的 full kernel-only diagnosis 为 175.364306 TFLOP/s / 83.706113%，只保留为 secondary。旧 steady 84.090119% 将 exact chunks 8–10 的分子除以含异步边界的 wall-clipped PROFILE_STEADY；正确 exact denominator 是 5,760 kernels / 1,149.913473 ms，对应 84.813119%。V2 50-call Kineto self-time 甚至推出不可能的约 301% official peak，因此被明确排除。

Step 49 ｜由数据制定 Attention 优化顺序

动作　将 complete-op role share、dominant launch geometry 与生产 Nsys share 合并。

Profiling / 结果　D128 video roles 占 Attention complete-op 91.3157%，dominant shape 占 64.8969%。Dominant kernel 为 HeadDim=128、BlockM=128、BlockN=64；608 CTAs 在 170 SM 上形成四个简化 wave，最后一 wave 仅 98 CTAs。Attention 占 GPU1 kernel-duration sum 23.0268%。

分析　P0 应覆盖全部六个 D128 production shapes 的 provider A/B，而不是 cherry-pick dominant row；P1 才是 BlockM/BlockN、persistent/work-stealing 与 tail-aware scheduling；P2 将短 D64 calls 批处理或融合。

下一步 / 决策　任何 candidate 至少要让 D128 aggregate elapsed 改善 1%，随后通过三次完整 20 秒请求、trajectory/perceptual gate、determinism、4+6 KV 和 zero-fallback 门禁。

Attention 的端到端收益上限

在 production Nsys 的 kernel-duration 边界上，即使所有 Attention kernel 达到官方 peak，GPU1 kernel sum 也只减少约 3.7520%，即 1.03898×。这不是 E2E wall 预测，但足以说明当前 Attention 优化应严格受收益门槛约束。[E-ATTN-V2]

第六部分

Outlier 统计与自研 Attention Kernel

11

Outlier 统计：把” 离群值伤害 FP8” 从直觉改写成量化结论

11.1 研究设计：一次不跑推理、不改代码的只读统计

这一章与前面每一章都不同：它没有生成任何一帧视频，没有替换任何 kernel，也没有产生任何可部署的候选。2026-07-21 的这次研究只做一件事——把”outlier 会伤害 FP8” 这句在 INT8 量化文献里被反复引用的直觉，放到 baseline27FPS 的 576 个 tensorwise-FP8 Linear 上，用全量权重与 Week 4 已经冻结的 activation observer bundle 做一次可证伪的量化检验。输入是只读的：模型权重文件与 Week 4 的 63,360 条 Streaming activation call records，输出是相关系数、效应量与离线反事实。任何” 提升” 数字在本章都不带部署含义。

Step 45 ｜口径冻结：先说清楚哪些数是精确的、哪些是采样的

动作 冻结统计口径：覆盖 576 modules（48 blocks × 12 families），对 13,690,208,256 个权重元素**全量精确**计算量化误差、amax、underflow 与 clip fraction；分位数与 kurtosis 因内存与排序代价，改用每层最多 262,144 个**确定性等距样本**。activation 侧使用 Week 4 留下的 63,360 条真实 Streaming activation call records（2 cells × 576 modules × 55 forwards）。每个相关系数做 4,000 次 family-stratified bootstrap，seed 20260721。

Profiling / 结果 两个 activation cell 为 person_car_seed12345 与 person_ceramics_seed12345。二者共用同一 seed，因此 prompt 效应与 seed 效应在本章**不可分离**；所有” 跨 prompt” 结论实际是” 跨这两个同 seed cell”。

分析 把精确项与采样项在设计阶段就分开，是为了避免后面出现” 用采样分位数去解释全量误差” 的混淆。全量精确的只有误差能量、amax、underflow 与 clip fraction 四类；tail 形状指标（p99.99、kurtosis）永远带采样误差。

下一步 / 决策 在此口径下先做权重侧相关性检验：tail 越重的层，是不是 FP8 误差越大。

11.2 权重侧：tail 指标预测 underflow，却完全不预测 FP8 误差

Step 46 | Channel amax skew 与 underflow 的强相关

动作　对 576 个 module 计算 weight channel amax skew（输出通道 amax 的偏斜程度），与 production tensorwise-FP8 的 nonzero-to-zero underflow 比例做 Spearman 相关。

Profiling / 结果　rho 0.812298099043934，family-stratified bootstrap 95% CI [0.7732686069245892, 0.8460783511978051]，p 1.6203374254531687e-136，n=576。

分析　这条相关强且稳，但它的含义比字面窄：channel amax 分布越不均匀，单一 tensorwise scale 就越迁就最大通道，小通道的小权重越容易在 E4M3 的次正规区被压成零。它只说明 underflow 的发生机制被 tail 解释了，没有说明 underflow 对总误差有多重要。

下一步 / 决策　直接检验同一个 tail 指标对最终关心的量——FP8 相对误差——是否有预测力。

Step 47 | 同一 tail 指标对 FP8 rel-L2 的相关跨零

动作　用 weight tensor amax/p99.99（尾部厚度的标准代理指标）对 production FP8 relative L2 做同样的 Spearman 相关与 bootstrap。

Profiling / 结果　rho −0.04594150028870141，95% CI [−0.12943355023446151, 0.03745939942104084]，p 0.27099150255913984。CI 跨零，不显著。

分析　这是本章第一个反直觉结果，而且是个**零结果**：在 E4M3 下，” 这一层的 max 比 p99.99 高多少倍” 对该层的重建误差几乎没有解释力。因此” 找出 outlier 最重的层、优先处理它们” 这条在 INT8 世界里天经地义的排序策略，在 Catnip 的 FP8 权重上没有统计依据。

下一步 / 决策　不要停在零结果上——零结果可能只是指标选错。用 INT8 做机制对照，看同一个指标在另一种数值格式下是否恢复预测力。

11.3　INT8 机制对照：为什么 outlier 文献搬不过来

Step 48 ｜同一指标、同一批层、换成 INT8

动作　对同一批 576 个 module、同一个 weight tensor amax/p99.99 指标，改为预测 sampled uniform-INT8 relative L2，并额外扫描 per-layer 最优 clipping alpha。

Profiling / 结果　INT8 侧 rho 0.8173159662656675, 95% CI [0.7842862832416607, 0.846847592520249]，p　1.493268769677772e-139。误差水平：FP8　exact all-weights median rel-L2 2.648264635354275%；INT8 sampled median 4.848858078072195%；INT8 best-clipped sampled median 2.1044623626881553%，对应 best MSE gain median 75.86543298607597%。最优 alpha 分布上，INT8 有 100% 的层最优 alpha < 1，而 FP8 有 0.5746527777777778 的层仍然选 alpha$= 1$（即不裁剪最好）。

分析　同一个 tail 指标，在 INT8 下 rho 0.817、在 FP8 下 rho -0.046，这就把零结果从” 指标无效” 翻转成” 格式差异”。机制是清楚的：E4M3 的 4 位指数把 max outlier 的动态范围直接吸收进指数位，主体值的表示不因为存在一个大 outlier 而被整体挤压；剩下的 3 位尾数在每个 exponent bin 内形成近似恒定的相对舍入误差，于是各层误差水平趋同、与 tail 形状脱钩。INT8 没有指数位，scale 是全层共享的线性因子，一个 outlier 会等比例放大所有元素的绝对量化步长——所以 100% 的层都想裁剪，而 FP8 有超过一半的层裁剪反而更差。

下一步 / 决策　结论上升为本章的核心判断：把 LLM.int8() ／ SmoothQuant ／ AWQ 那一套面向 INT8 的 clipping 与 smoothing 直接搬到 Catnip 的 FP8 上，**在理论层面就不应该 work**。这与 Week 4 Step 32 的实测完全吻合——SmoothQuant 在 calibration 上 median +22.782%，到 heldout 反转为 -2.991%。那次失败不是超参没调好，是机制不匹配。

FP8 与 INT8 的数不能放在一张精度榜上

上面 FP8 的 2.648264635354275% 来自 13,690,208,256 个元素的**全量精确**计算，INT8 的 4.848858078072195% 与 2.1044623626881553% 来自**每层最多 262,144 个等距样本**的估计。两者是不同估计量，只能各自与自身的 clipping 变体比较，**不能**据此宣称”FP8 比 INT8 精度高 X%”。本节唯一有效的跨格式结论是相关结构的差异（rho -0.046 对 0.817）与最优 alpha 分布的差异（0.5746527777777778 对 1.0），这两项都不依赖绝对误差水平的可比性。[E-FP8OUT]

表 11.1: 五条 Spearman 相关（n=576，4,000 次 family-stratified bootstrap，seed 20260721）

指标对	rho	95% CI	显著
weight channel amax skew → production underflow	0.812298099043934	[0.7732686069245892, 0.8460783511978051]	是
weight tensor amax/p99.99 → production FP8 rel-L2	−0.045941500288 70141	[−0.12943355023446151, 0.0374 5939942104084]	否
weight tensor amax/p99.99 → sampled uniform-INT8 rel-L2	0.8173159662656675	[0.7842862832416607, 0.846847592520249]	是
activation tail p95 → sampled tensorwise rel-L2 p95	0.379846334275848	[0.3165110992387452, 0.4462891648301316]	是
activation tail p95 → rowwise MSE gain median	0.44168781761016884	[0.3898034435906903, 0.4933550748389 1233]	是

11.4 权重侧反事实：四条路径，没有一条有肉

既然 tail 不预测 FP8 误差，那么” 针对 outlier 做权重侧改造” 还剩多少空间？下面四条路径全部是离线反事实，直接在权重上算，不涉及 kernel、不涉及推理。

Step 49 | Output-channel FP8：underflow 掉了 93.89%，误差几乎没动

动作　把 weight scale 从 production tensorwise 细化到 output-channel，在全量权重上精确重算误差、underflow、存储与压缩率。

Profiling / 结果　production tensorwise global rel-L2 为 0.02649727306907663（2.6497273069076632%）。output-channel 后 global MSE 改善 0.476%，rel-L2 降到 0.026434085599599103；underflow 从 9.601306661635195e-05 降到 5.869056182104899e-06，减少 93.89%；存储升到 1.000832950367647 bytes/weight，压缩率从 2.0 降到 1.9983354857224855。

分析　这是本章最重要的陷阱，值得写死：**underflow 被几乎消灭了，但 underflow 本来就不重要**。它是一个 count 型损失——被压成零的都是绝对值极小的权重，它们贡献的平方能量太小，主导不了总的 L2 能量。所以一个” 减少 93.89%” 的漂亮数字，只换来 0.476% 的 MSE 改善，还倒贴了压缩率。任何以 underflow 计数为优化目标的方案，都会在这里踩空。

下一步 / 决策　记录 **REJECT**作为” 指标选择错误” 的样本；继续检验裁剪与稀疏残差这两条真正针对 outlier 的路径。

与 Week 4 Step 27 的口径差异

Week 4 Step 27 对同一改动报告的是 weight relative-L2 从 0.0264973 降到 0.0264341、改善 0.2385%，以及 nonzero-to-zero 量化错误降低 95.16%。本章的 0.476% 是 **MSE** 口径（误差平方），0.2385% 是 rel-L2 口径（误差平方根），两者本就相差约一倍关系；93.89% 与 95.16% 则来自不同的 underflow 统计定义。两组数各自自洽，**不可互相替换引用**。[E-FP8OUT]

Step 50 | Dense clip oracle 与 sparse residual：一个没肉，一个只是上界

动作　路径三: 对每一层用全量权重穷举 best dense clipping threshold, 取 oracle 上界（现实中不可得，因为它用到了被量化数据本身）。路径四：sparse residual ——把 outlier fraction 最大的一小撮权重以 exact FP32 residual 加 uint32 index 单独存储，扫描 alpha。

Profiling / 结果　per-layer best dense clip oracle 仅带来 0.1478865618259051% MSE 改善。sparse residual alpha= 0.25：MSE 改善 2.9947691093950124%, rel-L2 0.026097491106692475，压缩率 1.9581202378585871，outlier fraction 0.0026734672194602444。把 alpha 推到 0.0625：MSE 改善 58.61854961069044%, 但压缩率跌到 0.9342419038528882——**比 BF16 还大**。

分析　dense clip 的 oracle 上界只有 0.1478865618259051%，意味着” 任何 clipping 方案” 在 FP8 权重上的天花板都低于千分之二 MSE——这与 Step 48 中 0.5746527777777778 的层最优 alpha 就是 1 完全一致。sparse residual 在 alpha= 0.25 时确实换到 2.9947691093950124% 的 MSE 改善且压缩率只掉到 1.9581202378585871，看上去是四条路径里唯一像样的；但它是**离线反事实上界**: 假设 exact FP32 residual 与 uint32 index 都免费存在, 没有编码实现、没有 GEMM kernel、没有端到端评估。alpha 一旦推到能大幅改善误差的 0.0625，压缩率就跌破 1.0，整个 FP8 的存在理由消失。

下一步 / 决策　四条权重侧路径全部不进入实现队列。把研究预算转向激活侧。

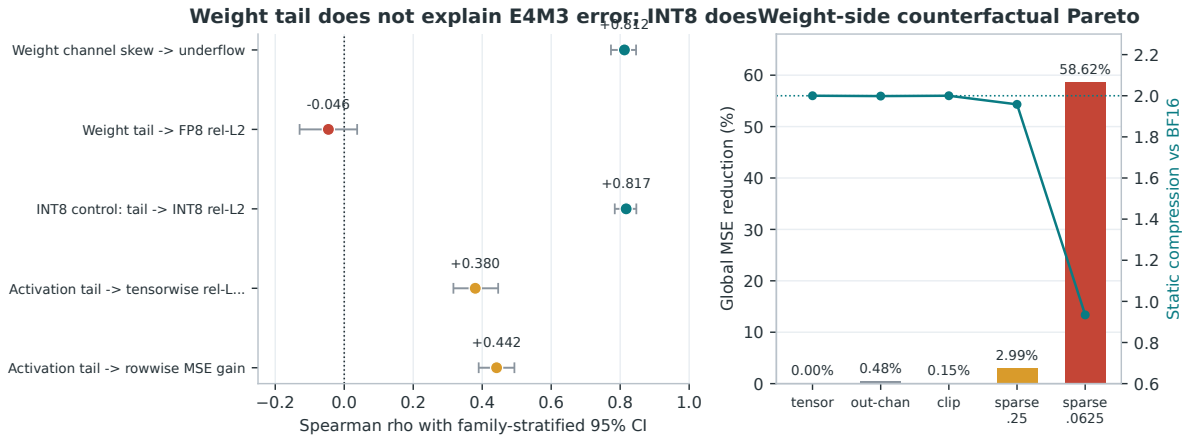

图 11.1: 左：五个 Spearman 相关及其 family-stratified bootstrap 95% CI。weight tail 指标对 E4M3 误差的 CI 跨零、不显著（rho −0.04594150028870141），而同一指标在 INT8 对照下 rho 0.8173159662656675——差异来自格式而非指标。右：权重侧四条反事实的 MSE 改善与静态压缩率 Pareto。sparse residual 曲线是假设 exact FP32 residual 加 uint32 index 的**离线上界**，不是可部署 kernel；alpha= 0.0625 点的压缩率 0.9342419038528882 已比 BF16 更大。全图为权重空间 layer-local 重建量，不代表 decoded media quality。

表 11.2: 权重侧四条离线反事实（全量精确，13,690,208,256 元素；baseline global rel-L2 0.02649727306907663）

反事实	MSE 改善	结果 rel-L2	静态压缩率
production tensorwise（基线）	—	0.02649727306907663	2.0
output-channel FP8	0.476%	0.026434085599599103	1.9983354857224855
per-layer best dense clip（oracle）	0.147886561825 9051%	—	2.0
sparse residual alpha= 0.25	2.994769109395 0124%	0.026097491106692475	1.9581202378585871
sparse residual alpha= 0.0625	58.618549610690 44%	—	0.9342419038528882

11.5 激活侧：效应量大一倍，而且随 prompt 移动

Step 51 | Family 效应量：激活侧远大于权重侧

动作 以 12 个 module family 为分组因子，对四个量分别做 Kruskal–Wallis 检验并换算 ϵ^2 效应量。

Profiling / 结果 激活侧：activation tail $H = 382.0066526981147$、$\epsilon^2 = 0.6578132140037495$；rowwise gain $H = 398.4847855421557$、$\epsilon^2 = 0.6870297615995669$。权重侧：channel skew $\epsilon^2 = 0.3380704097298892$；underflow $\epsilon^2 = 0.47745484541441224$。

分析 ” 这个 module 属于哪个 family” 在激活侧能解释约三分之二的组间变异，在权重侧只解释三分之一到不到一半。加上 activation tail p95 对 sampled tensorwise rel-L2 p95 的 rho 0.379846334275848（CI [0.3165110992387452, 0.4462891648301316]）、对 rowwise MSE gain median 的 rho 0.44168781761016884（CI [0.3898034435906903, 0.49335507483891233]）都显著为正——激活侧的 outlier 结构确实与量化损失挂钩，而权重侧不挂钩。

下一步 / 决策 后续 outlier 相关的研究预算全部投向激活侧；先检验激活的 outlier 结构在换 prompt 时还稳不稳。

Step 52 | 跨 prompt 稳定性：通道结构稳，模块排序不稳

动作 在两个 cell 之间比较三类稳定性：每模块输入通道 log-amax 的 rank 一致性、top-1% 通道集合重合度、exact argmax 通道是否相同；再对 576 个模块的 median error 做 paired Wilcoxon，并计算模块风险排序的跨 cell Spearman。

Profiling / 结果 通道结构三元组：log-amax rank Spearman 中位 0.9128367679598998；top-1% 通道集合 Jaccard 中位 0.5471698113207547；exact argmax 通道相同的模块占 0.625。分布层面：576 个模块的 median error 在

两格之间无整体位移，paired Wilcoxon statistic 81769.0，p 0.7413292858178809。但模块风险排序的跨 cell Spearman 只有 0.4356695853333234。

分析　这两个结论看起来矛盾，其实不冲突：**整体误差分布的中心可以几乎不动（Wilcoxon p 0.7413292858178809），同时" 哪个模块最危险" 这个排序随 prompt 大幅重排（rho 0.4356695853333234）**。前者是一阶矩，后者是秩序结构。工程含义很直接——" 用一个 prompt 找出 top-N 敏感模块、再对这批模块特殊处理" 这条路，在只有两个 cell 的证据下就已经不牢靠；而通道层面的 rank 一致性 0.9128367679598998 说明通道级的结构还算稳定，两者应该分开对待。

下一步 / 决策　任何模块级 top-N 选择策略必须在 ≥ 3 个独立 prompt/seed 上重新预注册；本章不给出这样的策略。

Step 53 ｜联合风险定位：一个只用来挑实验层的描述性 rank score

动作　定义描述性 rank score R，取四个指标（weight amax/p99.99、underflow、activation tail p95、rowwise local MSE gain）各自 rank percentile 的均值，对 576 个模块排序。

Profiling / 结果　最高风险模块集中在 blocks 35–45 的 attn2.to_out.0，并含少量晚层 attn2.to_q。典型例：transformer_blocks.36.attn2.to_out.0，R = 97.96006944444444 percentile，weight amax/p99.99 为 56.615857170282204，underflow 0.111624186452387%，activation tail p95 为 110.12057884925197，rowwise local MSE gain 40.6277317414952%。

分析　R 是四个 rank 的平均，没有任何概率语义——它**不是失败概率**，也没有经过跨 cell 验证（见 Step 52 的 0.4356695853333234）。它唯一的用途是在下一轮需要挑十几个层做深入实验时，提供一个比" 随便选" 更有信息量的起点。晚层 A2V 输出投影排在前面，与 Week 6 Step 43–44 中 audio path 传播问题的定位方向一致，但这只是方向一致，不构成互证。

下一步 / 决策　把 R 的 top 列表作为下一轮激活侧实验的候选池写入 artifact，不赋予任何 gate 权限。

Step 54 ｜实现一致性核对

动作　把本章的全量误差计算路径与 Week 4 一次独立先验审计的逐层结果对齐，比较 error-squared。

Profiling / 结果　最大逐层 error-squared 相对偏差 2.277322806636392e-07。

分析　该量级处于 FP32 累加的数值噪声范围内，说明本章 13,690,208,256 元素的全量计算与既有审计实现在同一条数值路径上，前面所有相关系数与反事实不是由某个新写的量化函数的 bug 造出来的。

下一步 / 决策　统计结论可以引用；转入限制声明。

11.6　五条必须写明的限制

1. **分位数与 kurtosis 是采样量**。误差、amax、underflow、clip fraction 为 13,690,208,256 元素全量精确；p99.99、kurtosis 等 tail 形状指标来自每层最多 262,144 个确定性等距样本。涉及 tail 形状的相关系数都带这层采样误差。

2. **activation 误差 probe 每 call 最多 64 行**。只有 activation amax 是全量精确的；rel-L2 p95、rowwise MSE gain 这类量基于每次 call 至多 64 行的子样本估计。

3. **只有两个 activation cell，且共用同一 seed**。person_car_seed12345 与 person_ceramics_seed12345 的 seed 相同，prompt 效应与 seed 效应不可分离；Step 52 的所有” 跨 prompt” 表述都受此限制。

4. **family-stratified bootstrap 未建模相邻 block 相关性**。重采样按 family 分层，但相邻 block 之间的统计相关没有进入模型，因此所有 95% CI 可能偏窄，显著性判断偏乐观。

5. **权重空间 L2 对 task importance 一视同仁**。本章所有权重侧结论都在 L2 意义上成立，而 L2 不区分” 哪些权重对最终视频/音频更重要”；并且 sparse residual 没有编码实现、没有 GEMM kernel、没有端到端评估，压缩率是静态估算值。

权重侧无肉，激活侧有肉

把”outlier 伤害 FP8” 这句直觉拆开之后，两边的答案是相反的。**权重侧无肉**：tail 指标对 E4M3 误差的相关 rho −0.04594150028870141、CI 跨零；per-layer best dense clip 的 oracle 上界只有 0.1478865618259051% MSE 改善；0.5746527777777778 的层最优 alpha 就是 1；output-channel 把 underflow 削掉 93.89% 却只换来 0.476%, 因为 underflow 是 count 型损失、平方能量不足以主导 L2。机制解释来自 INT8 对照——同一指标在 INT8 下 rho 0.8173159662656675、100% 的层都想裁剪：E4M3 的 4 位指数吸收了 max outlier 对主体值的冲击, 3 位尾数留下近似恒定的相对舍入误差。因此 LLM.int8() ／ SmoothQuant ／ AWQ 那套 INT8 outlier 方法搬到 Catnip FP8 上**理论上就不该 work**, Week 4 Step 32 的 calibration 赢、heldout 反转是机制问题不是调参问题。**激活侧有肉**：family 效应量 ϵ^2 0.6578132140037495 与 0.6870297615995669, 是权重侧 0.3380704097298892 ／ 0.47745484541441224 的近两倍；activation tail p95 对局部误差与 rowwise 收益的相关 rho 0.379846334275848 与 0.44168781761016884 都显著为正。但激活侧的入口条件也更严：通道 rank 稳（中位 0.9128367679598998），模块风险排序不稳（跨 cell 0.4356695853333234）。[E-FP8OUT]

