

岩芯矩阵

Ultra-Fast Token

从 KV Cache 不增长，到单用户百万 token/s

World-State EdgeRock DiffusionRock SiliconRock



技术摘要

不压缩 KV Cache，而是让它根本不增长；推理再长，内存不涨。

摘要. 长推理正在让内存成为部署的瓶颈。今天的大模型会把推理过程中的每一步都记进 KV Cache，因此推理越长，占用的显存就越多，一张卡能同时服务的请求也就越少。到了机器人、汽车这样的端侧设备上，这份不断膨胀的记忆还要和感知、控制争抢本就有限的内存。业界常见的办法是压缩，让每个 token 存得更小、读得更少。然而，只要缓存还在随推理增长，压缩就只是把内存墙往后推，并没有真正拆掉它。

岩芯矩阵（Rock Matrix）换了一个思路：与其压缩 KV Cache，不如让它根本不增长。为此，我们提出了世界状态（World-State）。模型不再保存一份越写越长的完整历史，而是维护一份容量固定的状态，用它概括此刻的世界，并随着新信息的到来不断更新。这样一来，推理再长，内存也不会上涨；内存不再取决于输入有多长，而是在设计时就已确定。

当然，固定容量的记忆并不新鲜，真正的难点在于，记忆一旦固定，模型往往会忘掉远处的关键细节，推理能力也随之下降。为此，我们原创了记忆岩锚（Rock Memory Anchors, RMA）：模型自己学会判断哪些细节值得长期保留，并像攀岩者钉下岩锚一样把它们固定住，无论来源多远，后面的推理都能直接读到。正因为如此，首个开源模型 EdgeRock-4B 在把注意力 KV 砍掉 99.1% 的同时，推理能力依然与领先的小模型处于同一水平。而且，每条请求的内存固定以后，同一张 GPU 能同时服务的请求多了，长推理吞吐随之提升到来源模型的 6.42 倍。

世界状态的终极形态，是一个全模态、any-to-any 的模型：它不仅能读懂文字、图像、视频、声音和传感器信号，也能把它们生成出来。其中，理解依靠世界状态持续积累，生成则交给 DiffusionRock，它一次生成一整块内容，而不是一个 token 接着一个 token 地往外写。

要让这样的模型真正快起来，还要解决推理的核心问题：搬运。今天，每生成一个 token，处理器都要把整套权重和不断变长的 KV Cache 从显存里重新读一遍，算力大多花在等数据上。SiliconRock 从根本上去掉了这两项搬运：权重被完全固化成芯片里的计算电路，生成时一个字节也不必搬运；世界状态又让 KV 的容量在设计时就已确定，可以整体留在片上。这样一来，解码路径上只剩下必要的计算，单用户百万 token/s 才有了实现的基础。三条产品线最终指向同一个目标，也就是 Ultra-Fast Token：让强模型持续地高速生成，既帮助 Agent 更快地完成任务，也让机器人跟上物理世界的节奏。

技术支点	
起点	KV Cache 不增长，推理再长，内存不涨。
核心	世界状态（World-State）。
原创	记忆岩锚（RMA），只记得记住的。
终极形态	全模态 any-to-any 模型，既理解也生成。
目标	Ultra-Fast Token，单用户百万 token/s。

核心主张

强：恒定内存下的强推理，推理能力不打折。

快：KV Cache 不增长，单卡吞吐最高 6.42 倍；权重完全固化，目标单用户百万 token/s。

产品交付形态	
EdgeRock：	恒定内存下的强推理模型，端侧与数据中心通用。
DiffusionRock：	any-to-any 模型的生成部分，一次生成一整块内容。
SiliconRock：	权重完全固化、状态留在片上，解码不再搬运权重。

验证口径

延迟：首候选动作时间和传感器到动作的总闭环时间。

内存 / 能效：连续运行内存曲线、上下文增长斜率和单位功耗吞吐。

引言

模型正在从回答问题走向完成任务。推理模型动辄思考几万 token，Agent 一跑就是几十轮工具调用，机器人要让感知与推理在同一个循环里持续运行。智能能进入多少产品，越来越取决于两件事：生成有多快，记忆能撑多久。