这一章没有任何端到端结论

本章全部数字都是 **layer-local reconstruction** 量：权重空间的 L2 ／ MSE、单层 activation 的 rel-L2 与 rowwise gain、静态压缩率。它们**不是** decoded 视频/音频的感知质量，**不是** generated FPS，也**不是**任何 gate 的通过依据。没有跑过一次推理，没有解码过一帧画面。四条权重侧反事实里唯一看似有收益的 sparse residual（alpha$=0.25$，MSE 改善 2.9947691093950124%）是离线上界，缺编码、缺 kernel、缺端到端评估；rank score R 是描述性统计，不是失败概率。把本章任何一个百分数当作” 精度提升了多少” 引用，都是口径错误。[E-FP8OUT]

12

Week 7：自研 SM120 MXFP8 Attention Kernel

Week 3–6 的全部 attention 都跑在 PyTorch SDPA 自动 dispatch 的 fused Flash kernel 上；2026-07-20 那次补测给出的 168.677600 TFLOP/s（full production-call mix、complete-op 口径）描述的正是那个状态，而不是本章的对象。Week 7 做的事只有一件：把 48 个 video-self attn1 从 PyTorch Flash 换成自研的 SM120 MXFP8 attention kernel，并在换上去之前，先把与它相关的四组精度候选、两组实现变体逐一跑完、逐一记录，包括全部被拒的部分。本章的性能数字与精度数字属于不同口径，任何一处都不能互相替换。

12.1 这个 kernel 到底是什么

数据流水线是固定的八段：BF16 Q/K → signed normalized H128 旋转 → E4M3 + UE8M0 K32 量化 → native SM120 block-scaled QK MMA → one-pass online softmax（reverse-N128 合并）→ P 在线动态 K32 量化 → native SM120 block-scaled PV MMA（FP32 累加）→ BF16 输出。整条链上没有把中间态落回 BF16 再重新量化的环节，因此后文的数值误差必须按“所选量化语义”而不是“实现失配”来解读。

技术栈由四层拼成，血统必须写清楚，否则无法解释后面哪些结论可以外推、哪些不能：

1. **QK/PV 矩阵乘核心**特化自 CUTLASS 4.5.2 官方 SM120 blockscaled GEMM example：mx_float8_t<e4m3>、OpClassBlockScaledTensorOp、ThreadBlockShape $128 \times 128 \times 128$、ClusterShape $1 \times 1 \times 1$、warp-specialized persistent + cluster launch control。
2. **attention 主体**是 FA3 血统的 warp-specialized persistent 核 compute_attn_ws。
3. **producer 层**是手写 CUDA：Q/K 的 H128 旋转 + 量化 + SF swizzle，以及 V 的序列轴 K32 量化 + 转置。
4. **集成层**是 pybind 扩展加 catnip_mxfp8::attention_bsh custom op；该 op 带 fake impl，使 torch.compile 保留一次不透明外部调用而不是尝试内联；替换 48 个 attn1 时是 fail-closed 的严格替换。

CUTLASS 源码注释明确指出 MXFP8 MMA 吞吐是 Ada FP8 MMA 的 2 倍。这条微架构事实不只解释本章 attention core 的加速比，也解释了后来 Linear 侧 K32 收益的来源——两者指向的是同一条 block-scaled MMA 通路，而不是两个独立的偶然优化。

GeForce 版 SM120 有两个必须提前接受的限制：不支持 TMA multicast；cluster shape 必须是 $1 \times 1 \times 1$。这意味着数据中心 Blackwell 上常见的多 CTA 协同取数策略在这块卡上不可用，producer 的搬运成本只能靠自己摊薄。

表 12.1: 主形状（S_q=2340、S_k=6240）生产路径的 launch 配置；数值来自 Nsight 采集，不是源码推断

Kernel	Grid	Block	其他
Fused attention core compute_attn_ws	[170,1,1]	[384,1,1]	dynamic shared 84992 B；168 registers/thread
H128 K-side producer	[25088,1,1]	[256,1,1]	28 registers/thread
H128 Q-side producer	[9728,1,1]	[256,1,1]	同一手写 producer，Q 侧规模
V producer（序列轴 K32 量化 + 转置）	[195,32,1]	[256,1,1]	44 registers/thread

12.2 正交旋转与概率量化：四组候选全部被拒

Step 52 | A3 Signed H128 旋转的模型内 calibration 消融

动作　设三条 lane：A0 生产 BF16 attention；A2 为 K32 E4M3 Q/K/V 加 fixed-scale E4M3 概率路径；A3 在 A2 之上叠加冻结的 role-specific signed normalized H128（post-RoPE Q/K）。A2/A3 使用 query-tiled 数值模拟器，query_tile=128。

Profiling / 结果　P1b 模型内 calibration 中，A3 在每一个 pooled 类别上都优于 A2：overall 改善 2.1063%（48.2238% → 47.2081%）、video input 2.0341%、video velocity 2.0385%、video latent 2.0347%、audio input 8.1237%、audio velocity 8.8755%、audio latent 7.2014%。解码媒体方向同样为正，A3 在四项视频指标与三项音频指标上全部胜过 A2。

分析　预注册门槛是 overall/input/latent ≥ 5%、video velocity ≥ 10%。A3 一项都没达到，因此失败原因是幅度不够，不是回归——方向、类别一致性、解码媒体三者都支持“旋转有效”，只是有效得不够多。

下一步 / 决策　A3 **REJECT**。必须说明这拒绝的是**这一组特定基**与 K32 fake-math

语义下的收益幅度，不构成“所有正交旋转无效”的证据。[E-W7-SM120]

Step 53 | A3 的独立 20 秒 Validation：改善是真的，含义要限定

动作　在 P1a 上做独立 20 秒 validation：离线 sampled attention replay，使用与 calibration disjoint 的 tailor_workshop，seed 31415。

Profiling / 结果　output.full.rel_l2 从 A2 的 2.9455% 降到 A3 的 1.9488%，相对改善 33.8397%；logits 改善 57.0710%；softmax 分布的 JS 改善 75.7148%。

分析　这三个数字的含义只有一句：在固定输入下，A3 比 A2 更接近 BF16 attention 参考。它**不等于**画质改善 33.84%——replay 是离线的、输入是固定采样的，误差没有经过 55 forwards 的完整传播，也没有经过 VAE 解码与感知评估。

下一步 / 决策　保留为“旋转方向正确”的独立证据，但不能作为 Step 52 门槛的补救，也不写进任何画质结论。[E-W7-SM120]

Step 54 | A3 Full Model-Loop FPS Recheck：一组只能当诊断读的数字

动作　2026-07-22 对 A0/A2/A3 三条 lane 做 full model-loop FPS recheck。

Profiling / 结果　A3 为 4.447527234 generated FPS（109948.736515 ms）；A2 为 4.806256391 generated FPS（101742.387466 ms）；A3 比 A2 慢 7.463796%。A0 BF16 in-study control 为 13.682666488 generated FPS。

分析　这三个 FPS 全部来自 fake-math accuracy provider：unfused PyTorch FP32 signed H128 FWHT、fake-quant/dequant、query-tiled FP32 数学；模型里**没有装任何 native SM120 FP8 kernel**。该冻结精度协议还禁用了 torch.compile、并使用 serial video decode。因此 4.45 与 13.68 之间的差距衡量的是模拟器开销，不是量化代价。

下一步 / 决策　记录为诊断值，绝不与生产 FPS 横比；任何把 4.447527234 当作“FP8 attention 性能”的引用都是口径错误。[E-W7-SM120]

Step 55 | A7 概率 K32：下溢大降，误差不动

动作　在 P1c 上把概率矩阵 P 的量化换成 K32（A7），与 A3 对比；同时用 D0 测 exact-P ceiling。

Profiling / 结果　pooled 非零下溢率从 20.0809% 降到 0.9589%，reduction 95.2250%。但 A7 相对 A3 的 full-vs-production defect 只改善 0.0267%（门槛 $\geq 10\%$）、P-quant defect 改善 0.0089%（门槛 $\geq 20\%$）、只捕获 exact-P ceiling 的 0.1700%（门槛 $\geq 60\%$）。25 项 accuracy gate 中 11 项失败。D0 测得的 exact-P ceiling 本身为 full-vs-production +15.7165%。

分析　天花板是存在的（15.7165%），但 A7 几乎一点都没吃到。这是本报告**第二**

次出现同一个结构：下溢统计大幅改善、端到端误差几乎不动——与 Week 6 权重侧 K32 的结论同构。两次都说明“下溢率”不是误差的因，它只是一个容易改善、也容易被误当成收益的中间统计量。

下一步 / 决策　A7 **REJECT**。今后任何以“减少下溢/减少截断”为立论的候选，必须先给出它对 defect 的贡献上限，再申请实验预算。[E-W7-SM120]

Step 56 | P1d Learned Orthogonal Rotation：三种参数化全部被拒

动作　在 P1d 上以 A3 为基线，对三种 learned orthogonal rotation 做 odd-chunk screen：dense_qk_upper、foldable_vo、joint_qk_vo。

Profiling / 结果　dense_qk_upper full mean 改善 +0.2460%、p99 回归 −2.1450%；foldable_vo mean +0.3819%、p99 −1.1788%、PV +0.6153%；joint_qk_vo mean +0.6228%、p99 −3.2462%。门槛为 mean ≥ 5%、p99 回归 ≤ 2%、foldable PV ≥ 10%。

分析　三者都在 mean 上远低于门槛，两者还在 p99 上超出允许回归。dense_qk_upper 的 fit loss 下降 7.255%，但迁移到 odd-chunk 后只剩 0.246%，是明显的局部过拟合：优化目标学到的是 calibration chunk 的分布，不是模型的通用结构。

下一步 / 决策　三者全部 **REJECT**。必须标注这是 calibration-only 的离线筛选，**不是**独立 validation，因此它只能否决候选、不能授予任何候选部署资格。[E-W7-SM120]

12.3　两个实现变体：数值全对，速度全输

Step 57 | Fused Cooperative H128：目标全达成，生产 shape 全回归

动作　把 Q/K 的 H128 producer 融进 attention 主核，用 cooperative launch 消除一次独立 kernel 的固定开销。

Profiling / 结果　功能与数值目标全部达成：5 个 shape 的 BF16 输出与独立 producer 路径逐 bit 一致（bitwise_mismatch=0、max_abs=0）；Nsight 确认 1 次 cudaLaunchCooperativeKernel、1 个 FuseH128=true 的 compute_attn_ws、0 个独立 signed-H128 kernel。但四个真实 Catnip 生产 shape 的 median latency 全部回归：1560×1560 +7.440%、1560×3120 +1.773%、2340×5460 +2.011%、2340×6240 +1.512%。只有合成小 shape（1/1/128/256/128）快 39.148%。整机 opt-in 为 17450.52366 ms / 28.02208171671566 generated FPS，相对相邻 r04 慢 0.221914479%。

分析　合成小 shape 的 39.148% 只说明“消除固定 launch 开销对单 tile 问题有效”，在生产 shape 上这项开销早已被摊薄，融合反而挤占了主核的寄存器与 shared memory 预算。整机那 0.221914479% 不能称为稳定因果回归：每个候选

只有一次相邻整机运行，不是 interleaved 重复 A/B。

下一步 / 决策　作为生产默认 **REJECT**，仅保留为 experimental entry point；生产默认保持“独立 Q/K H128 producer + attention core”。逐 bit 一致这一条被保留下来，它保证以后重新评估时不需要重跑精度。[E-W7-SM120]

Step 58 | FA3-Derived K/V Warp-Release：同样是零数值代价、全负收益

动作　移植 FA3 血统的 K/V warp-release 策略，让消费完的 warp 提前释放资源。

Profiling / 结果　五个 shape 全部 bitwise_mismatch=0、max_abs=0.0；但五个 shape 全部变慢，区间 +0.79% 到 +2.83%；Nsys 独立确认 +2.38% 与 +2.44%。未跑 480p20s 整机。

分析　与 Step 57 同因：SM120 的 cluster 与 TMA 限制下，主核已经处在寄存器压力边缘，提前释放带来的调度收益抵不过重新获取资源的代价。

下一步 / 决策　**REJECT**。因为隔离层面已经全负，按预注册规则不消耗整机 20 秒配额。[E-W7-SM120]

12.4　生产 Kernel 的覆盖合同与数值边界

替换是 fail-closed 的：mask、head count、dtype、device、batch、hidden、sequence-pair 中任何一项不匹配都直接抛错，**不静默回退 SDPA**。audio、video-text、a2v、v2a attention 是有意的非目标路径，不在覆盖范围内，也不因未覆盖而记为失败。

表 12.2: video self-attention 覆盖合同：48 blocks × 55 forwards = 2640 次调用

(S_q, S_k)	调用次数	说明
1560×1560	240	warmup/早期 chunk 形状
1560×3120	240	KV 尚未填满的过渡形状
2340×5460	240	稳定段前一形状
2340×6240	1920	dominant shape，占绝大多数调用
合计	2640	全部由 catnip_mxfp8::attention_bsh 承担

Step 59 ｜数值边界：把 3.7–3.8% 归给量化语义而不是实现

动作　分别对照“独立 quantized-semantics 模拟”与“BF16 cuDNN SDPA”两个参考，检查 fused kernel 的数值位置。

Profiling / 结果　fused vs 独立 quantized-semantics 模拟：rel-L2 0.000359506、cosine 1.0。fused vs BF16 cuDNN SDPA：rel-L2 0.0382514、cosine 0.999269485。LSE 由 ABI 分配但无效，rel-L2 为 1、cosine 为 0。

分析　与自身量化语义的模拟几乎完全重合（cosine 1.0），说明 kernel 实现忠实于设计语义；与 BF16 参考约 3.7–3.8% 的 rel-L2 因此应归因于**所选 online-dynamic 量化语义本身**，而不是实现失配。LSE 明确不作为通过条件——它只是 ABI 里被分配的一块缓冲，当前不产出有效值，任何下游都不得读取。

下一步 / 决策　接受该数值边界进入整机；同时把“LSE 无效”写入 ABI 文档，避免后续误用。[E-W7-SM120]

概率策略变更：A3 的拒绝不能外推到生产 kernel

r04 生产 kernel 使用的是 online_dynamic_one_pass，而 Step 52 被拒的 A3 使用的是 fixed_p256_two_pass。两者不是同一数学策略。这有两个方向的后果：其一，A3 的精度拒绝**不能**直接外推到生产 kernel；其二，也正因如此，生产 kernel 的分布级画质证据**仍然缺失**——没有任何一组已完成的精度实验测的是它实际在跑的那套概率语义。[E-W7-SM120]

12.5　性能：隔离对照与整机结果

Step 60 ｜ Dominant Shape 的隔离对照

动作　在 dominant shape、生产 ABI 下做 isolated per-call CUDA-event benchmark，对照本地 BF16 cuDNN SDPA。

Profiling / 结果　preprocess + fused attention median 0.751248 ms，BF16 cuDNN SDPA median 1.083568 ms，比值 1.44236x；仅 prequantized fused core median 0.551792 ms，core speedup 1.96373x。

分析　1.96373x 与 CUTLASS 注释中“MXFP8 MMA 吞吐为 Ada FP8 MMA 的 2 倍”基本吻合，说明核心确实吃到了 block-scaled MMA 通路。1.44236x 与 1.96373x 的差额就是 producer 侧的旋转、量化、SF swizzle 与 V 转置的成本——在 SM120 没有 TMA multicast 的前提下，这部分无法进一步摊薄到零。

下一步 / 决策　两个比值都必须标注为 isolated per-call CUDA-event benchmark：**不是 full-request，不是 FPS**，不得用于推算端到端收益。[E-W7-SM120]

表 12.3: Dominant shape 隔离对照（isolated per-call CUDA-event median，非 full-request、非 FPS）

路径	median (ms)	比值	口径
本地 BF16 cuDNN SDPA	1.083568	1x	对照基准
preprocess + fused attention	0.751248	1.44236x	含 producer 全部开销
prequantized fused core	0.551792	1.96373x	已量化输入，仅核心

Step 61 ｜整机 r04：28.084266773305128 Generated FPS

动作 在完整 832×480、20 秒合同下跑正式整机运行 r04。

Profiling / 结果 formal wall 17411.884168 ms，generated FPS 28.084266773305128（公式 489000/17411.884168）。该 timer 包含 11 个正式 DiT chunk、GPU1→GPU0 latent copy、11 次 video VAE decode、FIFO ordered drain 与最终双卡同步；排除模型加载、text encoding、权重转换、5 个 compile-warmup chunk、audio decode 与 media mux。相对 O7 三次均值 FPS 高 3.613154337%，wall 低 629.122156 ms。冷启动 warmup 创建 30 个 unique graph、耗时 257.558854 s；整个冷进程 413.836638 s。

分析 28.08 与 O7 之间的 +3.613154337% 是**历史对比**，不是严格因果 speedup：本次运行没有 adjacent provider-off control，两侧的 driver 状态、时钟与进程环境并非同一次会话内交替测得。隔离层面的 1.44236x 之所以只兑现成整机个位数百分比，也与 attention 在 GPU1 kernel 时间中的占比一致，并不矛盾。

下一步 / 决策 r04 作为 Week 7 的交付数字记录，但在报告中一律带“无相邻对照”限定；下一轮必须补 interleaved provider on/off 的重复 A/B。[E-W7-SM120]

Week 7 真正做成了什么

一条完整自研的 SM120 MXFP8 attention 通路已经落到生产默认路径上：CUTLASS 4.5.2 SM120 block-scaled GEMM 特化的 QK/PV 核心、FA3 血统的 compute_attn_ws、手写 producer 与 fail-closed 的 catnip_mxfp8::attention_bsh custom op，严格覆盖 48 blocks × 55 forwards = 2640 次 video self-attention，四组形状零静默回退。实现忠实度有硬证据（对自身量化语义模拟 rel-L2 0.000359506、cosine 1.0），隔离性能有硬证据（dominant shape 1.44236x、core 1.96373x），整机 r04 为 17411.884168 ms / 28.084266773305128 generated FPS。同样重要的是被拒的部分：A3 旋转、A7 概率 K32、三种 learned rotation、fused cooperative H128、FA3 warp-release 共六个候选全部 **REJECT**，其中后两个在逐 bit 一致的

前提下依然被速度否决——这说明 SM120 上真正稀缺的资源是主核的寄存器与 shared memory，不是 kernel 数量。[E-W7-SM120]

两个必须写明的未闭合项

一、**分布级画质证据缺失**。生产 kernel 跑的是 online_dynamic_one_pass，而全部已完成的精度实验（A2/A3/A7/P1d）测的是 fixed_p256_two_pass 或 fake-math 模拟语义。因此 rel-L2 0.0382514 / cosine 0.999269485 只是单点数值距离，报告**不能**据此声称画质无损；多 prompt/seed 的 trajectory、decoded perceptual 与 AV-sync 证据一项都还没有。二、**无相邻对照**。r04 相对 O7 三次均值的 +3.613154337% 是历史对比；fused cooperative H128 的 −0.221914479% 同样只有一次相邻运行。在 interleaved 重复 A/B 完成前，这两个百分比都不构成稳定的因果结论。[E-W7-SM120]

第七部分

判据证伪与历史 K32 工程基线

13

Week 8 上：验收判据的证伪与重建

13.1 起点：15 个联合风险最高的 Linear

第 12 章的 outlier 统计从 576 个 Linear 中挑出 15 个联合风险最高的模块：9 个 attn2.to_out.0 与 6 个 attn2.to_q，全部落在 blocks 32–47。这一节要回答的问题只有一个：**这些层是否真的重要**。为此设计了四条因果 lane。

表 13.1: Week 8 上半程的四条因果 lane

Lane	干预	角色
B	生产 576 层 tensorwise（T/T）	基线，复用 Week 5 references，不重跑
BF15	15 个最高风险层从 FP8 manifest 移除，保持 BF16	kill test，W15+A15 的联合上界
R15	15 个 family 匹配的最低风险层做同样处理	安慰剂对照
A15	15 个高危层改用 rowwise 动态激活 scale（R/T），其余 561 层保持 T/T	只动激活侧的可部署候选

BF15 之所以是最便宜的 kill test，是因为生产 Linear 为 W8A8：权重在加载时静态量化，激活在每次调用时动态量化。把一个模块从 manifest 中移除意味着它**从不被 wrap**，于是权重量化误差与激活量化误差同时消失。因此 BF15 测到的效应是 W15 与 A15 两项干预的联合上界——如果连它都测不出东西，更细粒度的 scale 分配就更没有理由测出东西。

13.2 先修实验台：conditioning identity 的 fail-closed 绑定

Step 60 | Week 4 lane runner 的 conditioning 漏洞与 wrapper 修复

动作 审计 Week 4 的 run_accuracy_lane.py：它对每一条 lane 都重跑一次 Gemma text encoder，且不把结果硬绑定到 reference identity。这正是让 Week 5 第一个候选作废的机制（CONDITIONING_DETERMINISM_INCIDENT）。因此新写 wrapper，装上 Week 5 的 fail-closed 冻结 conditioning bundle。

Profiling / 结果 核验通过：chef 运行记录的 video SHA 8d2049155657fad3...、audio 8f6e0841b47b429b...、tensor file a13550e6bd189625...，与事故文档记录的参考值逐字节一致；fallback_text_encoding_allowed=false。

分析 没有这一步，整批对比会全部无效：lane 之间的差异会混入 text encoder 的非确定性，任何 ± 几个百分点的结论都不可归因。修复的代价是一个 wrapper，作废的代价是整轮实验。

下一步 / 决策 在 identity 校验通过后才允许启动 BF15/R15/A15 四条 lane。[E-W8-PANEL]

13.3 三重证伪：端到端 oracle-L2 测的不是质量

Step 61 | BF15 kill test 与 R15 安慰剂并排跑

动作 在 chef 与 drummer 两个 calibration cell 上跑 BF15 与 R15，报告 oracle-L2 相对生产基线的改善百分比（正值 = 更接近 BF16 oracle）。

Profiling / 结果 见表 13.2。BF15/chef overall -3.5813%，BF15/drummer overall $+11.1544\%$；R15/chef overall $+8.6137\%$，R15/drummer overall -0.2423%。

分析 同一干预在两个 cell 上符号相反，且安慰剂在 chef 上全面优于处理组。

下一步 / 决策 不把这批数字当作候选排名，而是把度量本身送上审判席。

表 13.2: BF15 处理组与 R15 安慰剂的 oracle-L2 改善（%，正 = 更接近 BF16 oracle）

度量	BF15/chef	BF15/drummer	R15/chef	R15/drummer
video_latent	−3.5075	+10.7793	+8.4089	−0.3742
video_velocity	−3.7114	+11.4077	+8.5788	−0.1989
audio_latent	+8.9475	+1.0641	+19.8825	−0.0285
audio_velocity	+9.1224	+1.4602	+20.4246	+0.2675
overall	−3.5813	+11.1544	+8.6137	−0.2423

Step 62 ｜三个致命观察

动作 把表 13.2按“度量是否具备判别力”而不是“候选是否更好”重新读一遍。

Profiling / 结果 (1) 同一干预跨 cell 符号翻转：BF15 在 chef 上 video 变差 −3.5075% / −3.7114%，在 drummer 上大幅变好 +10.7793% / +11.4077%。(2) 安慰剂 R15/chef 在同一度量下**全面优于**处理组 BF15/chef：overall +8.6137% 对 −3.5813%，video 两项 +8.4089% / +8.5788% 对 −3.5075% / −3.7114%，audio 两项 +19.8825% / +20.4246% 对 +8.9475% / +9.1224%。(3) 全部效应落在 ±3%–20% 区间，与 Week 4/5/6 所有候选的历史摆动区间重合。

分析 观察 (2) 是决定性的：若把这张表当真，结论会是“应该去修最安全的层”，这在因果上荒谬，说明该度量在测扰动引起的**轨迹混沌重排**，而不是测误差修复。观察 (1) 说明符号本身不携带信息；观察 (3) 则给出了“chef 过 drummer 挂”的机制——Week 4–6 反复出现的跨 prompt 反转，其量级正是这个噪声带的量级。

下一步 / 决策 端到端 oracle-L2（及建立其上的 overall / primary gate）从验收体系中**REJECT**，降级为“轨迹分岔幅度”的描述性指标。

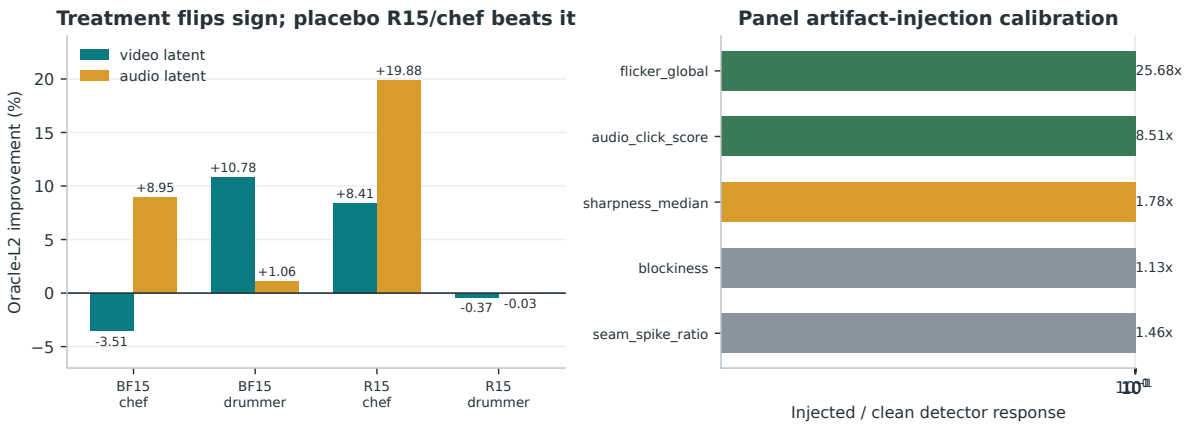

图 13.1: 左：BF15 处理组与 R15 安慰剂在 chef/drummer 两个 cell 上的 oracle-L2 改善百分比；处理组跨 cell 符号翻转，安慰剂在 chef 上全面优于处理组，两者共同证伪该度量的判别力。右：reference-free 质量面板的伪影注入标定倍数（对数轴）；av_sync 未通过标定，不在图中。

Step 63 ｜证伪的边界：局部单-forward oracle-L2 保留

动作 区分被证伪的度量与仍然有效的度量。

Profiling / 结果 被移除的是端到端 55-forward 传播后的 oracle-L2；保留的是局部单-forward oracle-L2——Week 6 中 exact-K32 的 -22.910% 是 320/320 strata 稳定的，属单调区间。

分析 差别在于是否经过 streaming 系统的反馈：单次 forward 内误差与扰动单调相关；一旦误差被写回 KV 与后续 chunk，轨迹分岔幅度就与误差大小脱钩。证伪针对的是传播段，不是量化段。

下一步 / 决策 局部指标继续作为 operator 级研究工具；感知层验收必须另建 reference-free 面板。

13.4 重建：reference-free 质量面板

Step 64 ｜指标与伪影注入标定

动作 对每个候选指标注入受控伪影，检查指标是否按预期方向、以可分辨的幅度响应。

Profiling / 结果 flicker_global $0.3046 \to 7.8236$（25.68 倍）；audio_click_score $0.8645 \to 7.3535$（8.51 倍）；sharpness_median $588.4807 \to 330.7725$ 且 drift 符号翻转；blockiness $1.0277 \to 1.1662$；seam_spike_ratio $3.3325 \to 4.8530$（$+46\%$，仅可作相对比较）；subject_consistency 与 clip_prompt_sim 用 CLIP ViT-B/32 作语义判据。av_sync 代理标定**失败**：400 ms 音频延迟未被检出。

分析 前六项在注入下有明确方向与量级，可作判定指标；av_sync 没有分辨率，留着只会制造虚假安全感。

下一步 / 决策 av_sync 降级为 diagnostic，不参与判定，补齐需引入 SyncNet。其余指标进入面板。[E-W8-PANEL]

表 13.3: 质量面板指标的伪影注入标定结果

指标	基线 → 注入后	幅度	面板角色
flicker_global	0.3046 → 7.8236	25.68 倍	判定
audio_click_score	0.8645 → 7.3535	8.51 倍	判定
sharpness_median	588.4807 → 330.7725	drift 符号翻转	判定
blockiness	1.0277 → 1.1662	绝对量小、方向正确	判定
seam_spike_ratio	3.3325 → 4.8530	+46%	判定（仅相对比较）
subject_consistency、clip_prompt_sim	CLIP ViT-B/32 语义判据	—	判定
av_sync	400 ms 延迟未检出	无响应	diagnostic，需 SyncNet

Step 65 | 标定过程本身修掉三个会造成误判的缺陷

动作 在注入标定不生效时，逐层排查是指标坏了还是注入坏了。

Profiling / 结果 (1) 初版 flicker 指标被自己的全局去均值**盲化**——全局亮度扰动在去均值后归零，指标对最典型的闪烁完全无响应。(2) ffmpeg 的 eq 表达式默认不逐帧求值，注入的时变亮度实际是常量，等于没注入。(3) seam 注入采用丢帧法，导致后续帧索引整体位移，比较的是错位帧对。最终注入改用 PyAV 逐帧精确编辑。

分析 三个缺陷中有两个在注入侧、一个在指标侧，但表现完全相同：指标“没反应”。若不做注入标定而直接上线面板，这三个缺陷都会以“候选没有质量变化”的形式伪装成好消息。

下一步 / 决策 面板的每一项指标必须先通过注入标定才允许参与判定；标定脚本与注入实现一并进入证据包。

Step 66 | 自然差异带与 seam 的基线事实

动作 为每个指标建立判定阈值：取 6 个 cell 上 |BF16 – FP8| 的 p95 作为自然差异带，候选偏离以带宽归一，记为 dev。

Profiling / 结果 带的含义是：两个**都已被接受为生产质量**的 lane 之间本来就存在的差异幅度。$|\text{dev}| \leq 1$ 表示面板没有观察到质量变化的证据。另测得：已接受的 BF16 生产输出在 chunk 边界天然存在中位 3.3 倍、峰值约 10 倍的帧差 spike。

分析 流式 VAE 的边界不连续是基线事实，不是候选引入的伪影，因此 seam_

spike_ratio 只能作相对比较，不能设绝对阈值。用 BF16–FP8 差异定带的代价是带偏宽，好处是阈值有业务含义：落在带内意味着“不比已上线的两条 lane 之间的差异更大”。

下一步 / 决策 面板判定统一用 worst |dev|；超带项必须逐项给出方向与人工抽查结论。

13.5 用新面板重审全部旧候选

Step 67 | 面板重审与 K32 判决反转

动作 把 Week 5–8 的候选输出全部送进 reference-free 面板，重新判定。

Profiling / 结果 见表 13.4。带内：BF15/chef 0.661、BF15/drummer 0.862、R15/chef 0.811、R15/drummer 0.953、K64/chef 0.988、ROW48/drummer 0.521、A15 四格记录到 0.883 / 0.778 / 0.887。出带：K32/chef 1.035、K32/drummer 1.337、K64/drummer 2.042、ROW48/chef 1.367、ROW192/chef 1.709（seam 6.5 对 3.1，本轮最强伪影信号）、tailor 1.287。

分析 两处解读必须修正。其一，K32/drummer 的出带项是 blockiness 1.0449，而 BF16 为 1.0426——两者几乎相同，是 FP8 基线恰好偏低才把 dev 顶出带，**不构成“K32 引入块伪影”的证据**；真正需要人工抽查的是 drummer 的 flicker 0.257 对 0.176。其二，tailor 的 1.287 出带方向向好：出带项是 blockiness -1.287，即块伪影更少。

下一步 / 决策 在感知面板意义下 K32 无确凿质量损失；Week 5 以被证伪的度量拒掉 K32（chef $+0.827\%$、drummer -7.064%）的判决**不再成立**，该 $+14.3\%$ stable-DiT 的吞吐候选恢复活跃。ROW192 维持“全家族 rowwise 不上”。[E-W8-PANEL]

表 13.4: reference-free 面板对旧候选的重审（worst |dev|，≤ 1 为带内）

候选 / cell	worst |dev|	判定	备注
BF15 / chef	0.661	带内	—
BF15 / drummer	0.862	带内	—
R15 / chef	0.811	带内	安慰剂同样带内
R15 / drummer	0.953	带内	—
K32 / chef	1.035	出带	幅度贴带边
K32 / drummer	1.337	出带	出带项 blockiness 1.0449 对 BF16 1.0426，非伪影证据；待抽查 flicker 0.257 对 0.176
K64 / chef	0.988	带内	—
K64 / drummer	2.042	出带	本轮次强信号
ROW48 / chef	1.367	出带	—
ROW48 / drummer	0.521	带内	—
ROW192 / chef	1.709	出带	seam 6.5 对 3.1，本轮最强伪影信号
A15（四格）	0.883 / 0.778 / 0.887	带内	—
tailor	1.287	出带	出带项为 blockiness −1.287，方向向好

13.6 激活解剖：决定后续工具选型

Step 68 ｜ 15 个目标 Linear 的 forward-pre-hook 逐 call 统计

动作 对 15 个目标 Linear 挂 forward-pre-hook，逐 call 记录 per-channel sum/-sumsq/amax 与 per-row amax 分位；共 55 calls × 15 modules，采用 eager 运行，因为 compile 区域内 hook 不触发。

Profiling / 结果 family 中位数见表 13.5。逐层要点：blocks.47.attn2.to_q 的 static_share 96.0%、centering 增益 11.6 倍；全部 6 个 to_q 的 centering 增益 ≥ 5.3、row_spike ≤ 1.18；全部 9 个 to_out.0 的 row_spike 为 3.2–5.5、centering 增益 ≤ 1.5。

分析 两族在两个轴上**干净双分裂**：to_q 的误差来自一个近乎静态的 per-channel 偏置（static_share 77.3%，去均值后 amax 降 5.44 倍），行方向几乎均匀（row amax max/p50 仅 1.11）；to_out.0 反过来，去均值无效（1.14 倍）而行方向 spike 达 4.71。这意味着两族需要两种不同的工具，用同一种粒度去覆盖必然一边浪费一边不够。

下一步 / 决策 进入时间结构核查，确认是否还需要 per-forward-slot 维度的差异化。

表 13.5: 两个 family 的激活统计（family 中位数，55 calls × 15 modules，eager）

Family	层数	static_share	centering 增益	channel amax max/p95	row amax max/p50
S2-Q (attn2.to_q)	6	77.3%	5.44 倍	13.73	1.11
S2-O (attn2.to_out.0)	9	45.2%	1.14 倍	9.78	4.71

Step 69 | 时间结构旁证与工具选型判决

动作 按 forward class 拆分 amax，检查是否存在值得差异化的时间结构。

Profiling / 结果 per-forward-class amax 分化平缓：S2-Q 从 nfe0 的 64.8 到 commit 的 61.5；S2-O 从 nfe0 的 8.1 到 nfe1 的 5.9。nfe0 只比其他 slot 高 8%–30%。

分析 Week 5 的 temporal routing 假设是“不同 timestep 需要不同 scale”，但对这 15 层，slot 间差异远小于 channel 间（13.73）与 row 间（4.71）差异，因此 per-forward-slot 差异化的边际价值有限——这也回过头解释了 Step 36 的五个 temporal 候选为何在 overhead gate 前就不值得付出代价。

下一步 / 决策 按 family 分工具：to_q 六层用动态 mean-centering（每 call 一次均值归约加一个 rank-1 GEMV 补偿，数学精确）；to_out.0 九层用 per-token rowwise（R/T 后端现成，scale 为精确 FP32，不经 E8M0）。

13.7 成本与一个额外的独立证据

Step 70 | 三条干预 lane 的性能与显存成本

动作 在 chef/drummer 上测量各 lane 的 stable-DiT 成本与 GPU1 剩余显存。

Profiling / 结果 BF16-rescue 15 层：stable-DiT −0.91% / −0.97%（chef/drummer），GPU1 剩余 2.65 / 2.61 GiB；R15：−0.94% / −0.95%；A15 四格：−0.00% 到 −0.19%，剩余 2.90–2.98 GiB。

分析 BF15 与 R15 的成本几乎相同（约 −0.9%），这本身是第四个旁证：成本只取决于“有 15 个模块走 BF16”，与是哪 15 个无关；而 A15 只改激活 scale 粒度，成本近乎为零且显存更宽裕。15 层 BF16-rescue 在 dual-5090 的容量口径下可承受，不是被显存否掉的方案。

下一步 / 决策 A15 以近零成本进入后续候选池；BF15 保持为诊断工具而非部署形态。

表 13.6: 三条干预 lane 的成本口径（stable-DiT timer，非完整请求 wall time）

Lane	stable-DiT 成本	GPU1 剩余显存
BF15（15 层 BF16-rescue）	−0.91% / −0.97%（chef/drummer）	2.65 / 2.61 GiB
R15（安慰剂 15 层）	−0.94% / −0.95%	同量级
A15（15 层 R/T）	−0.00% 到 −0.19%（四格）	2.90–2.98 GiB

Step 71 | A15/tailor：“距离 ≠ 质量”的第四个独立证据

动作 检查 A15 在 validation cell tailor 上的 latent 距离与解码后音频统计是否一致。

Profiling / 结果 A15/tailor 在 latent 空间离 oracle 最远：audio_latent −50.780%。但解码后音频统计反而最接近 BF16：audio_rms −23.60 dB，对比 BF16 的 −23.58 与 FP8 的 −25.25；spectral_var 更平稳（0.39 对 0.50）；silence 更低（1.8% 对 2.4%）。

分析 同一候选在两套口径下给出完全相反的排名，且这次是在**解码后**的物理量上给出的——不是又一个代理指标之间的分歧。前三个证据分别来自跨 cell 符号翻转、安慰剂反超、以及 K32 的面板重审；这是第四个、且方法上独立的证据。

下一步 / 决策 latent-space 距离在验收链条中不再具有否决权；A15 继续走面板与人工抽查。[E-W8-PANEL]

方法论结论

本章推翻的不是某个候选，而是 Week 4–6 全部精度判决所依赖的度量。安慰剂 R15/chef 在同一度量下全面优于处理组 BF15/chef（overall $+8.6137\%$ 对 -3.5813%），这条单一事实就足以证明端到端 oracle-L2 测的是扰动引起的轨迹混沌重排，而非误差修复；跨 cell 符号翻转与 $\pm 3\%$–20% 的效应带则解释了此前“chef 过 drummer 挂”的全部机制。因此端到端 oracle-L2 及其上的 overall/primary gate 退出验收体系，降级为描述性的轨迹分岔幅度；局部单-forward oracle-L2 因 exact-K32 的 -22.910% 在 320/320 strata 稳定而保留。取而代之的是经伪影注入标定的 reference-free 质量面板，其直接后果是 K32 判决反转：在感知意义下无确凿质量损失，$+14.3\%$ stable-DiT 的吞吐候选恢复活跃。[E-W8-PANEL]

新面板自身的限制

面板只能证明“没变坏”，不能证明“变好”：$|\text{dev}| \le 1$ 的含义是候选与生产 lane 的差异不超过两条已接受 lane 之间的天然差异，这是一个非劣性判据，不是优越性判据。av_sync 代理未通过 400 ms 延迟的注入标定，已降级为 diagnostic，因此**本轮所有候选都没有音画同步的验收证据**，补齐需引入 SyncNet。$|\text{dev}| \le 1$ 的阈值尚未经人评校准——带宽由 BF16–FP8 的 p95 差异定义，只保证口径自洽，不保证与人眼判断对齐；K32/drummer 的 flicker 0.257 对 0.176 与 K64/drummer 的 2.042 都必须先过人工抽查才能进入部署讨论。此外 seam_spike_ratio 仅可作相对比较：已接受的 BF16 生产输出在 chunk 边界本来就有中位 3.3 倍、峰值约 10 倍的帧差 spike，流式 VAE 的边界不连续是基线事实。[E-W8-PANEL]

14

Week 8 中：历史 landscape K32 MXFP8 基线 V2

历史证据对象

本章的 832×480 K32 baseline V2 是当前 Portrait stack 的前序工程证据。其配对 A/B 结论仍有效，但绝对 FPS 不再是当前 baseline headline；current benchmark 见第 18 章。

14.1 先把“K32”这个词钉死

第 14 章证伪了当初拒绝 K32 的判据之后，Week 8 面对的第一个问题不是“要不要上 K32”，而是“上的到底是哪一个 K32”。Week 5 与 Week 6 里出现过至少四种被同一个名字指代的东西：exact-FP32 K32 simulator、ceil-pow2 K32 simulator、native E8M0 provider、以及保留 BF16 权重的 fake quant。它们的数值行为完全不同，把它们混在一句话里，任何结论都不成立。

Step 68 ｜把候选定义写成可执行门禁，而不是写成描述

动作 把 Week 5 的 native MXFP8 K32 Linear 路径固化为唯一候选定义：576 个 Linear 全部替换为 native K32 MXFP8——E4M3 qdata，沿 GEMM 的 K 维每 32 个元素一个 UE8M0 scale，activation 按每行成组、weight 按每输出行各自成组；单次 aten._scaled_mm 调用、BF16 输出、fused bias。同时把四条禁令写进执行门禁：禁止 split-K、禁止 fake quant、禁止保留 BF16 权重、禁止 fallback 与 upcast。

Profiling / 结果 正式运行的执行门禁观测到 576 次 K32 替换、0 次 fallback、0 次 weight upcast、0 字节保留 BF16 权重，且没有出现新的 formal compile graph。

分析 “576 次替换”与“0 次 fallback”必须同时成立才有意义：只看替换数会把 fallback 混进来，只看 fallback 数无法排除覆盖不足。0 字节保留 BF16 权重是把 fake quant 从定义里剔除的唯一硬证据；无新 formal compile graph 说明这条路径没有偷偷把工作量转移到编译期之外的另一张图上。

下一步 / 决策 以该定义为唯一候选身份进入配对实验；任何不满足四条禁令的变体不得复用本章任何数字。[E-W8-K32]

Step 69 ｜ Attention 侧不动，避免两个变量同时改

动作　attention 侧沿用 Week 7 的 48 个自研 SM120 MXFP8 attn1（H128 fusion disabled），不随本章一起改动。

Profiling / 结果　候选与控制组在 attention 配置上完全一致，两侧差异仅来自 576 个 Linear 的 scale granularity 与 scale encoding。

分析　Week 7 的 attention 路径本身仍有未结的 fusion 议题；若与 Linear 量化同批改动，本章测到的增益将无法归因到任一侧。保持 attention 冻结，是让“配对”这个词真正成立的前提。

下一步 / 决策　把 attention 冻结写入 release manifest 的 runtime 记录，使后续候选无法在不改身份的情况下悄悄换掉它。

14.2　严格配对 A/B：这份报告一直缺的东西

前七周所有的对比，本质上都是“不同批次、不同栈、不同时间”的横向对照。本章第一次给出严格配对设计。

Step 70 ｜ 三对独立 fresh 进程，交替顺序

动作　执行三对独立的 fresh 进程运行，顺序交替为 B/K、K/B、B/K；每条 lane 使用独立且在启动前必须不存在的 compile-cache root。

Profiling / 结果　六个作业全部完成，三对的顺序、进程隔离与 cache root 不存在性由验证器独立复核通过。

分析　交替顺序用于抵消“先跑的那条 lane 承担冷机代价”这类顺序效应；cache root 必须不存在，是为了让每条 lane 都从零开始编译，否则候选可能白嫖控制组留下的 Inductor 产物，把 warm throughput 增益与 cache 复用混为一谈。

下一步 / 决策　在此设计下才允许计算逐对增益。[E-W8-K32]

Step 71 ｜ 两个 FPS 指标的分子分母都不同，不可混用

动作　固定两个指标定义：stable-DiT FPS = 288 / ∑(chunks 5..10 的 dit_wall_seconds)；generated FPS = 489 / (sampling + 流水 video decode) 秒。

Profiling / 结果　stable-DiT 的分子是 288 帧、分母只含六个稳定 chunk 的 DiT wall；generated FPS 的分子是完整合同的 489 帧、分母含 sampling 与流水化 video decode。

分析　两者分子不同(288 对 489)、分母覆盖的阶段也不同(只含 DiT 对含 decode)，因此任何一方的增益都不能声称为另一方的增益，更不能把其中之一称作端到端收益。stable-DiT 是把 warmup 与 decode 剥离后的核心计算指标，generated FPS 才更接近用户可感知的产出速率。

下一步 / 决策　全章数字一律标注所属 timer；门禁阈值分别针对两个指标独立设置。

表 14.1: 三对交替顺序配对运行的 stable-DiT FPS（288 / ∑ chunks 5..10 dit_wall）

配对	执行顺序	tensorwise 控制组	K32 候选	逐对增益
Pair 1	B/K	32.2906767035197	36.84488074452695	14.103773924659912%（中位）
Pair 2	K/B	32.34437926716349	36.87106602064691	13.995280960853007%（最小）
Pair 3	B/K	32.32320669321489	36.88304357580583	高于中位配对
mean	—	32.319420887966025	36.86633011365989	—
CV	—	0.0683391179798529 8%	0.0432257473010205 65%	ratio CV 0.04551887 470930984%

Step 72 | stable-DiT：三对全部同向，最小一对也过门

动作　对三对结果分别计算配对比值，并单独统计 baseline、candidate 与 ratio 三条 CV。

Profiling / 结果　median paired gain 14.103773924659912%，minimum pair gain 13.995280960853007%；tensorwise mean 32.319420887966025（CV 0.0683911797985298%），K32 mean 36.86633011365989（CV 0.043225747301020565%），ratio CV 0.04551887470930984%。

分析　三对同向且最小增益仍接近 14%，说明这不是顺序效应或单次抽样波动。ratio CV（0.0455%）低于 baseline CV（0.0683%），意味着配对相除抵消掉了一部分共同噪声——这正是配对设计相对独立三次运行的价值所在。注意该增益只属于 stable-DiT timer，不含 warmup 与 decode。

下一步 / 决策　进入 generated FPS 复核。

Step 73 | generated FPS：完整 489 帧合同下的增益

动作　在同一批六个作业上，用 489 / (sampling + 流水 video decode) 计算 generated FPS。

Profiling / 结果　tensorwise mean 28.344309608927105 → K32 mean 32.57240180650768，增益 14.916899567908072%；K32 三次分别为 32.552835914832734 / 32.59177997937012 / 32.57258952532018，CV 0.04881249457193308%。

分析　generated FPS 的增益（14.916899567908072%）略高于 stable-DiT 的 median 增益（14.103773924659912%），但两者分子分母都不同，这个差值不构成“decode 也变快了”的证据，只能说明在这条合同下两个 timer 的响应方向一致。K32 侧 generated FPS 的 CV 为 0.04881249457193308%，与 stable-DiT 侧的 0.043225747301020565% 同量级。

下一步 / 决策　把两个 timer 的数字分别写入 release manifest 的性能决策记录，禁止在摘要中合并成单一“加速比”。

表 14.2: 配对实验门禁与判定

门禁	观测值	判定
median paired gain $\geq$ 10%	14.103773924659912%（stable-DiT）	PASS
每对 gain $\geq$ 8%	最小一对 13.995280960853007%	PASS
baseline CV $\leq$ 2%	0.06833911797985298%	PASS
candidate CV $\leq$ 2%	0.043225747301020565%	PASS
ratio CV $\leq$ 2%	0.04551887470930984%	PASS
GPU1 free $\geq$ 536,870,912 B	K32 最小 1,707,868,160 B；tensorwise 2,200,698,880 B	PASS

Step 74 ｜显存：过门但余量确实变窄

动作　记录两条 lane 的 GPU1 最小 free 与 max allocated。

Profiling / 结果　K32 最小 GPU1 free 1,707,868,160 B（1.591 GiB），tensorwise 2,200,698,880 B；K32 GPU1 max allocated 30,504,942,592 B，tensorwise 30,044,127,232 B，差 460,815,360 B。

分析　K32 的 per-32 scale 张量与相应的 quant 中间量把 max allocated 抬高了约 460.8 MB，最小 free 余量从 2.20 GB 降到 1.71 GB。门禁阈值 536,870,912 B 仍有约 3.18 倍余量，但这条余量是在当前 20 秒合同、当前 attention 配置下测到的；更长合同或恢复 H128 fusion 都会重新消耗它。

下一步 / 决策　把 GPU1 free 门禁保留在后续所有配对候选中，不因本次 PASS 而放宽。

14.2.1 必须同等醒目的反向结果：冷启动更慢

Step 75 | K32 的编译代价是真实且可测的回退

动作　在同一批配对作业上统计 compile warmup 与整个 fresh 进程 elapsed。

Profiling / 结果　compile warmup 均值 K32 327.122399 s 对比 tensorwise 265.215584 s，惩罚 61.906815 s（+23.342073%）；整个 fresh 进程 elapsed 481.48748628570075 s 对比 421.0619265236892 s（+14.350753643505222%）。

分析　这不是噪声，也不是可以在摘要里省略的细节：在一次冷启动的完整 20 秒请求里，K32 把 warm 阶段省下的时间又在编译阶段还了回去，整进程 elapsed 反而多出 60.4 s。stable-DiT 的 +14.10% 与整进程 elapsed 的 +14.35% 数值接近纯属巧合，方向相反，绝不可互相抵消叙述。

下一步 / 决策　K32 的定位明确为 warm throughput 候选。本章不支持、也禁止引用任何“K32 冷启动更快”的说法；一次性、短生命周期或频繁重启的部署形态下，K32 是净负收益。

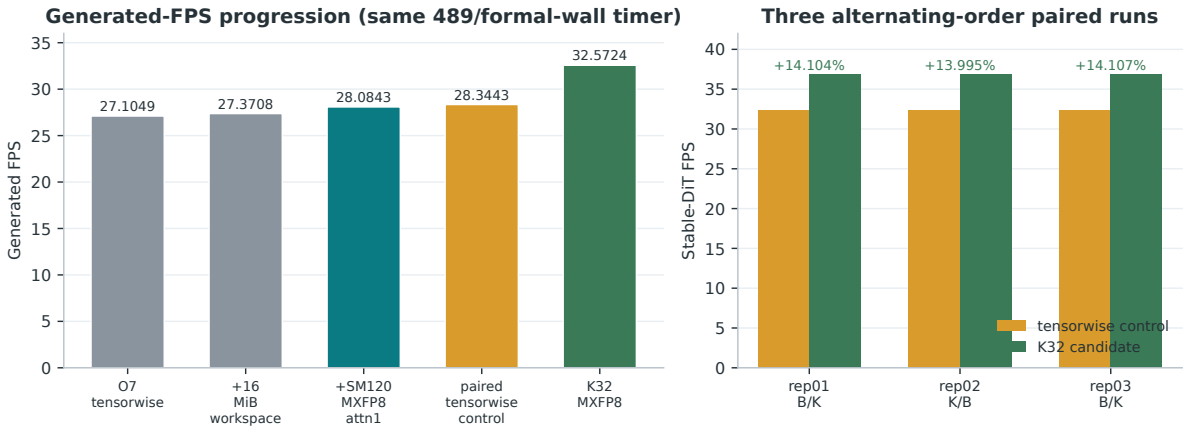

图 14.1: 左：同一 489 帧 / formal-wall timer 下的 generated FPS 演进——O7 27.104924、Week 3 27.370839、Week 7 r04 28.084267、Week 8 同栈 tensorwise 控制 28.344310、K32 32.572402。右：三对交替顺序配对运行的 stable-DiT FPS 与逐对增益。只有右图是严格因果配对（同一进程批次、交替顺序、独立且不存在的 compile-cache root）；左图前四项跨越不同代码栈、不同时间与不同依赖版本，只能读作历史轨迹，不能读作受控 A/B。

14.3 attempt06：六次尝试之后才拿到的证据

Step 76 | 前五次尝试没有产出可引用证据

动作　按时间线复核六次 attempt，只把满足全部执行门禁与双验证器的那一次采纳为最终证据。

Profiling / 结果　attempt06 从不存在的 cache 开始，通过 77 项执行门禁、13 项基础验证器、13 项 portable 验证器；stable-DiT 36.909282 FPS（dit_wall 7,802.915238 ms）、generated 32.609774 FPS（sampling_and_video_decode_wall 14,995.504211 ms）。

分析　attempt06 的两个 FPS 都略高于三对配对实验的对应均值（36.86633011365989 与 32.57240180650768），这是单次运行与三次均值的正常差异，不构成更高的可引用数字；对外引用一律使用配对均值。

下一步 / 决策　attempt06 作为唯一被 manifest 绑定的执行证据；前五次尝试保留在账本中，不得作为性能引用来源。[E-W8-K32]

Step 77 | 冷启动代价的来源：五个 warmup DiT chunk

动作　拆解 attempt06 的 fresh compile warmup 构成与 Inductor cache 增长。

Profiling / 结果　fresh compile warmup 305.712944 s，五个 warmup DiT chunk 分别 53.932 / 60.167 / 66.672 / 63.190 / 52.696 s（合计 296.657 s），内部引擎 elapsed 458.276779 s。cache 从 0 文件、0 字节增长到 1,756 个 Inductor 文件、599,228,407 字节。

分析　296.657 s 占 305.712944 s 的绝大部分，说明编译代价几乎全部落在这五个 warmup chunk 上，而不是分散在运行期。599,228,407 字节的 Inductor 产物是这段时间的物化结果，也解释了为什么“cache root 必须不存在”这一条门禁不可放松——一旦允许复用，冷启动惩罚会凭空消失，配对结论随之失真。

下一步 / 决策　把 cache 文件数与字节数写入证据信封，作为“确实从零编译”的可校验事实。

Step 78 | 媒体合同与执行门禁闭合

动作　核对 attempt06 的输出媒体是否满足冻结的 20 秒推理合同。

Profiling / 结果　输出恰为 489 帧 H.264 832×480、19.56 秒，加 48 kHz 立体声 AAC。执行门禁观测到 576 次 K32 替换、0 次 fallback、0 次 weight upcast、0 字节保留 BF16 权重、无新的 formal compile graph。

分析　帧数与时长必须精确匹配，否则 generated FPS 的分子 489 就不成立——这是把 FPS 从“自称”变成“可核算”的唯一方式。音轨规格同样入合同，因为

Week 6 的 audio p95 tail 说明跨模态误差会绕过视频指标。
下一步 / 决策　媒体合同校验前置于任何性能数字的发布。

Step 79 | 独立 portable 验证器：不信任生成方的自证

动作　用与运行方独立的 portable 验证器重新绑定运行身份。
Profiling / 结果　验证器绑定 40 个关键文件、13 个已加载模块来源、16 个 attention artifact、3 个预构建 CUDA 扩展、6 个 Python 包版本、精确的 bitsandbytes CUDA 12.8 二进制、host runtime ABI、完整模型身份、result-local prompt/contract 以及外层 replay 命令；并要求 pre-run 与 post-run 的 source/ABI/Gemma 快照一致。
分析　pre/post 双快照一致性是防止“运行中途换实现”的关键：只在开始时拍一次快照，无法排除进程内热替换。绑定 13 个已加载模块来源而不是仅绑定磁盘文件，是因为磁盘上存在但未被加载的文件不影响结果，而被加载的旁路实现会。
下一步 / 决策　portable 验证器与基础验证器必须双双通过，缺一即视为无效运行。

Step 80 | Week 3 的 16 MiB cuBLASLt workspace 结束“待部署”状态

动作　把 CUBLASLT_WORKSPACE_SIZE=16384 从 Week 3 的候选状态转为 Week 8 全部正式运行的冻结环境变量之一。
Profiling / 结果　该变量在本章的 6 次 performance 运行与 8 次 calibration 运行中全部生效，共 14 次。
分析　Week 3 把它 **PROMOTE**时留下的尾巴是“部署脚本仍需内置环境检查并重新做三轮 fresh acceptance”；Week 8 的 fresh 进程设计天然满足这一条，且环境变量已进入 release manifest 的执行门禁检查。因此它不再是待部署候选，而是控制组与候选组共享的固定环境——这也意味着它的 -0.865511% wall 收益已被吸收进 tensorwise 控制组基线，不再计入 K32 的 14% 之中。
下一步 / 决策　**PROMOTE**状态关闭；后续任何配对实验的两条 lane 都必须携带该变量，否则视为环境漂移。

14.4 发布身份：内容寻址 manifest，不是 hermetic digest

Step 81 | release manifest 绑定什么、明确不绑定什么

动作 发布内容寻址 release manifest catnip-k32-reusable-v1:sha256:e032427521bb6fe1815c090bfd58915e5a5e34d5c7c77495eb60df83b02d5925，其 ID 是对除 ID 字段本身之外的规范化清单内容取 SHA256。

Profiling / 结果 manifest 绑定固定的 20 秒推理合同、模型身份、40 个选定的 runtime 源/二进制记录、runtime ABI、必需的执行门禁检查、attempt06、三对性能决策与 calibration 自动证据；同时冻结 claim 边界：身份为 reusable_experiment_runner，formal_baseline_accepted=false。

分析 它明确不是 hermetic container digest：不哈希 CUDA driver、不哈希每一个 Torch/cuBLAS/cuDNN 共享对象、不哈希每一个传递性 site-package。这意味着它能保证“同一份清单描述的是同一组被选定的文件与同一套门禁”，但不能保证“在任意机器上按位重现”。把这条边界写进 manifest 本身，比写进文档更难被忽略。

下一步 / 决策 对外引用一律带上 formal_baseline_accepted=false；manifest 可用于复现实验，不可用于宣称正式基线。[E-W8-K32]

14.5 解码媒体质量：自动面板 AUTO_PASS，人工评审仍未开始

Step 82 | calibration 执行与自动判定

动作 沿用第 14 章的 reference-free 面板，对 K32 与 tensorwise 两条 lane 各跑 calibration。

Profiling / 结果 calibration_execution PASS（8/8 运行、8/8 独立验证）；calibration_automatic PASS，两条 lane 均 AUTO_PASS。

分析 执行 PASS 与自动判定 PASS 是两件事：前者只说明 8 次运行都完成且被独立验证，后者才涉及指标阈值。把两者拆成独立状态位，是为了避免“跑完了”被读成“质量过了”。

下一步 / 决策 进入偏差明细核对。

表 14.3: K32 lane 的非零自动有害偏差（以冻结自然带宽归一）

Cell / 指标	说明	归一偏差
drummer_rooftop_seed13579 / flicker_global	本 lane 最大值	1.8357545923880418
violinist_station_seed77777 / blockiness	—	0.6205448576104072
chef / blockiness	—	0.5436534483290965
scientist / blockiness	—	0.1702499922810123
drummer / blockiness	—	0.09754742031431718
drummer / seam_spike_ratio	—	0.08223347851634989
每个 K32 cell / audio_click_score	全部为零	0.0

Step 83 | 为什么 1.8357 仍判 AUTO_PASS

动作　按预注册规则判定：孤立报警阈值为严格大于 2.0，且要求没有任何指标在两个及以上 cell 超过 1.0。

Profiling / 结果　K32 最大自动有害偏差为 1.8357545923880418，来自 drummer_rooftop_seed13579 的 flicker_global：该 cell 中 BF16/FP8 已知良好边缘为 0.17829132080078125，K32 越过该边缘 0.1166534423828125，除以冻结自然带宽 0.06354522705078125 得 1.8357545923880418。同期 tensorwise 控制组最大值为 0.9264946533145333，来自 violinist_station_seed77777 的 blockiness。

分析　1.8357545923880418 未超过 2.0，且表 14.3 中没有任何单一指标在两个及以上 cell 超过 1.0，两条规则同时满足，因此判 AUTO_PASS。但它与阈值只差 0.1642，并且明显高于同期控制组的 0.9264946533145333——这不是“质量与 tensorwise 相当”的证据，只是“未触发预注册的自动报警”。audio_click_score 在每个 K32 cell 都为 0.0，说明 Week 6 定位到的 audio tail 在本次面板的这一指标上没有复现，但该面板本身不覆盖 Week 6 使用的 p95 口径。

下一步 / 决策　记录 flicker_global 为重点观察项，交由人工盲评裁决。

表 14.4: 冻结的自然带归一宽度。band 内部哈希见正文。

指标	归一宽度
flicker_global	0.06354522705078125
seam_spike_ratio	2.0890769623390737
blockiness	0.011582732200622559
audio_click_score	0.09535861015319824

Step 84 | 带宽定义用最大值而不是 p95

动作　固定自然带宽度的定义：四个 calibration cell 上 BF16 与生产 FP8 的**最大**绝对差；validation 与 heldout 数值从不参与。

Profiling / 结果 四个宽度见表 14.4，band 内部哈希 c72cff01e42fb6e490de5c6d95d217fa35b09b3d0ca5176a4f67f31686fddad4。

分析 用最大值而非 p95，是刻意选择更宽的带——带越宽，归一后的偏差越小，判定越宽松。这个方向的保守性对候选有利而非有害，因此不能用“阈值定得松”来解释 AUTO_PASS 之外的任何结论；反过来说，若改用 p95，1.8357545923880418 会进一步放大。validation 与 heldout 不参与带宽构造，是为了防止用待评估数据定义评估尺度。

下一步 / 决策 带宽与哈希一并冻结进 manifest，任何后续候选不得重算带宽。

Step 85 | 承认并修正 heldout 泄漏

动作 复核 Week 8 第一版 reference-free 面板的构造过程。

Profiling / 结果 第一版在构造自然带时使用了原本作为 final-heldout 的 scientist-greenhouse/seed42424 与 violinist-station/seed77777 两格，构成泄漏。

分析 一旦这两格参与了带宽定义，它们就永久失去 heldout 资格——事后声明“只用了一点点”无法恢复其独立性。这是本章唯一一处协议层面的错误，且它直接影响本章引用最多的两个数字（violinist blockiness 与 scientist blockiness 都来自这两格）。

下一步 / 决策 V2 协议记录该泄漏，把两格重分类为 calibration（calibration 现共四格），并另行保留两个不透明的 final-heldout-v2 占位符，需 hash-bound 书面解锁才能访问。本章所有 calibration 结论因此只在四格 calibration 上成立，不构成任何 heldout 泛化证据。

AUTO_PASS 的覆盖面

calibration_automatic 的 AUTO_PASS 只覆盖四个自动伪影指标：flicker_global、seam_spike_ratio、blockiness、audio_click_score。manual_semantic_review_required 仍为 **true**。语义漂移、主体一致性、prompt 遵从度、音画同步等均不在这四个指标的能力范围内，AUTO_PASS 对它们不作任何陈述。

14.6 锁定链：为什么现在还不能叫基线

表 14.5: Week 8 结束时的质量锁定链

阶段	状态	依据 / 解锁条件
calibration_execution	PASS	8/8 运行、8/8 独立验证
calibration_automatic	PASS	两条 lane 均 AUTO_PASS
calibration_manual	PENDING	需两名独立评审；公开包已就绪、handoff 文档与响应验证器已就绪；formal_adjudication_exists=false
calibration_freeze	**BLOCKED**	marker status/quality_calibration.freeze.json 不存在
validation	LOCKED	required 4 runs、completed 0；解锁需 freeze marker
heldout_v2	EMBARGOED	需 hash-bound 书面解锁
overall_status	CANDIDATE_PENDING_MANUAL_CALIBRATION	

Step 86 | 唯一的阻塞点是人，不是数据

动作 核对锁定链上每一环的解锁前置条件。

Profiling / 结果 calibration_manual PENDING → calibration_freeze **BLOCKED**（marker 文件不存在）→ validation LOCKED（required 4、completed 0）→ heldout_v2 EMBARGOED；overall_status = CANDIDATE_PENDING_MANUAL_CALIBRATION。

分析 链上后三环全部是机械依赖：freeze marker 由人工评审裁决产生，validation 由 freeze marker 解锁，heldout_v2 由书面解锁。因此本章无法推进的唯一原因是两名独立评审尚未完成盲评，而不是缺数据、缺算力或缺验证器——公开包、handoff 文档与响应验证器都已就绪。

下一步 / 决策 K32 保持候选身份；在 formal_adjudication_exists 转为 true 之前，不产出 freeze marker，不启动 4 次 validation 运行。

14.7 把这套方法变成可复用的 harness

Step 87 | 后续候选不得就地替换 K32 release

动作 把本章的配对流程固化为 scripts/run_paired_candidate.py：绑定精确的 K32 release 作为控制组，按 control/candidate、candidate/control、control/candidate 顺序执行三对隔离的 20 秒运行，每条 lane 有独立且不存在的 compile-cache root。

Profiling / 结果 候选侧由 write_pair_lane_evidence.py 写 COMPLETE 信封——该 writer 从不宣布候选 PASS；六个作业结束后由 validate_paired_candidate.py 独立重放三个控制组 baseline reference、从原始 chunk timing 重算 stable-DiT FPS、检查媒体合同，并拒绝身份 / 顺序 / 环境 / cache / artifact 漂移。

分析 把“写证据”与“判定 PASS”拆到两个程序里，是为了消除自证：writer 只能陈述发生了什么，判定权在独立的 validator。validator 从原始 chunk timing 重算 FPS 而不是读取 writer 报告的 FPS，使得 writer 侧的计算错误或篡改无法传递到结论。后续任何 kernel / 量化 / attention / compile / 调度想法都应以此方式与固定的 K32 release 对照，而不是把 K32 release 改掉再自比。

下一步 / 决策 harness 与 K32 release 一同发布。**配对验证 PASS 只是描述性性能证据；解码媒体 calibration 与 validation 仍是质量权威**，任何候选不得以配对 PASS 绕过质量链。[E-W8-K32]

第一次拿到严格配对 A/B

在此之前，这份报告里所有的“更快”都建立在不同批次、不同代码栈的横向对照上；Week 8 第一次用三对交替顺序、独立 fresh 进程、独立且不存在的 compile-cache root 的设计，把 K32 与同栈 tensorwise 控制组放进同一批实验。stable-DiT median paired gain 14.103773924659912%、最小一对 13.995280960853007%、ratio CV 0.04551887470930984%；generated FPS 由 28.344309608927105 升至 32.57240180650768，增益 14.916899567908072%。六项门禁全部 PASS，attempt06 通过 77 项执行门禁与两套各 13 项验证器，并以内容寻址 manifest catnip-k32-reusable-v1:sha256:e032427521bb...5925 固定身份。Week 3 的 16 MiB cuBLASLt workspace 在此一并结束候选状态，成为两条 lane 共享的冻结环境。[E-W8-K32]

这一章不能被读成什么

- **formal_baseline_accepted=false**。manifest 的身份是 reusable_experiment_runner，overall_status 为 CANDIDATE_PENDING_MANUAL_CALIBRATION。K32 不是已接受的正式基线。
- **冷启动更慢，且幅度可观**。compile warmup 327.122399 s 对比 265.215584 s（惩罚 61.906815 s，+23.342073%），整个 fresh 进程 elapsed 481.48748628570075 s 对比 421.0619265236892 s（+14.350753643505222%）。K32 只是 warm throughput 候选。
- **两个 FPS 指标不可混用**。stable-DiT 的分子是 288、分母只含 chunks 5..10 的 dit_wall；generated FPS 的分子是 489、分母含 sampling 与流水 video decode。两者的增益不能互相引用，也都不是端到端部署收益。
- **AUTO_PASS 只覆盖四个自动伪影指标**，manual_semantic_review_required 仍为 true；K32 最大归一有害偏差 1.8357545923880418 距报警阈

值 2.0 只差 0.1642，且远高于同期控制组的 0.9264946533145333。

- **calibration 结论只在四格 calibration 上成立**。第一版面板曾把两个 final-heldout cell 用于构造自然带，构成泄漏；V2 已重分类并另设两个 hash-bound 的 final-heldout-v2 占位符，本章不含任何 heldout 泛化证据。
- **manifest 不是 hermetic container digest**——不哈希 CUDA driver、不哈希每个 Torch/cuBLAS/cuDNN 共享对象、不哈希每个传递性 site-package。
- **唯一的阻塞是盲评**。calibration_manual PENDING（formal_adjudication_exists=false）→ calibration_freeze **BLOCKED**→ validation LOCKED（required 4、completed 0）→ heldout_v2 EMBARGOED。

15

Week 8 下：历史 landscape K32 的设备分离 Profiling

可迁移的 profiling evidence，不是 Portrait benchmark

本章对 832×480 历史 K32 stack 的 attribution 用于排序后续优化面。它不更新当前 Portrait 的绝对 FPS；Week 9 已用其中的 V-layout hypothesis 完成独立 A/B，current baseline 见第 18 章。

15.1 测量对象、插桩方式与扰动口径

本章的全部测量都指向同一个冻结对象：release catnip-k32-reusable-v1:sha256:e0324275... 的 K32 baseline。本轮不修改任何 baseline runtime，只做观测；所有实现改动一律推迟到下一轮，以保证归因与既有速度口径可比。[E-W8-PROF]

Step 76 ｜绕过损坏的 NVTX trigger，仍然跑完整 20 秒请求

动作 对冻结 K32 baseline 采一次不改实现的 20 秒完整请求 Nsys trace。

Profiling / 结果 该宿主机的 Nsys 2025.3 NVTX trigger 损坏，无法按 range 启动采集；launcher 改用已验证的 --start-later 会话，在三个 compile warmup chunk 之后开始 trace，离线报告再裁剪回既有的 PROFILE_REQUEST NVTX range。应用侧仍跑完整的冻结 20 秒请求，并产出正常推理门禁。

分析 工具缺陷被隔离在采集侧而不是应用侧：被裁掉的是 warmup 段，不是被测请求本身；formal 口径仍然是 PROFILE_REQUEST 这一个 NVTX range，与既有章节的请求口径对齐。

下一步 / 决策 允许以该 trace 做设备分离归因，但速度对外口径不采用其中任何 wall。

Step 77 ｜先量化插桩扰动，再决定哪个口径可以对外

动作 对未插桩三次运行取均值，与同配置的 Nsys 插桩运行逐项对照。

Profiling / 结果　未插桩三次均值 stable-DiT 36.866330 FPS、generated 32.572402 FPS；Nsys 插桩运行为 36.301144 / 31.907889，即分别引入约 1.533% 与 2.040% 的扰动。sampling+video decode wall 从 15,012.713 ms 增至 15,325.364 ms，长 2.083%。

分析　2% 量级的扰动足以淹没 promotion gate 所关心的差异，但不足以破坏 kernel 之间的相对占比，因此 trace 可用于结构归因、不可用于速度声明。

下一步 / 决策　速度对外口径继续使用未插桩基线 36.866330 / 32.572402 FPS；本章所有百分比均标注是插桩 trace 内的相对占比。

口径纪律

本章出现三种互不等价的时间：未插桩 wall（唯一可对外的速度）、Nsys 插桩下的 formal PROFILE_REQUEST wall 15,325.152 ms（归因分母）、以及隔离 CUDA-event benchmark 的投影（既不是 wall 也不能相加）。三者不得互换，也不得混合成同一张收益表。[E-W8-PROF]

15.2　设备分离的 Nsys SQLite 归因

Step 78 ｜先让 self-check 全绿，再看任何百分比

动作　从原始 SQLite 只读复算，按设备分离统计，并对齐 stage interval、attention count 与 sequence-pair 三类门禁。

Profiling / 结果　最终 self-check：45 个分析窗口、12 个 stage interval gate、3 个 attention count gate、5 个 sequence-pair gate 全部 PASS，无 failure。GPU1 共 369,801 次 kernel launch，kernel sum = union = 13,469.349 ms，stream-overlap factor 1.0000；formal PROFILE_REQUEST NVTX wall 15,325.152 ms。

分析　stream-overlap factor 恰为 1.0000 说明 GPU1 上没有可观测的 kernel 并发重叠，kernel sum 可以当作该卡的串行占用；但它仍然只是 GPU1 的占用，不是请求 wall。

下一步 / 决策　以 GPU1 kernel sum 作为卡内份额分母，以 formal wall 作为请求份额分母，两个分母全程分别标注。

表 15.1: GPU1 稳态 chunk 5–10 envelope 的 kernel taxonomy（互斥 name-based 分类，占 GPU1 kernel sum）

类别	占 GPU1 kernel sum
K32 scaled-mm GEMM	38.489%
BF16 GEMM/GEMV	15.673%
Layout/cat/copy fused	11.071%
Custom attention fused core	10.906%
PyTorch SDPA/Flash	7.406%
Reduction/norm	6.711%
Elementwise	3.745%
K32 scaled-mm support	2.477%
Custom attention V K32 quant	1.966%
Custom attention H128 Q/K quant	1.556%

Step 79 | scaled-mm 与 custom attention 的正式份额，以及它只是下界

动作　把稳态 envelope 的分类结果扩展到 formal 请求口径，并对三类显式命名的 custom-attention kernel 单独合计。

Profiling / 结果　formal 口径下 K32 scaled-mm GEMM 为 31,680 launches / 5,284.153 ms，占 kernel sum 39.231%、占请求 wall 34.480%；scaled-mm support 另占 2.500%（336.677 ms、55,440 launches），两者合计 41.731%。三类显式命名的 custom-attention kernel 合计 1,810.616 ms，为 GPU1 kernel sum 的 13.442%、formal wall 的 11.815%（稳态口径为 1,140.626 ms / 14.428%）。

分析　这个 13.442% 是 name-based 部分量，也就是下界：它既不是完整的 provider e2e，也不是模型里全部 attention。provider 周边的 cast 与 layout 很可能被归入通用 Layout/cat/copy fused 或 Elementwise 类别，因此真实 attention 面更大而不是更小。

下一步 / 决策　凡引用 attention 份额，一律写成" 三类命名 kernel 的下界"；完整 provider 边界成本改由 Step 84 的隔离 benchmark 单独测。

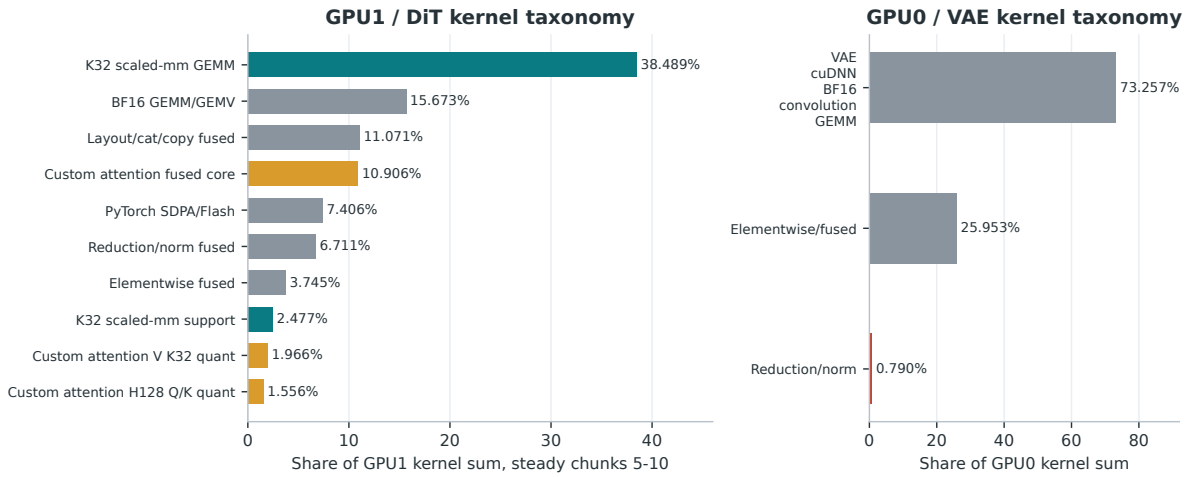

图 15.1: 左：GPU1 稳态 kernel taxonomy，K32 scaled-mm GEMM 与 support 合计 40.966%，三类命名 custom-attention kernel 合计 14.428% 且只是 name-based 下界；右：GPU0 taxonomy，VAE cuDNN BF16 convolution GEMM 占 73.257%。两卡 kernel sum 分属不同设备，绝不可相加成端到端 wall。

Step 80 ｜单个 NFE 的稳定性，以及 NFE 0 不能混入同一分布

动作　下钻到 chunk 10 的逐个 NFE，检查 scaled-mm 份额是否稳定。

Profiling / 结果　chunk 10 的 NFE 2 中 K32 scaled-mm 占 38.772%；NFE 1–3 都约 271.5 ms，其中 scaled-mm 约 102–103 ms。NFE 0 明显更短（230.178 ms），对应不同的 sigma 与 cache state。

分析　NFE 1–3 的重复性说明稳态份额不是被少数长尾 NFE 拉出来的平均值；NFE 0 是结构不同的一步，把它并入同一分布会同时污染均值与方差。

下一步 / 决策　后续所有 per-NFE A/B 只在 NFE 1–3 上比较，NFE 0 单独列出。

Step 81 ｜三个 P0 scaled-mm shape 与正式 symbol 身份

动作　对稳态 scaled-mm 做 shape 审计，并用 Nsys 结构信息核对 symbol、device、grid、block、shared memory。

Profiling / 结果　三个 P0 shape 合计占稳态 scaled-mm 87.665%：$2340 \times 4096 \times 4096$ 占 36.949%（grid [19,32,1]）、$2340 \times 16384 \times 4096$ 占 26.256%（grid [19,32,1]）、$2340 \times 4096 \times 16384$ 占 24.460%（grid [19,128,1]）。三者 block 均为 [384,1,1]、static shared 0 B、dynamic shared 81,920 B，并与 full trace 在 demangled symbol、device、grid、block、shared 上完全一致（Nsys 等价门禁 3/3 PASS）。scaled-mm shape 审计 122/122 PASS。

分析　前两个 shape 的 grid 完全相同却 K 不同，直接证明 grid 不能用来区分 K；只能靠 shape 审计而不是 launch 几何做归类。选中的正式 symbol 是 SM120 block-scaled E8M0/E4M3 CUTLASS GEMM，而不是旧 probe 里出现过的 sm89_xmma——旧 probe 的结论不能继承。

下一步 / 决策　把这三个 shape 固定为 scaled-mm 方向的正式 NCU 目标，并以等价门禁作为任何后续 kernel 替换的前置条件。

Step 82 | GPU0 侧的分类

动作　以同一套互斥分类统计 GPU0。

Profiling / 结果　稳态 taxonomy：VAE cuDNN BF16 convolution GEMM 73.257%、elementwise/fused 25.953%、reduction/norm 0.790%；formal 口径分别为 8,554.951 ms / 73.350%、3,016.668 ms / 25.865%、91.562 ms / 0.785%。

分析　GPU0 的结构比 GPU1 简单得多，单一类别就占近四分之三，意味着一旦 GPU0 进入关键路径，可动的面非常集中。

下一步 / 决策　把 VAE 卷积列为 P1 的第一目标，作为瓶颈转移的灵敏度边界而不是当前的收益来源。

Step 83 | 流水平衡：本章最重要的结构性结论

动作　对稳态 chunk 同时取 DiT wall、DiT CUDA event、VAE CUDA event 与两卡 busy 覆盖率。

Profiling / 结果　稳态 chunk 的 DiT wall 约 1,317–1,326 ms，DiT CUDA event 约 1,358–1,366 ms，VAE CUDA event 约 1,248–1,250 ms。稳态 envelope 中两卡同时 busy 87.327%，任一卡 busy 99.640%。

分析　同时 busy 87.327% 既证明流水重叠充分，也说明两卡 kernel sum 绝不能相加成端到端 wall。这里没有与 DiT wall 同口径的 VAE wall，因此不能声称严格的 wall-time crossover；只能给出约 5–9% 的启发式区间：GPU1 落入这一改善量级之后，GPU0/VAE 可能接近或进入关键路径。实际转折点还取决于 queue、backpressure、chunk 调度与最终 drain。

下一步 / 决策　把该区间写进所有 GPU1 优化的验收假设：stable-DiT FPS 可能继续明显上涨，而 generated FPS 不会长期按相同比例上涨。

瓶颈转移不是理论风险

GPU1 kernel sum 13,469.349 ms 与 GPU0 卷积 8,554.951 ms 分属两块卡，任何把它们相加再算" 总收益" 的推导都是错的。稳态 VAE CUDA event 约 1,248–1,250 ms 已经逼近 DiT wall 1,317–1,326 ms，因此 GPU1 侧 5–9% 的改善之后，收益曲线会先变平再取决于 VAE。[E-W8-PROF]

15.3　Attention 分阶段隔离 benchmark

Step 84 ｜四组生产形状的 CUDA-event 分阶段矩阵

动作　加载冻结生产二进制，做 attention 分阶段 CUDA-event benchmark，覆盖全部四组生产 (S_q, S_k)，每组 20 次预热加 100 次正式计时；12/12 二进制哈希核验，JIT build disabled，并用 Nsys 结构追踪核对 kernel 身份与 launch 数。

Profiling / 结果　20 秒加权投影下 provider e2e 为 2,461.068 ms；分阶段为 fused QK+online softmax+ 概率量化 +PV 1,347.229 ms（54.742%）、Q+K+V 量化合计 442.333 ms（17.973%）、BSH→BHSD 输入 layout 392.751 ms（15.959%）、Q/K FP32→BF16 边界 cast 307.814 ms（12.507%）、BHSD→BSH 输出 layout 74.550 ms（3.029%）、K/V padding diagnostic 60.204 ms（2.446%）。所有组件中位数之和为 2,628.061 ms。

分析　这些是各 stage 独立 CUDA-event 中位数按生产调用次数加权后的投影，不是同一次调用的时间分解。组件和 2,628.061 ms 大于独立测得的 provider e2e 投影 2,461.068 ms，这个差异本身就是非可加性的直接证据：不能相加成 wall time，不能反推" 未覆盖开销"，更不能宣称 cast+layout 的名义合计 31.5% 可以回收。

下一步 / 决策　分阶段矩阵只用于排序候选，不用于任何收益承诺；每个边界改动都必须单独回到四形状 CUDA event 加固定 20 秒请求验证。

表 15.2: Attention 分阶段 20 秒加权投影（独立 CUDA-event 中位数，非可加）

Stage	投影 (ms)	占 e2e 投影
Fused QK + online softmax + 概率量化 + PV	1,347.229	54.742%
Q + K + V 量化合计	442.333	17.973%
BSH→BHSD 输入 layout	392.751	15.959%
Q/K FP32→BF16 边界 cast	307.814	12.507%
BHSD→BSH 输出 layout	74.550	3.029%
K/V padding diagnostic	60.204	2.446%
组件中位数之和	2,628.061	—
独立测得 provider e2e 投影	2,461.068	100%

表 15.3: 四组生产形状对 provider 投影的贡献

(S_q, S_k)	调用次数	隔离中位 (ms)	投影 (ms)	占比
1560×1560	240	0.281328	67.519	2.743%
1560×3120	240	0.447424	107.382	4.363%
2340×5460	240	1.062080	254.899	10.357%
2340×6240	1,920	1.057952	2,031.268	82.536%

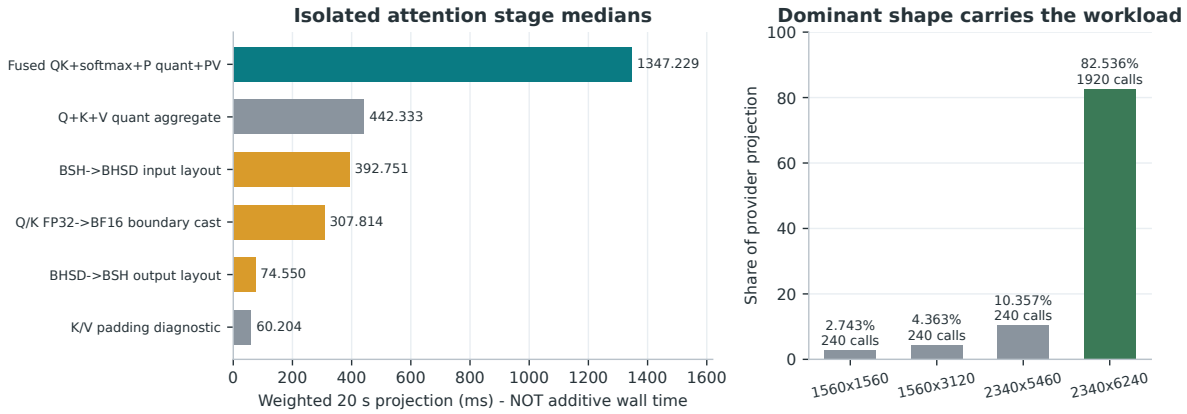

图 15.2: 左：隔离 stage 中位数的 20 秒加权投影，各 stage 非可加，组件和 2,628.061 ms 大于独立测得的 provider e2e 投影 2,461.068 ms；右：四组生产形状的贡献占比，主形状 2340×6240 以 1,920 次调用占 82.536%。左图任何柱形都不代表可回收的 wall time。

Step 85 ｜按主形状排序，而不是按最小形状上最漂亮的百分比

动作　把四形状贡献与 BF16 SDPA dispatcher 诊断放在一起比较。

Profiling / 结果　主形状 2340×6240 以 1,920 次调用贡献 2,031.268 ms / 82.536%。BF16 SDPA dispatcher 诊断值为 2,942.592 ms（119.566%）；主形状下自定义 provider 隔离中位 1.057952 ms 对比 BF16 SDPA 1.297024 ms，减少 18.432%、约 1.226 倍。

分析　18.432% 是隔离诊断，不等于全模型会获得同等收益；而 82.536% 的集中度说明，任何只在 1560×1560 上取得漂亮相对数的优化，对 20 秒请求几乎没有影响。

下一步 / 决策　后续 kernel 工作一律按主形状排序，小形状仅作为正确性与回归覆盖。

Step 86 ｜compute_attn_ws 能测什么、不能测什么

动作　尝试把 fused core 内部再拆成 QK、online softmax、概率量化、PV 四段。

Profiling / 结果　能精确测到 prequantized 输入到 fused core 输出的整体 CUDA-event 时间、正式 kernel 名称与 launch 数、四形状的中位/均值/P95，主形状约 0.597

ms/call。不能拆分 QK 矩阵计算、online row max/sum、softmax normalization、概率 MXFP8 量化、PV 矩阵计算，以及它们之间的 TMA/QMMA/warp-specialized pipeline overlap。

分析　它们属于同一个 CUDA kernel；给内部源码块加 NVTX 无法在 device kernel 内生成独立的 GPU wall range，所以这不是" 还没做"，而是这条测量路径本身不成立。

下一步 / 决策　内部结构只能靠硬件计数器回答，因此 fused core 的下一步强依赖 NCU 解锁（见 Step 89–90）。

Step 87 | Q/K 单 launch 合并：加权更慢，拒绝

动作　把 Q 与 K 的量化合并成单次 launch，在四形状上做诊断对比。

Profiling / 结果　1560×1560 快 17.117%，但 1560×3120 慢 3.899%、2340×5460 慢 1.854%、2340×6240 慢 1.731%；按 20 秒生产调用次数加权后慢 0.934%，即 212.655 ms 升至 214.641 ms。

分析　最小形状上的 17.117% 完全被主形状的 1.731% 回退吃掉，这正是" 按主形状排序" 规则的实例；同时说明 launch 数并不是这条路径的主导成本。

下一步 / 决策　不推进全局 Q/K single-launch，**REJECT**；若将来只在小形状分支启用，必须重新做四形状加权而不是引用局部数字。

Step 88 | native SDPA 分支的 role 归因

动作　对 PyTorch SDPA/Flash 这一 broad 分类按 role 下钻。

Profiling / 结果　broad name-based 部分量为 7.406%，其中 native core 765.893 ms、TorchInductor support 240.614 ms；H128 regular video-to-text 单项 589.289 ms，占 native core 的 76.943%；SplitKV combine 总计仅 17.289 ms。

分析　这个分支只有一个明确的 P0，即 H128 regular video-to-text；SplitKV combine 的 17.289 ms 不具备优化价值，投入产出比远低于 scaled-mm 与 fused core。

下一步 / 决策　H128 video-to-text 进入 P4，排在 scaled-mm、VAE 与 provider 边界之后。

15.4　Nsight Compute 权限缺口

Step 89 ｜ NCU 第三次被同一个错误码阻断

动作　在冻结镜像内跑最小 NCU probe，确认是否可以采集硬件计数器。

Profiling / 结果　probe 返回 ERR_NVGPUCTRPERM。宿主机驱动为 RmProfilingAdminOnly:1；容器虽 uid=0，但 current 与 bounding capability set 均不含 cap_sys_admin 与 cap_perfmon。这是第三次记录该错误（2026-07-20 两次、2026-07-23 一次）。

分析　capability 不在 bounding set 时，容器内 sudo、setcap 或任何普通提权都不能凭空增加该能力；并且在本机驱动策略下 CAP_PERFMON 不是 CAP_SYS_ADMIN 的等价替代。因此这不是容器内可以绕开的问题。

下一步 / 决策　Nsight Compute 方向 **BLOCKED**，改为把解锁动作明确移交宿主机管理员。

表 15.4: NCU 解锁的两条路线（均须宿主机管理员执行）

路线	动作	约束
A	宿主机设置 NVreg_RestrictProfilingToAdminUsers=0	persistent 写法是 /etc/modprobe.d 配置，随后重建 initramfs 并重启；在用的 NVIDIA 模块无法安全 reload
B	以 CAP_SYS_ADMIN 同时进入 bounding 与 effective set 的方式重启容器	仅改容器内提权无效，必须由宿主机侧重建容器

Step 90 ｜ targeted NCU 已就绪，只等权限

动作　预先写好并自检两个 targeted NCU launcher，使权限恢复当天即可取数。

Profiling / 结果　run_ncu_attention.sh 固定主形状 2340×6240，对 Q（signed_h128_mxfp8_physical_sf_kernel 第一次 occurrence）、K（同 symbol 第二次）、V（v_transpose_mxfp8_k32_kernel）与 fused core（compute_attn_ws）各采一个目标 launch，显式收集九个 section（LaunchStats、Occupancy、SpeedOfLight、ComputeWorkloadAnalysis、MemoryWorkloadAnalysis、SchedulerStats、WarpStateStats、InstructionStats、WorkloadDistribution），只有 fused core 额外收 SourceCounters；四项都使用 --launch-count1--killyes。run_ncu_scaled_mm.sh 同样已就绪，并已通过三项 Nsys 等价门禁。两个 launcher 都在创建输出目录之前做 fail-closed 权限预检，任何证据缺失都以 status 3 退出。

分析　fail-closed 预检的目的很具体：避免留下貌似已运行的半成品目录，从而在后续复盘中被误当作真实证据。

下一步 / 决策　在取得 counter 之前，不能断言 fused core 或 scaled-mm 是 compute-bound、memory-bound、register-bound 还是 softmax dependency-

bound。

15.5　计时口径之外的阶段（Week 8 首次量化）

Step 91 ｜不在 generated FPS 分母内的时间

动作　在同一条 trace 上量化 generated FPS 分母之外的阶段。

Profiling / 结果　media write/encode 7,424.061 ms，且该 NVTX 范围内没有任何 CUDA kernel；audio decode 313.123 ms（其中嵌套 VAE decode 311.391 ms）；text encoder 801.745 ms；condition copy 6.417 ms。

分析　media write/encode 的量级与 formal 请求 wall 同数量级，但它完全不在 generated FPS 分母内，也不消耗 GPU kernel 时间；把它计入模型性能会同时高估问题和高估收益。

下一步 / 决策　若目标包含用户等待时间，应以未插桩、预热后的 request-to-final-MP4 wall 另行复测，并明确禁止通过降低编码质量来偷换模型生成性能。

15.6　下一步排序

15.6.1　全局 P0：K32 scaled-mm GEMM

量级必须诚实地写出来。若把全部稳态 scaled-mm GEMM 加快 10%，GPU1 kernel sum 约减少 304.281 ms，相当于 steady envelope 的 3.73% 串行上限；若三个 P0 shape 各加快 20%，设备归因约减少 89.4 ms/steady chunk。由于 GPU0 的 VAE 与之并行，实际 generated FPS 收益会小于这两个数字。[E-W8-PROF]

15.6.2　Attention 内部 P0：主形状的 fused core，但必须先开放 counter

若 fused core 加快 20%，隔离 20 秒投影减少约 269.446 ms；对应到 full trace，稳态 fused core 862.188 ms 的 20% 约 172.438 ms，约为 steady envelope 的 2.11% 串行上限。counter-to-action 顺序由源码审计给出：

1. 先测 FA3 风格的 cache hint，即 Q 用 EVICT_FIRST、K/V 用 EVICT_LAST。
2. 确认存在 one-stage starvation 之后，才尝试两个 K/V stage；纸面 shared 约 101,376 B，已接近源码 guard。
3. 若 SourceCounters 定位到 exp2、running max-sum 或 P quant，则只改指令调度。

4. 若证据指向 register pressure，再去缩短 live range。
5. 若 tensor pipe 已经高利用，则不要先增加 stage。

表 15.5: 下一轮优先级与前置条件

优先级	方向	依据与前置条件
P0	K32 scaled-mm GEMM（三个 P0 shape）	稳态占 38.489%、formal 占 kernel sum 39.231%；三个 shape 占稳态 scaled-mm 87.665%；Nsys 等价门禁 3/3 PASS
P0	主形状 2340 × 6240 的 compute_attn_ws	该形状占 provider 投影 82.536%；内部结构不可用 NVTX 拆分，必须先解锁 NCU
P1	GPU0/VAE 瓶颈转移灵敏度	第一目标 VAE cuDNN BF16 convolution GEMM（稳态 73.257%），第二为 elementwise/fused（25.953%）；latent copy 约 0.16 ms/chunk 不是目标
P2	provider 边界/layout 与 producer 融合	每次只改一个边界，先跑四形状 CUDA event，再跑固定 20 秒请求
P3	全局 Q/K single-launch	20 秒加权慢 0.934%，**REJECT**，不推进
P4	native SDPA 的 H128 video-to-text	589.289 ms，占 native core 76.943%，该分支唯一明确的 P0
P5	media write	7,424.061 ms，但不在 generated FPS 分母内；仅当目标包含用户等待时间时才做

15.6.3 full-model A/B 的强制报告项

所有 full-model A/B 都必须同时报告 stable DiT FPS、generated FPS、per-chunk DiT/VAE、backpressure/drain 与两卡 busy。只报告其中一项（尤其是只报 stable DiT FPS）会让瓶颈转移完全不可见：在 GPU1 改善落入 5–9% 区间之后，stable DiT FPS 可能继续明显上涨，而 generated FPS 不会按相同比例上涨。[E-W8-PROF]

Week 8 下的两个 P0 与一堵 VAE 墙

设备分离归因把优化面收敛成两个 P0：全局层面是 K32 scaled-mm GEMM（formal 口径 5,284.153 ms、kernel sum 39.231%、请求 wall 34.480%，加上 support 合计 41.731%），attention 内部层面是主形状 2340 × 6240 的 compute_attn_ws（占 provider 投影 82.536%）。两者的串行上限都不大：scaled-mm 全体加快 10% 对应 steady envelope 3.73%，fused core 加快 20% 对应 2.11%。更重要的是那堵墙：稳态 VAE CUDA event 约 1,248–1,250 ms 对 DiT wall 约 1,317–1,326 ms，两卡同时 busy 87.327%、任一卡 busy 99.640%，因此 GPU1 侧约 5–9% 的改善之

后 GPU0/VAE 可能进入关键路径，这是启发式区间而不是 wall-time crossover。[E-W8-PROF]

两块最大的优化面仍是黑盒

Nsight Compute 因 ERR_NVGPUCTRPERM 第三次被阻断（RmProfilingAdminOnly:1，bounding set 无 cap_sys_admin/cap_perfmon），解锁必须由宿主机管理员执行路线 A 或 B。在取得硬件计数器之前，占 GPU1 最大份额的 K32 scaled-mm GEMM 与占 attention 最大份额的 fused core 都无法判定是 compute-bound、memory-bound、register-bound 还是 softmax dependency-bound；任何据此提出的 kernel 改动都只是猜测。同时提醒：本章的 name-based attention 份额 13.442% 是下界而非全量，分阶段矩阵不可相加（组件和 2,628.061 ms > e2e 投影 2,461.068 ms），cast+layout 名义合计 31.5% 不可被当作可回收预算。[E-W8-PROF]

第八部分

Portrait 基线、Week 9 证据与当前 v2

16

Catnip Portrait v1：历史工程基线

历史结论：v1 已被 v2 取代

catnip-k32-vdirect-portrait-480x832-v1 于 2026-07-26 成为默认 engineering-performance baseline，并于 2026-07-28 被第18章的 v2 取代。v1 在双 RTX 5090 上以 480×832 portrait geometry 完成三次独立 fresh-cache 的 20 s 请求：**37.486982 stable-DiT FPS** 与 **33.070761 generated FPS**，population CV 分别为 0.043284% 与 0.064495%。这些数值保留为历史前序证据，不再是 current KPI。

该结论是**工程吞吐与运行时合同**结论，不是视觉质量已经正式验收的结论。三次运行均通过 request、K32、V-direct、KV、compile 与 media gates，且 fallback/upcast/retained converted BF16 weight 均为零；但 manifest 明确规定 formal_video_quality_acceptance=false。因此它只能表述为当时的 active engineering baseline，不能表述 portrait quality unchanged。

16.1 基线身份、工作负载与计时口径

Portrait 只改变输出空间方向，未改变每 latent frame 的 390 spatial tokens、frozen noise、chunk schedule 或双卡 placement。该选择不是将 landscape 的数字重新贴到 portrait 上，而是在新 geometry 下重新执行完整请求，并以三个 create-only result roots 验证运行时合同。

表 16.1: Portrait v1 的历史冻结合同

项目	冻结值
Baseline identity	catnip-k32-vdirect-portrait-480x832-v1；SUPERSEDED，active 2026-07-26 至 2026-07-28
Geometry / media	480×832，requested 20.0 s，489 frames，25 FPS playback，encoded duration 19.56 s
Streaming semantics	62 LF，11 chunks，55 DiT forwards，first 4 LF sink + latest 6 LF recent KV
Placement / scheduling	GPU1 complete DiT；GPU0 text encoder 与 VAE；per-chunk video decode pipeline depth 3，video drain 后单次 audio decode
Stable-DiT FPS	$288/\sum \text{DiT wall}_{\text{chunks 5–10}}$；不是 MP4 playback FPS
Generated FPS	489/(sampling + pipelined video decode)；不包含 text、audio decode 或 media write
Compilation	48 Transformer blocks independently torch.compile(mode=default)；每个 repetition 以 absent compile cache 开始

16.2 三次独立运行：v1 历史吞吐证据

表 16.2: Portrait baseline 的未插桩重复性

Run	Stable-DiT FPS	Generated FPS	Sampling + video decode wall (ms)
rep01	37.464035	33.040672	14,799.941
rep02	37.498314	33.087641	14,778.932
rep03	37.498596	33.083971	14,780.572
Mean	**37.486982**	**33.070761**	14,786.482
Population CV	0.043284%	0.064495%	—

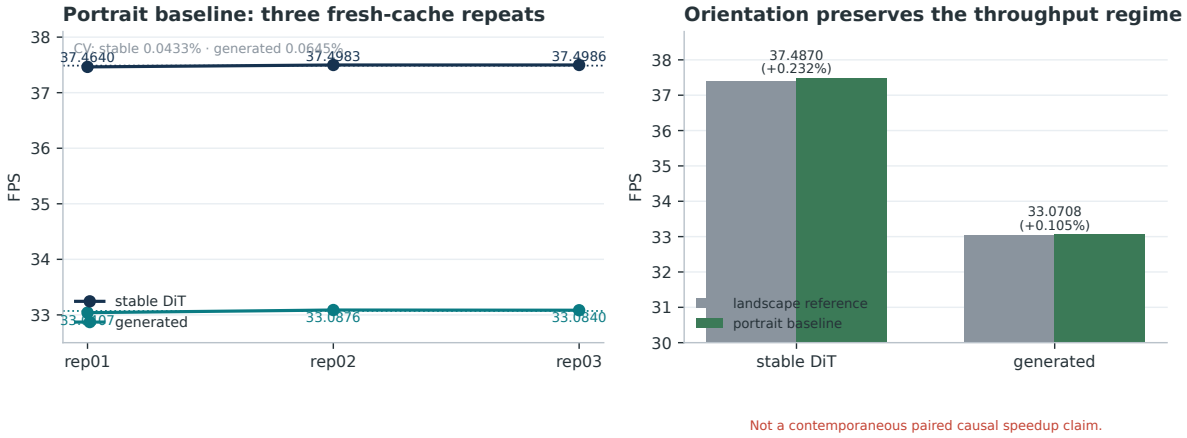

图 16.1: 左：Portrait baseline 三次 fresh-cache 独立运行，stable-DiT 与 generated FPS 均具有很低的 population CV。右：最新 landscape V-direct reference 与 Portrait 的均值只相差 +0.231955% / +0.105325%。这不是 contemporaneous paired A/B，故仅支持 throughput regime preserved，不支持 portrait orientation 带来因果 speedup 的说法。

16.3 Mixed-precision system contract

当前配置不是 full-model FP8。Transformer source dtype 为 BF16；选定的 576 个 Linear 使用 native K32 MXFP8 scaled-MM：float8_e4m3fn qdata、沿 GEMM K 维的 32-element E8M0 scale、BF16 output 与 fused BF16 bias。该集合覆盖 13,690,208,256 个 weight elements，约占 all-Linear weight elements 的 72.138575%。其余模块仍按照各自的 BF16 或已冻结实现执行。48 个 video-self attn1 使用 V-direct-BSH provider；text encoder 为 bitsandbytes INT8。

表 16.3: Portrait 基线的 fail-closed runtime 断言

断言	观察值与解释
Selected Linear coverage	576 native K32 MXFP8 Linears；在 weight storage、activation quantization 与 scaled-MM execution 三处均有 gate
Fallback calls	0；不得将 SDPA 或 BF16 Linear fallback 混入 FP8 performance claim
Weight upcast calls	0；不得在 GEMM 前把 FP8 weight 转回 BF16
Retained converted BF16 weight bytes	0；不得将 converted BF16 shadow weight 保留在运行时
Attention installation	48/48 attn1 strict fail-closed；若 provider 未正确安装，运行应停止而不是静默替换

16.4 Portrait 与历史 landscape evidence 的关系

最新 832 × 480 V-direct engineering reference 是 37.400230 stable-DiT FPS / 33.035966 generated FPS；Portrait 的对应均值为 37.486982 / 33.070761。二者请求有相同 token count 和 streaming ABI，但不是同一时刻交替执行的 paired A/B，也没有以 orientation 为唯一实验变量的预注册因果设计。因此可接受的结论仅为：portrait re-orientation preserves the established throughput regime；观测差异不应解释为 causal speedup。

质量与因果边界不可省略

Portrait examples 可用作 qualitative demonstration，但不代替 blind review、perceptual panel 或 heldout acceptance。类似地，Week 8 的 36.866330 stable-DiT / 32.572402 generated FPS 与 Week 9 的 V-direct paired result 都属于前序 832 × 480 stack 的历史证据；它们解释为何当前 stack 采用 K32 与 V-direct-BSH，但不能替代本章的 Portrait 重复性测量。

16.5 可复现证据

本章数值由现有、内容寻址的 artifacts 导出：catnip-portrait-baseline/results/BASELINE_MANIFEST.json、results/PERFORMANCE_SUMMARY.json、EXPERIMENT_RESULT.md 及三个 results/performance/rep*/portrait_baseline_gate.json。每个 performance run 都由独立 gate hash 绑定；本报告未观看输出视频来推出质量结论。[E-PORTRAIT]

17

Week 9：Profiling-guided boundary optimization 与路线图

结论先行：一项 accepted engineering change，一项完成的负结果

Week 9 没有把 Week 8 profiling 的百分比直接当成端到端收益，而是先修正微基准的 stationarity，再以 four-shape isolated provider、Nsys launch structure 与三对 full-model A/B 逐层验证。唯一通过的变更是 **V-direct-BSH**：让 attention provider 的 V producer 直接消费 producer-side BSH storage，消除显式 V layout/padding 边界。它在历史 832×480 frozen K32 stack 上取得 provider-weighted +7.5686%，并在三对 20 s full-model A/B 上取得 stable-DiT median **+1.289408%**、generated median **+1.284480%**，所有三对均为正。

同一周对 dominant scaled-MM shape 的 cuBLASLt heuristic top-k pinning 也完成了正确的 early stop：bitwise-correct candidate 仅 +0.058769%，低于预注册 +1.0% gate，终态为 **TIMING_REJECTED**。这拒绝的是 frozen 1 MiB workspace、use_fast_accum=false 下的 algorithm pinning，不是否定 GEMM 的所有优化空间。

17.1 证据对象与基线边界

本章所有 causal A/B 的 subject 是 Week 8 的历史 832×480 K32 stack：20 s request、489 frames、11 chunks、55 DiT forwards、4+6 KV 和 48 attn1 provider modules。起始未插桩 reference 为 36.866330 stable-DiT FPS / 32.572402 generated FPS。它是 Week 9 candidate 的严格 control，不是当前 Portrait baseline 的绝对 benchmark。

禁止跨合同替换 headline

本章的 37.400230 / 33.035966 是 V-direct-BSH candidate 在历史 landscape 合同上的 A/B 均值；当前对外 engineering KPI 已更新为 章 18 的 37.557043 / 34.392314。Week 9 可证明一个 attention-boundary 变更的工程收益，Portrait v1 与 v2 分别提供历史几何稳定性和当前集成 KPI；三者不能合成为单一 causal claim。

17.2 Profiling 结论如何变成可检验问题

Week 8 device-separated profiling 给出了方向而不是收益承诺：GPU1 的 K32 scaled-MM GEMM 为稳态 kernel sum 的 38.489%，三个 P0 GEMM shape 占其 87.665%；GPU0 的 VAE cuDNN BF16 convolution GEMM 占 73.257%。Attention isolated provider 的主形状 2340×6240 占投影 82.536%，fused core 占投影 54.742%。同时，FP32→BF16 boundary cast 与 BSH→BHSD layout 的独立 CUDA-event 投影名义和为 775.116 ms（31.495%）。

最后一个数字尤其容易被误读。各 stage 是独立 median 再按 production call count 加权，components sum 为 2,628.061 ms，而 independently measured provider e2e projection 为 2,461.068 ms；两者不相等正说明执行区间存在 overlap。因而 31.495% 是 candidate ranking evidence，**不是可回收 wall-time budget**。

表 17.1: 从 Week 8 profiling 到 Week 9 实验问题的映射

Profiling observation	Week 9 testable hypothesis / boundary
V layout/padding 处于高优先级 boundary 区域	只改变 V consumption：control materializes BSH→BHSD；candidate direct-strided BSH consumption；Q/K、fused core、output、KV、weights、VAE 与 scheduler 固定
主形状占 provider projection 82.536%	不能只测漂亮的小形状；必须覆盖四个 production (S_q, S_k)，并以 production call mix 加权
两卡 DiT/VAE 已接近平衡	full-model 必须同时报告 stable-DiT 与 generated FPS；GPU1 优化不可外推为同等 generated-FPS 增益
三个 scaled-MM shape 集中度高	先做 low-cost algorithm screen；当 dominant shape 未达 gate 时按合取规则 early stop

17.3 先修 measurement stationarity，再运行 candidate

早期 attention micro A/B 发现系统性位置偏差。Week 9 的 stationarity diagnostic 将 slow/reference 的 +0.183656 ms/call 差异拆开：host enqueue +0.182802 ms（99.535%）与 inter-kernel gap +0.172042 ms（93.676%）均显著，而 summed kernel duration 反而为 -0.002643 ms。前两项有重叠，不能相加；但它们足以排除将 host launch noise 归咎于 kernel 的解释。

因此后续改为四形状 blocked ABBA/BAAB，且在 formal timing 前要求 steady precondition。baseline-versus-baseline neutrality 最终达到 NEUTRALITY_ESTABLISHED：31/31 hard checks PASS，weighted candidate/con-

trol ratio 为 1.000135913，CV 0.083171%，90% CI 为 [0.999953358, 1.000320021]。这不是 candidate gain，而是后续收益解释成立的前提。

17.4 V-direct-BSH：从 isolated evidence 到 20 s A/B

Control 路径在 attention core 前显式将 BF16 V 从 BSH materialize 为 BHSD；padded shape 还需要 V padding fill/copy。Candidate 保留 producer-side strided BSH storage，由预编译 custom CUDA op 直接读取，Q/K conversion、fused core、output layout 与其余模型语义不变。该设计不把 launch count 当作性能结论，而是同时验证数值、structure 与 full-model effect。

表 17.2: V-direct-BSH 的逐层验证

层级	结果	可得结论
Four-shape provider e2e	1560×1560 +9.8102%；1560×3120 +7.8235%；2340×5460 +12.4382%；2340×6240 +6.8667%；weighted +7.5686%	provider boundary 确有时间收益；不能直接预测整机收益
Nsys structure	aligned 10→9 launches/call；padded 14→11；160 calls 总 launches 960→800	省掉的 layout/padding work 没有通过 hidden generic copy 重新出现
20 s paired full model	stable DiT 36.923232→37.400230；generated 32.624456→33.035966	Frozen K32 stack 上的 engineering-performance improvement

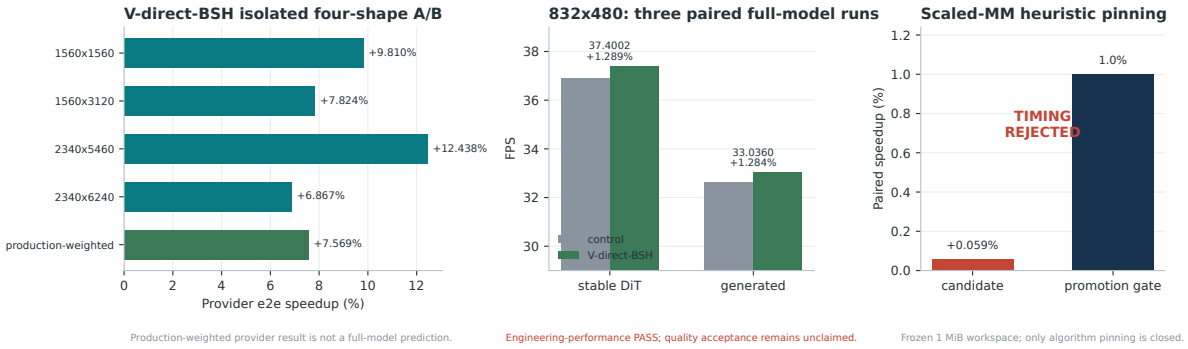

图 17.1: 左：V-direct-BSH 在四个生产 shape 与 production-weighted provider e2e 上均为正。中：历史 832×480 stack 上三对 full-model A/B 的均值与 paired median gain；该实验仅通过 engineering-performance gate。右：dominant scaled-MM heuristic pinning 虽 bitwise correct，但 +0.058769% 未达到 +1.0% promotion gate，故正确终止。

表 17.3: 历史 landscape 合同上的 V-direct-BSH full-model performance gate

Metric	Control mean	Candidate mean	Paired median gain
Stable-DiT FPS	36.923232	37.400230	+1.289408%
Generated FPS	32.624456	33.035966	+1.284480%
Positive pairs	3/3	3/3	worst stable pair +1.198660%
Stable CV	0.066586%	0.021267%	ratio CV 0.076200%
Peak allocated delta	—	GPU0 0 B; GPU1 -18,190,848 B / pair	within gate
Execution / K32 gates	6/6	6/6	77/77 candidate checks in each candidate lane

V-direct 的 claim boundary

full-model performance gate 的 blocking checks 全部通过，但 legacy raw prompt contract 含一个 terminal LF，production normalizer 在所有六条 lane 中一致地移除了该 LF。versioned adjudicator 证明 runtime prompt bytes 相同，故 paired_performance_evidence_valid=true；同时保留 original_contract_exact_compliance=false。更重要的是，candidate_quality_accepted=false 与 production_acceptance_decided=false。故本章不将该结果表述为 quality acceptance 或 final production promotion。

17.5 已完成的负结果：cuBLASLt heuristic pinning

Phase 3 在 production-equivalent K32 W8A8、BF16 bias/output、FP32 accumulate、frozen 1 MiB workspace 与 use_fast_accum=false 下，对 $2340 \times 4096 \times 4096$ dominant shape 进行 correctness-filtered top-64 heuristic screen。返回的四行仅有两个 distinct fingerprints；唯一 candidate 在 normal、sparse 与 HDR cases 都与 production output bitwise equal，但 control/candidate block median 仅为 0.139531201 / 0.139427197 ms。

预注册 primary gate 为 paired median speedup $\geq 1.0\%$；实测 +0.058769%。由于 three-shape stage 是 conjunction，dominant shape 已失败即不可能总体通过，故没有继续执行另外两个 FFN shape、complete Linear、Nsys 或 20 s video。这是 protocol-compliant early stopping，而不是缺失实验。

17.6 最新 NCU counter-free evidence 与路线图更新

2026-07-27 发现 ncu--setnone 能在没有 PerfWorks counters 的情况下导出 launch configuration、occupancy limiting factor 与 wave quantization；tensor-pipe utilization、warp stalls、achieved occupancy 与 NCU duration 仍然被权限阻断。该新证据缩小了盲区：

- compute_attn_ws 主形状 grid 为 170 CTA，等于 170 SM，恰好 1.00 wave；不存在 tail wave。168 registers/thread 与 84.992 KiB dynamic shared memory 同时 binding，理论 occupancy 25%。这不是可线性回收的 75% headroom；要达到 2 CTA/SM 需要近乎重写 kernel。
- P0 scaled-MM $2340 \times 4096 \times 4096$ 为 608 CTA / 3.58 waves，wave efficiency 为 89.4%，仍有 10.6% tail。因而 stream-K 或 persistent scheduling 是可检验假设；增大 tile 或 pipeline stage 则被 register/shared-memory 余量证伪为低优先级。

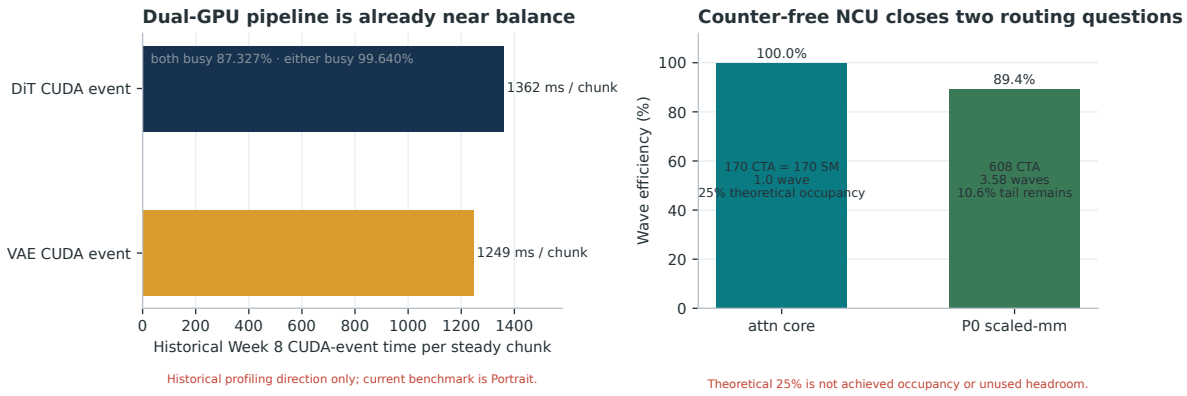

图 17.2: 左：历史 Week 8 steady chunk CUDA-event 显示 DiT 与 VAE 已接近平衡，说明只优化 GPU1 会很快受 VAE wall 限制。右：counter-free NCU 的 wave facts；理论 25% occupancy 不等于 achieved occupancy，也不应被误读为可直接回收的性能空间。

表 17.4: 报告日期下的优化排序

Priority	Action	Evidence-based admission rule
P0 dual-track	GPU0 VAE: channels_last、cuDNN autotune、torch.compile、decode tile A/B	与 GPU1 并行执行；否则 generated FPS 将在约 34 附近受到 pipeline balance 限制
P0 boundary	Q/K cast fusion 或 producer direct-layout	每次只改一个边界；four-shape CUDA-event A/B 后再进 20 s paired full-model gate
P0 GEMM	Epilogue fusion of next-layer K32 quant/scale pack；或 stream-K/persistent scheduling	heuristic pinning 已关；不得把其负结果外推到 workspace、CUTLASS 或 epilogue fusion
P1 precision-preserving attention	attn2 video-to-text 的 BF16 provider replacement	只换 provider，不降低精度；该 family 是质量敏感区
P1 quality	blind review → freeze → validation → heldout-v2	工程吞吐不自动升级为正式 quality baseline
P2 user waiting	NVENC / streaming mux	单独量 request-to-final-MP4 wall；不混入 generated-FPS 分母

17.7 证据定位

有效终态分别为 PHASE02_V_DIRECT_BSH_FULL_MODEL_V2_RESULT.md 与其 FINAL_VALIDATION.json、PHASE03_SCALED_MM_TOPK_RESULT.md 与 FINAL_STATUS.json、以及 NCU_COUNTERFREE_FINDINGS_20260727.md。早期 V1 harness failure、V2 extreme-BF16 finite-gate rejection 与 full-model attempt01/02 都被隔离为历史基础设施记录，不能作为 V-direct candidate efficacy 的证据。[E-W9]

18

2026-07-28：Portrait v2 基线晋升

当前默认基线

catnip-k32-vdirect-single-tile-compile-portrait-480x832-v2 已按用户决策晋升为当前 engineering baseline，并取代上一章记录的 v1。冻结的 20 秒合同、K32 Linear、V-direct-BSH attention、4+6 KV 与双卡 placement 均未改变；本次只接纳已经完成独立报告与配对实验的 VAE 运行时改动。本文不重复该 VAE 报告的实验过程。[E-PORTRAIT-V2]

18.1 晋升内容与当前 KPI

表 18.1: Portrait v2 当前状态

项目	当前值与证据边界
Release identity	catnip-k32-vdirect-single-tile-compile-portrait-480x832-v2；ACTIVE，adopted 2026-07-28
Accepted runtime changes	每个 Catnip latent window 使用 single tile；仅编译 video_vae_decoder.forward，mode 为 default
Authoritative warm KPI	三对 Phase 3 candidate mean：**34.392314 generated FPS**、**37.557043 stable-DiT FPS**；generated-FPS population CV 0.085863%
Paired compile effect	相对 single-tile eager，generated-FPS paired median gain **+1.837371%**；不与前一阶段增益相乘
Resource gate	三对正式运行的最小 GPU0 headroom 为 4.923584 GiB
Promotion smoke	最终无争用 r03 PASS：34.408774 generated FPS、37.566968 stable-DiT FPS、GPU0 headroom 4.728271 GiB
Media gate	H.264 480 × 832、489 frames、25 FPS、19.56 s；AAC 48 kHz stereo

18.2 晋升门禁与声明边界

最终 r03 smoke 保留了 576 个 K32 Linear、48 个 V-direct-BSH attn1、55 次 DiT forward、first 4 LF + latest 6 LF KV 与 depth-3 decode；16 次 compiled decoder forward 只产生一个初始图，正式推理期间没有新增 unique graph。fallback 与 weight upcast 继续为零。默认入口现为：

```
/workspace/LTX/.venv/bin/python \
  /workspace/catnip-portrait-baseline/run_baseline.py
```

只接受 warm-throughput，不改写质量结论

本次晋升的权威性能口径是 warm steady state。fresh-process wall 的 paired median 反而增加 10.802722%，所以不得宣称 cold-start speedup。已有 VAE 报告提供的 real-latent comparison 只有一个 frozen prompt、没有 ground truth 或正式 blind panel，且未直接比较 compiled variant；因此本报告仍不声明 formal ground-truth quality acceptance。两次并发作业导致、且发生在 VAE decode 前的 OOM smoke 已标为 invalid，不参与性能或算法结论。

18.3 证据入口

当前指针、v2 manifest、性能汇总、晋升报告与最终 smoke gate 分别位于 `/workspace/catnip-portrait-baseline/results/CURRENT\_BASELINE.json`、`results/v2/BASELINE\_MANIFEST\_V2.json`、`results/PERFORMANCE\_SUMMARY\_V2.json`、`BASELINE\_V2\_RESULT.md` 与 `results/v2/smoke\_20260728\_r03/portrait\_baseline\_gate.json`。VAE 的详细设计、实验矩阵和视频比较继续以现有 Week 10 与 comparison 报告为准，本文不复制。[E-PORTRAIT-V2]

第九部分

最终决策与实验账本

19

当前 Portrait 生产方案、历史 profiling 边界与下一步

Current engineering state

当前基线是 Portrait catnip-k32-vdirect-single-tile-compile-portrait-480x832-v2，而非本章后续引用的历史 832×480 K32 profiling subject 或已被取代的 Portrait v1。它以 576 K32 MXFP8 Linears、48 V-direct-BSH attn1、GPU1 DiT / GPU0 text+VAE placement 与 depth-3 decode，在三对正式 candidate runs 中达到 37.557043 stable-DiT FPS / 34.392314 generated FPS。最终无争用 r03 smoke 通过全部运行时与媒体门禁；该性能结论只覆盖 warm steady state，formal ground-truth quality acceptance 仍未建立。

本章的 kernel taxonomy、VAE crossover 与 NCU 方向来自历史 landscape profiling，因而只用于 priority ordering。Week 9 已使 V-direct-BSH 的 historical engineering A/B 通过，并用 counter-free NCU 关闭了 wave/tile 两个问题；这些更新均不把历史 profiling 的绝对 FPS 覆盖到 Portrait。

19.1　推荐拓扑

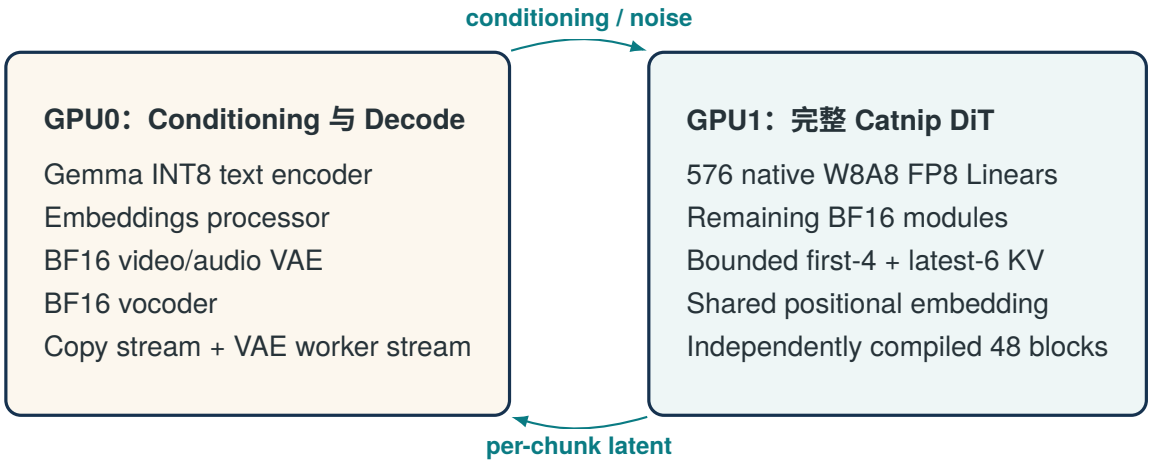

图 19.1： 最终双 RTX 5090 placement。关键路径上不再搬运 24/24 block boundary 的完整 Transformer state。

表 19.1: 推荐目标配置与当前验收状态

项目	冻结配置
Quantization	576 selected Linear 全部为 native K32 MXFP8：E4M3 qdata，沿 GEMM 的 K 维每 32 个元素一个 UE8M0 scale（activation 每行、weight 每输出行各自成组）；BF16 output + fused BF16 bias；禁止 split-K、fake quant、保留 BF16 权重、fallback 与 upcast
Attention	48 个 video-self attn1 使用自研 SM120 MXFP8 kernel（signed H128 → E4M3/UE8M0 K32 → block-scaled QK → online softmax + online P K32 quant → block-scaled PV → BF16）；H128 producer 不与 core 融合
GEMM	Native aten._scaled_mm；use_fast_accum=False；fallback/upcast/retained BF16 converted weight = 0
Streaming	Portrait 480×832、20 s、489 frames、11 chunks、55 forwards、first 4 + latest 6 LF KV；每 LF 390 spatial tokens
Scheduling	GPU1 DiT 与 GPU0 逐 chunk VAE overlap；depth-3 video decode pipeline，video drain 后一次 audio decode；当前 v2 每个 latent window 使用 single tile
Compilation	48 DiT blocks independently compile；v2 另编译 video_vae_decoder.forward，只声明 warm-throughput
Provider	V-direct-BSH 48/48 attn1 strict fail-closed；历史 16 MiB cuBLASLt workspace 保持冻结环境的一部分
Current Portrait baseline	Phase 3 candidate mean 34.392314 generated FPS；mean 37.557043 stable-DiT FPS；generated CV 0.085863%；25 playback FPS
Historical paired controls	tensorwise/K32 与 control/V-direct 的严格 landscape A/B 分别保留为因果工程证据；不得替代 Portrait KPI
Historical delivery	O7 mean wall 18,041.006 ms；mean 27.104924 generated FPS（Week 2 交付，保留为历史参考）
Release identity	catnip-k32-vdirect-single-tile-compile-portrait-480x832-v2；ACTIVE engineering baseline，warm-throughput only
Quality policy	Portrait runtime/media contracts PASS，但不主张 ground-truth perceptual quality 已验收；全家族 rowwise、channelwise、SmoothQuant 仍非默认生产策略

19.2 已经完成且可复用的成果

1. 正确的 20 秒 Streaming ABI 与 4+6 KV 语义。
2. 完整 mixed DiT 单卡 placement 与 TE/VAE 功能拆分。
3. 576-module native scaled-MM 与 0 fallback/upcast 门禁；量化编码已由 tensor-

wise 演进为 K32 MXFP8。

4. Bounded KV、shared PE、fused bias 与逐 chunk VAE scheduler。
5. 48-block compile、duplicate V2A fold 与五 shape warmup。
6. O7 三轮稳定的 27.104924 generated FPS 交付证据，以及已固化的 16 MiB cuBLASLt workspace。
7. 自研 SM120 MXFP8 attention kernel：CUTLASS 4.5.2 blockscaled 特化 + FA3 血统 persistent core + 手写 producer，fail-closed 替换 48 个 attn1。
8. **Reference-free 解码媒体质量面板**：经伪影注入标定，以 BF16 与生产 FP8 的自然差异带归一化。
9. **内容寻址的可复用实验基线与配对候选 harness**：三对交替顺序、独立 cache root、独立验证器重放。
10. 576 层 outlier 统计与 INT8 机制对照，明确了权重侧与激活侧的收益分布。
11. Week 8 设备分离 profiling：45 分析窗口、12 stage gate、3 attention count gate、5 sequence-pair gate 全 PASS。
12. Portrait v2 晋升：接纳已独立报告的 single-tile 与 video VAE decoder compile；三对 candidate mean 34.392314 generated FPS，final r03 smoke PASS。

19.3　下一步按证据排序，而不是继续铺候选矩阵

表 19.2: 下一轮优先级与 admission gate

优先级	实验	为什么现在做	Admission
P0 counters	申请完整 NCU counter permission	ncu--setnone 已得到 launch/occupancy-limit/wave，但 tensor-pipe、stalls、achieved occupancy 与 duration 仍不可见	宿主机解除 profiling restriction，或容器取得 CAP_SYS_ADMIN；不得把 theoretical occupancy 当实测效率
P0 质量	两人独立盲评	它是 freeze、validation 与 heldout-v2 的唯一阻塞；公开包与响应验证器已就绪	两份独立裁决写入 quality_calibration. \protect\penalty\z@ freeze.json
P0 吞吐	Epilogue fusion 或 scaled-MM scheduling	三 P0 shapes 占稳态 scaled-mm 87.665%；heuristic pinning 已以 +0.058769% 正确拒绝	每个新 hypothesis 独立 pre-register；stream-K/persistent 可测，tile/stage 扩张已降级
P0 冷启	Compile cache 与预热摊销	v2 的 warm throughput 已提升，但 fresh-process wall paired median 增加 10.802722%；当前不能宣称 cold-start speedup	固定 cache 生命周期，拆分 graph build / compile / first execution；不得用 warm KPI 掩盖请求首轮等待
P1	compute_attn_ws 主形状	占隔离 provider 投影 54.742%，主形状占 82.536%；grid=170 CTA 已无 tail wave	只有得到 tensor/stall evidence 才进入 core rewrite；不把 25% theoretical occupancy 当作 free headroom
P1	Attention provider 边界	V-direct-BSH 已全模型验证；Q/K cast 与 direct-layout 仍为单变量候选	每次只改一个边界；four-shape CUDA event 后再跑固定 20 秒 paired request
P2	Native SDPA 的 H128 video-to-text	589.289 ms，占 native core 76.943%	只换 provider 不降精度；该家族是 outlier 统计确认的敏感区
P2	Media write 与 mux	7,424.061 ms 且无 CUDA kernel，在计时口径外但在用户等待内	单独测 request-to-final-MP4 wall；不得以降低编码质量换取生成性能

19.4　仍未完成，且不能在报告中假装完成

- 同 commit、同合同、同 timer 的 BF16 与 FP8 正式性能 A/B。K32 与 tensorwise 之间已有严格配对对照，但 BF16 与 FP8 之间仍然没有，因此本报告不给出官方 BF16→FP8 speedup 倍数。
- 两人独立盲评、calibration freeze marker、validation 的 4 次运行与 heldout-v2；当前状态依次为 PENDING、BLOCKED、LOCKED (0/4) 与 EMBARGOED。
- 完整 NCU 的 tensor-pipe、warp stall、achieved occupancy 与 duration counters；ncu--setnone 已得到 launch、理论 occupancy 限制因子与 wave 数据，但不替代上述 counters。

- 自研 SM120 attention 生产 kernel 的分布级画质证据。被拒的 A3 使用 fixed_p256_two_pass，而生产 kernel 使用 online_dynamic_one_pass，两者不是同一数学策略，因此 A3 的精度拒绝既不能外推，也不能替代生产 kernel 的质量论证。
- AV-sync gate：代理指标标定失败（400 ms 音频延迟未检出），已降级为 diagnostic，需引入 SyncNet 级方法。
- 质量面板 |dev| 阈值尚未经人评校准；面板只能证明“没变坏”，不能证明“变好”。

最终判断

工程路径现在有四个清楚的状态：**历史交付**是 O7 的 tensorwise W8A8 scaled-MM（27.104924 generated FPS）；**历史 K32 engineering evidence** 包括 832 × 480 的 K32 paired A/B、Week 9 V-direct-BSH +1.289408% stable-DiT gain 与 Portrait v1；**当前工程基线**是 Portrait catnip-k32-vdirect-single-tile-compile-portrait-480x832-v2，37.557043 stable-DiT / 34.392314 generated FPS；**未完成项**是 cold-start、blind review、freeze、validation 与 heldout-v2。最终 r03 smoke 已通过，但 formal ground-truth quality acceptance 尚未成立。

精度路径的收敛方式与上一版完全不同。上一版把问题窄化为“如何在 provider 可表示的 scale 集合中保留 K32 局部收益”，而这个问法建立在一个已被证伪的判据上。真正的收敛发生在方法层：先证明旧判据不能验收，再重建可标定的解码媒体面板，然后用它重新审判被旧判据毙掉的候选——K32 正是这样从 **REJECT** 回到工程基线的。后续任何候选都应先问“判据本身能否分辨好坏”，再问“这个候选是否更好”。

历史 profiling 已证明两卡流水接近均衡，并据此推动了本次 GPU0 路径优化。v2 现在把 current warm KPI 提到 34.392314 generated FPS，但该晋升不等于新的 device-separated profiling：后续若要重新判断瓶颈，必须在 v2 上重新测量，不能继续把历史 5–9% crossover 或 99.640% busy 当成当前精确占比。GPU1 的 scaled-mm/attention 候选与 GPU0 的后续工作都必须用 v2 paired full-model gate 验收。

20

独立实验账本：成功、失败与阻断原因

本章刻意与主线分开。主线解释“为什么进入下一步”，实验账本则保证失败不会被成功叙事淹没。**Reject** 表示实验有效、candidate 假设被数据否定；**Blocked** 表示已有有效 gate 结果，但 phase 或后续范围按协议被锁住；**Invalid** 表示基础设施或合同错误，不能产生科学结论；**Not run** 表示门禁正确早停，不是遗漏。

20.1 被保留或提升的关键尝试

表 20.1: 成功链路与保留理由

阶段	尝试	结果与保留理由	状态
Week 2	480-module memory-first FP8	让 chunk 0 通过，暴露 KV/PE lifetime；作为容量诊断中间态	保留证据
Week 2	Bounded KV + shared PE + fused bias	11 chunks 完成，reserved 降到 memory gate 内	保留
Week 2	Host worker async VAE	修复 same-thread false overlap；形成 O0 19.080 FPS	保留
O2	48-block independent compile	O1→O2 observed wall -2,892.619 ms	保留
O4	Fold duplicate V2A	observed wall -2,380.908 ms；保留 cache 副作用与 residual 顺序	保留
O5	576-module coverage	wall -824.946 ms，coverage 72.138575%	保留
O6	PE metadata key + 5-shape warmup	27.059941 FPS，formal graph delta 为空	保留
O7	Strict entry + serial warmup	三轮 27.104924 FPS，CV 0.046536%	交付

阶段	尝试	结果与保留理由	状态
Week 3	16 MiB provider workspace	wall -0.865511%，FPS +0.873067%；已固化进 Week 8 冻结启动环境，6 次 performance 与 8 次 calibration 运行中生效	已固化
Week 6	Exact-FP32 K32 simulator	Local +22.910%，四条 video block-probe lanes 通过；提供机制证据	研究保留
Attention V2	25-shape complete-op timing	全部 ≥ 20 s、production layout、可复算；取代 NVML/V1。该测量对象为 2026-07-20 的全 PyTorch Flash SDPA 状态	历史 Headline
Outlier 统计	576 层全量权重 + 63,360 activation records	INT8 对照证明 outlier 直觉不能移植到 E4M3；权重侧无肉、激活侧有肉	研究保留
Week 7	自研 SM120 MXFP8 attention	48 个 attn1 替换；r04 整机 28.084266773 generated FPS；主形状隔离 1.44236×	生产默认
Week 8	质量面板证伪旧判据	处理组跨 cell 符号翻转、安慰剂优于处理组；判据降级为描述性指标	方法学
Week 8	Reference-free 解码媒体面板	伪影注入标定：flicker 25.68×、click 8.51×、sharpness 1.78×、blockiness 1.13×	新验收权威
Week 8	Native K32 MXFP8 baseline V2	三对交替配对 stable-DiT median gain 14.103774%、最小配对 13.995281%；两 lane 解码媒体 AUTO_PASS	**PROMOTE**
Week 8	Attention stage benchmark	4 组生产 shape、20 warmup + 100 formal、12/12 二进制哈希核验	Historical profiling evidence
Week 9	V-direct-BSH attention boundary	历史 832 × 480 K32 stack 的三对 full-model A/B：stable-DiT median +1.289408%，generated +1.284480%，6/6 execution gates PASS；quality acceptance 未声明	Engineering promote only

阶段	尝试	结果与保留理由	状态
Week 9	Counter-free NCU --setnone	获得 launch、wave 与 theoretical occupancy facts；attention 无 tail wave，P0 GEMM 89.4% wave efficiency	方法学保留
Portrait v1	K32/V-direct Portrait baseline	三次 fresh-cache mean 37.486982 stable-DiT / 33.070761 generated FPS；runtime contracts PASS, quality acceptance=false	历史基线
Portrait v2	Single-tile + compiled video VAE decoder	三对 candidate mean 37.557043 stable-DiT / 34.392314 generated FPS；final r03 smoke PASS；只声明 warm-throughput	当前基线

20.2 有效实验，但假设被拒绝

表 20.2: 科学上有效的 rejected attempts

阶段	候选	不 work 的直接原因	决定
O3	Selective cuDNN SDPA	单 shape 快 19.7%，完整请求却慢 220.053 ms；shape mix/workspace/scheduling 抵消	REJECT
Week 3	External M padding	Square/FFN up/down 分别仅 0.991985×/0.994219×/0.984182×；预计请求 +52.517 ms，未计 pad/crop	REJECT
Week 3	Shape-specific fast-accum	Square micro 1.0237×，full median 反而慢 2.402 ms / 0.013329%	REJECT
Week 3	Shared activation quant	Physical quant calls -25%，但 full wall 仅 -0.166418%，低于 0.25%，且轨迹改变	REJECT
Week 3	Text K/V cache	48 layers×5 states 需 3,840 MiB，直接 OOM；9 layers 720 MiB 也超 128 MiB gate	REJECT
Week 3	Simple QKV/KV concat	未计 split/norm/RoPE/layout 的乐观 request ceiling 仅 0.173474%	REJECT
Week 3	Shared quant + provider	比 provider-only 仅多 9.716 ms / 0.054386%，低于 incremental gate 且 seam 更差	REJECT
Week 4	Rowact/tensorweight	单 cell aggregate 比 tensorwise 更差	REJECT
Week 4	K/V BF16 rescues	K-only、V-only，以及后续成功的 K+V 均未改善 rowwise aggregate	REJECT
Week 4	Hybrid-K	Overall worst -20.320%，audio median 退化	REJECT
Week 4	Tensor act + channel W	Overall worst -22.632%，stable speed worst -2.323%	REJECT
Week 4	Full rowwise	Overall worst -7.563%；audio worst -61.777% / -65.655%	REJECT
Week 4	SmoothQuant	Calibration median +22.782%，heldout median -2.991%、worst -18.750%	REJECT
Week 5	A-row/W-channel A2V	Calibration 为正，tailor validation -12.150% / -10.925%；audio tail 更差	REJECT
Week 5	Sensitive top-48	Drummer overall 正，但 audio -2.299% / -2.456%	REJECT

阶段	候选	不 work 的直接原因	决定
Week 5	Top-96/top-192	首个 calibration cell 分别 -10.586% / -6.134%	REJECT
Week 5	Video FFN row/channel	Chef -21.386% / -26.148%	REJECT
Week 5	K32	Week 5 观测值保留（chef +0.827%、drummer -7.064%）；**其拒绝依据于 2026-07-22 被证伪**，2026-07-22/23 在同栈配对实验中 stable-DiT median gain 14.103774%，解码媒体面板 AUTO_PASS	判据失效后重开并已 PROMOTE
Week 5	Logical K64	Chef +17.009%，drummer -9.357%；不是 compact native K64	REJECT
Week 6	K32 pow2/native E8M0	Local mean rel-L2 比 tensorwise 差 1.056%；scale encoding 抹掉 exact-K32 优势	REJECT
Week 6	A2V hybrid rescue	只保护 q/out 仍为 4/8；audio error 来自更广 graph	REJECT
Week 6	Attention tensorwise + FFN K32	八个 block/modality gates 全部失败，0/8	REJECT

20.3 有效 Gate，但 Phase 或后续范围被阻断

表 20.3: Blocked 与 candidate reject 的层级区别

阶段	有效 gate 结果	为什么被阻断	被锁范围
Week 5 temporal	CUDA overhead 10.19%–20.98%	动态 branch 的开销在视频前已超过门槛	5 个 temporal full videos
Week 6 Phase 3	8 个 video metric blockers	Audio metric 可研究，video registry 尚无 production 决策资格	Production registry、final-heldout
Week 6 Phase 4	四条 video block probes 通过、四条 audio p95 失败；总 gate 4/8	Exact K32 有机制证据，但 propagation admission 未闭合	Phase 5、validation、heldout

20.4 无科学结论的 Invalid / Infrastructure Failures

表 20.4: 必须保留但不得当成候选效果的失败

阶段	失败	为什么无科学结论 / 如何修正
Capacity	240-module 三次 chunk-0 OOM	模型 +KV/PE 下界超卡；没有性能或质量结论，扩 coverage 后重启
Capacity	480-module chunk-1 OOM	Append-before-evict lifetime peak；实现 bounded KV 后重启
Scheduler	Same-thread VAE false overlap	不同 CUDA stream 仍被约 1.25 s Python launch 阻塞；迁入 host worker

阶段	失败	为什么无科学结论 / 如何修正
Conditioning	Structured prompt r01	错把视觉 prompt 用作 audio conditioning；整轮作废
O2	Compile warmup r01 OOM	Allocator reserved-but-unallocated 约 2 GiB；冷 cache + expandable segments
O6	Four-shape warmup r01	Formal chunk 4 新编译 6 graphs、wall 68.435 s；补第五 shape
O7	First cold warmup OOM	RoPE buffer 时 reserved 32.913 GB；改 serial warmup/allocator cleanup
Week 2 profile	两个 compiled profiler runs	一个缺 PYTHONPATH，一个 package invocation 错误；均不进入归因
Week 2 Nsys	NVTX-triggered capture 空报告	触发条件错误；改 delay/duration capture
Week 4	BF16 oracle r01	VAE chunk 6 OOM；r02 成功，仅 r02 作 oracle
Week 4	K+V BF16 rescue r01/r02	两次 GPU1 OOM；这两个 run 没有 candidate 效果结论，后续成功 retry 才进入 Reject
Week 4	Observer FP32 index	超过 2^{24} 的 FP32 linspace 导致 CUDA assert；改 INT64
Week 5	First candidate conditioning	Tensor identity 不一致；结果 invalid，修复后重跑
Week 5	K32/K64 first runner	缺 weight_inverse_scale 接口；修复后独立 retry
Week 6 Phase 0	First finalizer ambiguous	Evidence 记录不完整，不是 provider fail；补全 contract 后重审
Week 6 Phase 1	_CUuuid serialization	12 partial tensors 禁止复用；修 JSON metadata 后重新 publication
Week 6 Phase 2	Venv symlink	解析到 system Python，torch import fail；0 GPU requests
Week 6 Phase 4	Capture attempt 01	请求后对超大 tensor torch.quantile 失败；未发布，流式统计后重跑
Week 6 launcher	/usr/bin/time exit 127	容器无该 binary；在 Python/CUDA 前退出且没有 artifact，不是 provider/GPU failure
Week 6 paired	Modality order mismatch	V1 schema 顺序不一致，停止 publication，按 canonical schema 重跑

阶段	失败	为什么无科学结论 / 如何修正
Week 6 hybrid	11-field vs 19-field schema	Attempt 01 停止，不纳入 gate，升级 schema 后重跑
NCU	两次 ERR_NVGPUCTRPERM	缺 host capability/driver policy；不是 kernel fail，也没有 counter 数据
Attention NVML	100% busy	Layout 不匹配且指标只代表 busy；V2 complete-op 取代
Attention V1/V2 Kineto	Kernel-self / 301%	计时边界错误或不可能值；只保留 provider/name
Portrait v2 smoke r01/r02	并发 Week 10 KV study 占用 GPU，VAE decode 前 OOM	外部资源冲突；没有 candidate FPS 或 VAE 结论，清场后 r03 完整通过

20.5 被协议正确阻断的 Not-Run 项

表 20.5: Not run 不等于实验遗漏

项目	为什么没有运行
Phase I FP8 KV storage POC	用户明确要求不做
Week 3 padding/text cache/simple fusion full videos	Micro/feasibility ceiling 已低于 gate 或容量不可行
Week 5 五个 temporal videos	CUDA overhead 10%–21%，协议前置阻断
Week 5 combination runs	Survivor set 为空，组合输入集合严格为空
Week 5 final-heldout	Calibration + validation 没有 survivor
Week 6 Phase 5 / validation / heldout	Paired propagation 只有 4/8，前置 gate 未通过
Decoded perceptual / AV-sync / blind review	尚无候选取得运行授权

20.6 2026-07-21 之后新增的条目

表 20.6: Week 7–8 的 rejected、invalid 与 blocked 条目

阶段	候选 / 事件	直接原因	决定
Week 7	A3 signed H128 旋转	每个 pooled 类别都改善（overall 2.1063%、audio 最高 8.8755%），但预注册门槛为 overall/input/latent $\geq$ 5%、video velocity $\geq$ 10%；是幅度不够而非回归	REJECT
Week 7	A7 概率 K32	非零下溢从 20.0809% 降到 0.9589%（-95.2250%），但 full-vs-production defect 仅改善 0.0267%，只捕获 exact-P ceiling 的 0.1700%；25 项 gate 中 11 项失败	REJECT
Week 7	P1d learned rotation	odd-chunk screen 上三种变体 mean 仅 +0.246%～+0.623%，门槛 $\geq$ 5%；fit loss 降 7.255% 属明显局部过拟合	REJECT
Week 7	Fused cooperative H128	5 shape 逐 bit 一致、单 cooperative launch 达成，但四个生产 shape median 全部回归 +1.512%～+7.440%	REJECT
Week 7	FA3 K/V warp-release	五 shape bitwise 一致但全部变慢 +0.79%～+2.83%；Nsys 复核 +2.38%/+2.44%	REJECT
Week 8	全局 Q/K single-launch	仅最小 shape 快 17.117%；20 秒加权慢 0.934%，主形状慢 1.731%	REJECT
Week 9	cuBLASLt heuristic top-k pinning	dominant 2340 × 4096 × 4096 bitwise correct，但 paired median +0.058769% < +1.0% gate；按 conjunction 规则 early stop	REJECT
Week 8	全家族 rowwise（ROW192）	面板 worst dev 1.709，seam spike 6.5 对比基线 3.1，为本轮最强伪影信号	REJECT
Week 7	MXFP8 r01 / r02	外部作业占用 GPU，在 provider hook 与 formal timer 之前主动停止；无 FPS	Invalid
Week 7	MXFP8 r03	provider dtype gate 过严，编译 warmup 阶段失败；无 FPS	Invalid
Week 8	BF15 首次 drummer 运行	编译预热阶段被操作者中断，无 run.json、无 probe、无对比；已隔离并标注	Invalid
Week 8	Anatomy capture 首次尝试	与前一运行的 teardown 撞车导致 GPU0 OOM；清理后重跑成功	Invalid
Week 8	面板 av-sync gate	注入 400 ms 音频延迟未被检出，代理指标标定失败	降级 diagnostic
Week 8	第一版面板的自然带	使用了原 final-heldout 两格，构成泄漏；V2 协议重分类为 calibration 并另设 heldout-v2	记录并修正
Week 8	两人独立盲评	公开包与响应验证器已就绪，等待两名独立评审	PENDING
Week 8	Calibration freeze	marker status/quality_calibration.freeze.json 不存在	BLOCKED
Week 8	Validation 4 runs	需 freeze marker 解锁；已完成 0/4	LOCKED
Week 8	Heldout-v2 两格	需 hash-bound 书面解锁	EMBARGOED
NCU	第三次 ERR_NVGPUCTRPERM	宿主机 RmProfilingAdminOnly=1，容器 effective 与 bounding set 均缺 CAP_SYS_ADMIN；CAP_PERFMON 非等价替代	行政阻塞

账本的使用规则

本章现在有两类规则，必须分开使用。

第一类：数据否决，不再回锅。 external M padding、simple QKV/KV concat、text K/V cache（48 层 × 5 states 需 3,840 MiB）、原 dynamic torch.cond 路由、全局 Q/K single-launch、H128 producer 融进 fused core、全家族 rowwise。这些都有直接的测量否证，不应原样重跑。

第二类：判据失效，必须重审。 所有以端到端 trajectory rel-L2 作为 accept/reject 依据的结论——包括 Week 4 与 Week 5 的全部精度 reject、Week 6 的 propagation

4/8 阻断——都建立在一个已被证伪的度量上。它们不自动变成“其实可行”，而是**回到未判定状态**，需要用解码媒体面板重新审判。K32 就是第一个走完这条路径并回到工程基线的候选。
基础设施失败仍必须在修复后用新 run ID 重启；由门禁阻断的 heldout 不得为了“补齐表格”而访问。已经发生过一次 heldout 泄漏（Week 8 第一版面板），其代价是永久损失两格独立性——这是本报告中最应避免重演的错误。

A

构建与复现入口

A.1 重建图表与 PDF

报告图表全部从 `data/report_metrics.json` 生成，不解析 Markdown 表格。构建脚本先运行 Matplotlib，再用 XeLaTeX 两遍以上解析目录、图表和引用：

```
cd /workspace/catnip-technical-report-latex
./build.sh
```

等价的展开命令：

```
MPLBACKEND=Agg PYTHONDONTWRITEBYTECODE=1 \
  /workspace/LTX/.venv/bin/python scripts/generate_figures.py

LC_ALL=C.UTF-8 TZ=UTC SOURCE_DATE_EPOCH=1784592000 \
  latexmk -xelatex -interaction=nonstopmode -halt-on-error \
  -file-line-error -outdir=build main.tex
```

构建不使用 -shell-escape，代码块不依赖 minted。最终 PDF、编译 log、file list 与 SHA256 保留在报告目录和 `build/`。

A.2 当前一键推理：Portrait v2

当前默认入口解析 `results/CURRENT\_BASELINE.json` 并固定到 v2 runner/config；默认合同是 480×832、20 秒、489 帧与 first 4 + latest 6 LF KV：

```
/workspace/LTX/.venv/bin/python \
  /workspace/catnip-portrait-baseline/run_baseline.py
```

版本化入口与冻结配置分别为 `/workspace/catnip-portrait-baseline/run\_portrait\_baseline\_v2.py` 与 `configs/portrait\_baseline\_v2.json`。当前指针和 v2 manifest 绑定其 SHA256；结果目录由 runner 以 create-only 语义创建。该入口包含已接受的 single-tile latent-window decode 与 compiled video VAE decoder，只声明 warm steady-state performance。

A.3 历史一键推理：Week 8 K32 基线

Week 8 的自包含 runtime 保留为历史 832×480 K32 复现入口。它不从 Week 2/5/7 的实验目录 import 文件；模型权重与 Python 环境仍在 `/workspace/LTX`。三个阶段共用同一个 result 目录，且 result 路径永不覆盖：

```
cd /workspace/week8-k32-baseline
RESULT="results/$(date -u +%Y%m%d_%H%M%S)_832x480_20s_k32"

CUDA_VISIBLE_DEVICES=0,1 /workspace/LTX/.venv/bin/python scripts/run_k32.py \
  --preflight-only --prompt-file configs/default_experiment_prompt.txt \
  --result-dir "$RESULT"

CUDA_VISIBLE_DEVICES=0,1 /workspace/LTX/.venv/bin/python scripts/run_k32.py \
  --dry-run --prompt-file configs/default_experiment_prompt.txt \
  --result-dir "$RESULT"

CUDA_VISIBLE_DEVICES=0,1 /workspace/LTX/.venv/bin/python scripts/run_k32.py \
  --prompt-file configs/default_experiment_prompt.txt --result-dir "$RESULT"

/workspace/LTX/.venv/bin/python scripts/validate_k32.py --result-dir "$RESULT"
/workspace/LTX/.venv/bin/python scripts/validate_baseline_reference.py \
  bind-result --result-dir "$RESULT"
```

preflight 与 dry-run 都是严格阶段：解析结构化 prompt，要求 result/staging/output 路径未被占用且可写，校验 cache policy，全量哈希两个 checkpoint 与 16 个 Gemma runtime 文件（共 24,414,137,010 字节），哈希冻结的 runtime artifacts，并在隔离子进程中查询 Torch/CUDA。两者都不把 Torch import 进父 launcher，不启动推理，不创建 result 或 cache。

CUBLASLT_WORKSPACE_SIZE=16384 已固化进该 runtime 的冻结启动环境，不再需要调用者手工导出。--cache-policywarm 是吞吐基线并复用已编译产物；--cache-policyfresh 要求 cache root 不存在，用于测量冷编译（attempt06 的 fresh compile warmup 为 305.712944 秒，cache 从 0 文件增长到 1,756 个 Inductor 文件、599,228,407 字节）。

校验发布身份本身，以及严格重放某个 result 的 baseline reference：

```
CUDA_VISIBLE_DEVICES=0,1 /workspace/LTX/.venv/bin/python \
  scripts/validate_baseline_reference.py validate-release

CUDA_VISIBLE_DEVICES=0,1 /workspace/LTX/.venv/bin/python \
  scripts/validate_baseline_reference.py validate-reference --result-dir "$RESULT"
```

后续候选不应就地替换该 release，而应使用 `scripts/run_paired_candidate.py` 绑定精确的 K32 release 作为控制组，按 control/candidate、candidate/control、con-

trol/candidate 顺序执行三对隔离的 20 秒运行，最后由 `scripts/validate_paired_candidate.py` 独立重放并拒绝身份、顺序、环境、cache 或 artifact 漂移。

A.4 历史 O7 交付包复现入口

Week 2 的一键入口保留为历史交付复现路径，其结果对应 27.104924 generated FPS，不代表当前基线：

```
cd /workspace/week2-improved-fp8_scaled

export CATNIP_CHECKPOINT=/workspace/LTX/models/ltx_rl16/merged_sst1500_rl16.safetensors
export LTX_BASE_CHECKPOINT=/workspace/LTX/models/LTX-2.3/ltx-2.3-22b-dev.safetensors
export GEMMA_TEXT_ENCODER=/workspace/LTX/models/gemma-3-12b-it-qat-q4_0-unquantized
export CUBLASLT_WORKSPACE_SIZE=16384

./run_inference.sh \
  --prompt-file "$PWD/experiments/split_fp8_dit_vae_profile/configs/prompt_person_car_20s.txt
    "
```

在该历史入口中，CUBLASLT_WORKSPACE_SIZE 仍必须由调用者在首次 CUDA 初始化前设置。--dry-run 会检查 canonical contract、prompt、frozen noise、FP8 manifest、4+6 KV、geometry 与输出冲突，但不会加载模型资产。

A.5 Week 3 Profiling 离线复算

```
cd /workspace/week3-fp8-gemm-optimization

/workspace/LTX/.venv/bin/python scripts/analyze_final_nsys.py \
  --json-out /workspace/technical-report-evidence/week3-reanalysis/nsys_analysis.json \
  --markdown-out /workspace/technical-report-evidence/week3-reanalysis/nsys_analysis.md

/workspace/LTX/.venv/bin/python scripts/analyze_final_kineto.py \
  --json-output /workspace/technical-report-evidence/week3-reanalysis/kineto_analysis.json \
  --markdown-output /workspace/technical-report-evidence/week3-reanalysis/kineto_analysis.md
```

两条命令只读打开现存 trace，不占用 GPU；输出时间戳可能变化，核心数值与 checksum 必须一致。

A.6 Attention V2 完整复算

```
cd /workspace/technical-report-evidence/attention-fa-util-20260720
```

```
python3 analyze_fa_full_mix.py \
  --summary results/run_20260720_fa_full_mix_r01/summary.json \
  --telemetry results/run_20260720_fa_full_mix_r01/telemetry.csv \
  --inventory production_nsys_inventory.json \
  --output /tmp/catnip_fa_full_mix_recomputed.json

cmp /tmp/catnip_fa_full_mix_recomputed.json \
  results/run_20260720_fa_full_mix_r01/analysis.json

sha256sum -c SHA256SUMS
```

Byte-identical JSON 复算使用 workspace system Python 3.12.3；不同 Python minor version 可能只在浮点最后一位出现序列化差异。重新启动正式 GPU 矩阵必须使用新的唯一 run ID：

```
bash run_full_mix_formal.sh NEW_UNIQUE_RUN_ID
```

该 launcher 拒绝覆盖旧结果，并强制每个 shape 运行 20 秒。

B

证据索引、引用与限制

B.1 内部证据映射

表 B.1: 正文 Evidence ID 对应的权威文件或目录

Evidence ID	Grade	路径 / 说明
E-MASTER	A/B	`/workspace/CATNIP_TECHNICAL_REPORT.md`；原综合 Markdown 报告
E-W1-RUNBOOK	B	`/workspace/week2-improved-fp8_scaled/docs/runbooks/catnip_quantization_execution_runbook.md`
E-W2-RUNBOOK	B	`/workspace/week2-improved-fp8_scaled/docs/runbooks/catnip_bf16_vs_fp8_scaled_mm_runbook.md`
E-W2-HOW	B	`/workspace/week2-improved-fp8_scaled/HOW_WE_REACHED_27FPS.md`
E-W2-RUN	A	`/workspace/week2-improved-fp8_scaled/results/fp8_scaled/run_20260715T123627Z_1720695`
E-W3-FINAL	A	`/workspace/week3-fp8-gemm-optimization/docs/FINAL_TECHNICAL_REPORT.md`
E-W3-NSYS	A	`/workspace/technical-report-evidence/week3-reanalysis/nsys_analysis.json` 与原 trace
E-W3-KINETO	A	`/workspace/technical-report-evidence/week3-reanalysis/kineto_analysis.json`
E-W4-FINAL	A	`/workspace/week4-fp8-accuracy/FINAL_FP8_ACCURACY_REPORT.md`
E-W4-GATES	A	`/workspace/week4-fp8-accuracy/results/gate_d/accuracy_matrix_summary.json`; `gate_e/gate_e_results_summary.json`; `gate_f_rowwise/gate_f_rowwise_results_summary.json`; `smoothquant_heldout_gate/gate_report.json`
E-W5-FINAL	A	`/workspace/week5-fp8-scale-allocation/FINAL_SCALE_ALLOCATION_REPORT.md`

Evidence ID	Grade	路径 / 说明
E-W5-SUMMARY	A	`/workspace/week5-fp8-scale-allocation/results/preselection_final_summary.json` 与 `dynamic_cond_cuda_gate.json`
E-W6-LOCAL	A	`/workspace/week6-streaming-av-quant-research/PHASE4_LOCAL_FACTOR_RESULTS.md`；`results/phase4_factorization/grid/local_factor_summaries.json`
E-W6-PAIRED	A	`/workspace/week6-streaming-av-quant-research/PHASE4_PAIRED_PROPAGATION_RESULTS.md`；`paired_block/paired_block_summaries.json`
E-W6-AUDIO	A	`/workspace/week6-streaming-av-quant-research/PHASE4_AUDIO_TAIL_RETROSPECTIVE_RESULTS.md`
E-W6-RESCUE	A	`/workspace/week6-streaming-av-quant-research/PHASE4_HYBRID_A2V_RESCUE_RESULTS.md` 与 `PHASE4_QUICK_ATTENTION_FFN_K32_RESULTS.md`
E-BF16-FRESH	A/C	`/workspace/week6-streaming-av-quant-research/results/phase2_reference`
E-ATTN-NCU	A	`/workspace/technical-report-evidence/attention-ncu-20260720`
E-ATTN-V2	A	`/workspace/technical-report-evidence/attention-fa-util-20260720`；含 pre-run/post-run manifests、raw、analysis 与 SQLite extractor
E-FP8OUT	A	`/workspace/catnip-fp8-outlier-statistics`；`report/main.pdf`、`data/processed/analysis_summary.json`、`int8_control_summary.json`、11 张图与 5 张表
E-W7-SM120	A	`/workspace/week7-blackwell-fp8-attention`；`sm120_fp8_attention/FULL_20S_FPS_REPORT.md`、`FUSED_H128_COOPERATIVE_REPORT.md`、`P1B_MODEL_LOOP_REPORT.md`、`P1C_A7_PROBABILITY_K32_REPORT.md`、`P1D_LEARNED_ORTHOGONAL_ROTATION_REPORT.md`
E-W8-PANEL	A	`/workspace/week8-bf15-causal-lanes`；`P0_REPORT_20260722.md`、`configs/layer_sets.json`、`results/panel/adjudication.json`、`results/anatomy/*/anatomy_analysis.json`

Evidence ID	Grade	路径 / 说明
E-W8-K32	A	`/workspace/week8-k32-baseline`; `performance/summary.json`、`status/stage_status.json`、`status/baselines/k32_reusable_experiment_v1.json`、`scores/calibration_current/`、`references/calibration_band_v2.json`、`reports/K32_V2_EXPERIMENT_LOG.md`
E-W8-PROF	A	`/workspace/week8-k32-profiling`; `DETAILED_PROFILING_REPORT.md`、`ATTENTION_KERNEL_SOURCE_AUDIT.md`、`GPU1_KERNEL_SHAPE_AUDIT.md`、`SCALED_MM_NCU_HARNESS.md`、`results/ncu_permission/`
E-W9	A	`/workspace/week9-attention-boundary-engineering`; V-direct full-model A/B、scaled-MM rejection 与 counter-free NCU findings
E-PORTRAIT	A	`/workspace/catnip-portrait-baseline/results/BASELINE_MANIFEST.json`、`PERFORMANCE_SUMMARY.json` 与三个 v1 performance gates；历史 v1 证据
E-PORTRAIT-V2	A	`/workspace/catnip-portrait-baseline/results/CURRENT_BASELINE.json`、`v2/BASELINE_MANIFEST_V2.json`、`PERFORMANCE_SUMMARY_V2.json`、`/workspace/catnip-portrait-baseline/BASELINE_V2_RESULT.md` 与 final r03 smoke gates

B.2 外部技术参考

参考文献

[1] Jay Shah, Ganesh Bikshandi, Ying Zhang, Vijay Thakkar, Pradeep Ramani, and Tri Dao. *FlashAttention-3: Fast and Accurate Attention with Asynchrony and Low-precision*. arXiv:2407.08608, 2024. https://arxiv.org/abs/2407.08608.

[2] Ted Zadouri, Markus Hoehnerbach, Jay Shah, Timmy Liu, Vijay Thakkar, and Tri Dao. *FlashAttention-4: Algorithm and Kernel Pipelining Co-Design for Asymmetric Hardware Scaling*. arXiv:2603.05451, 2026. https://arxiv.org/abs/2603.05451.

[3] NVIDIA. *NVIDIA RTX Blackwell GPU Architecture*. https://images.nvidia.com/aem-dam/Solutions/geforce/blackwell/nvidia-rtx-blackwell-gpu-architecture.pdf.

[4] PyTorch. *torch.nn.functional.scaled_dot_product_attention*. https://docs.pytorch.org/docs/main/generated/torch.nn.functional.scaled_dot_product_attention.html.

[5] NVIDIA. *ERR_NVGPUCTRPERM: Permission issue with Performance Counters*. https://developer.nvidia.com/ERR_NVGPUCTRPERM.

B.3 报告限制

1. Week 1 与 Week 2 O0–O7 的部分 raw 已清理，历史 waterfall 属于等级 B；它不是严格 factorial A/B。
2. Week 3 profiling、Week 4–6 gates 与 Attention V2 有原始/结构化 evidence；但 NCU hardware counters 仍缺权限。
3. 27.104924 / 27.370839 / 32.572402 都是 formal generated throughput（489 除以 sampling 加流水 video decode 的秒数），不是冷启动 shell E2E，也不是 25 FPS playback。
4. Week 8 首次量化了计时器之外的阶段：text encoder 801.745 ms、condition copy 6.417 ms、audio decode 313.123 ms（其中嵌套 VAE decode 311.391 ms）、media write/encode 7,424.061 ms 且该 NVTX 范围内没有 CUDA kernel。它们都不在 generated FPS 的分母内。
5. Nsys 插桩会使 stable-DiT 与 generated 分别下降约 1.533% 与 2.040%；插桩运行不得用于对外性能口径。

6. 168.677600 TFLOP/s 描述的是 2026-07-20 全 PyTorch Flash SDPA 的 attention，不描述当前 48 个 video-self attn1 的自研 SM120 MXFP8 provider。
7. Fixed-seed deterministic 不等于 BF16 equivalence；媒体 hash 只能证明候选自身稳定。
8. Local relative-L2、propagated block error 与 decoded perceptual quality 不可互相替代。
9. Final-heldout 未访问是协议要求；没有 survivor 时补跑 heldout 会污染实验设计。
10. 16 MiB workspace 对 PyTorch/CUDA/cuBLAS 与 RTX 5090 provider 版本敏感；升级栈后必须重新验收。
11. FA3/FA4 的 75%/71% headline 来自 H100/B200 的最佳 shape；不能与 RTX 5090 的 production mix 直接排名。
12. 当前 Portrait v2 的 34.392314 generated FPS 是 warm steady-state 三对 candidate mean。fresh-process wall paired median 增加 10.802722%，所以没有 cold-start speedup claim；已有 VAE 质量对比也不构成 ground-truth 或 blind-panel acceptance。