今天的大多数模型把推理历史存在 KV Cache 里，每生成一个 token 就多存一份。推理越长，占的显存越多，一张卡能同时服务的请求越少；到了端侧，KV Cache 还要和感知、控制争抢有限的内存；每个新 token 都要回头读一遍越积越多的历史，越写越慢。压缩、稀疏、量化和混合架构降低了每个 token 的成本，但缓存仍随推理增长：放慢增长，只是把内存墙推远。

岩芯矩阵的回答更彻底：不压缩 KV Cache，而是让它根本不增长。我们用世界状态（World-State）承载模型的全部历史，在恒定内存里留住推理需要的关键细节，再把这份确定性转化为速度：从 EdgeRock 的恒定内存模型，到 DiffusionRock 的整块生成，再到 SiliconRock 的模型专用芯片，目标是单用户百万 token/s 的 Ultra-Fast Token。

本文先概览岩芯矩阵的产品体系与三条设计原则，再依次介绍 EdgeRock、性能、DiffusionRock 与 SiliconRock，最后给出应用、部署与路线图。

岩芯矩阵概览

岩芯矩阵的目标是 Ultra-Fast Token：让强模型以单用户百万 token/s 的速度持续生成。实现它的核心技术是世界状态：模型携带的全部历史放进一份容量固定的状态里，KV Cache 不随推理增长。沿着这条路走下去，终点是一个全模态、any-to-any 的模型，它既能理解，也能生成。三条产品线因此各司其职：EdgeRock 负责在恒定内存下理解与推理，DiffusionRock 负责整块生成，SiliconRock 则把状态与权重都固定的模型做进芯片。

要点

- **不增长**：不压缩 KV Cache，而是让它根本不增长；推理再长，内存不涨。
- **强**：恒定内存下的强推理，推理能力不打折。
- **快**：单卡长推理吞吐最高 6 倍以上；目标单用户百万 token/s。
- **端云通用**：端侧记忆预算固定，数据中心按请求数规划容量。
- **三位一体**：EdgeRock、DiffusionRock、SiliconRock 共享同一份世界状态。
- **已开源**：EdgeRock-4B 的权重与推理代码已经发布。

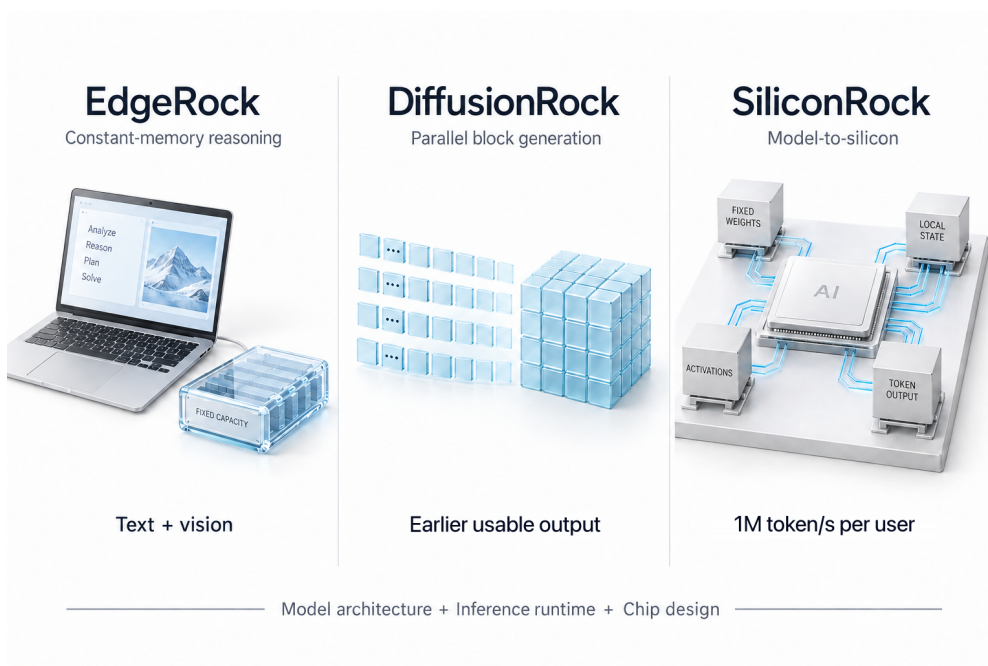


图 1. 岩芯矩阵的三个方向：EdgeRock 恒定内存，DiffusionRock 整块生成，SiliconRock 固化进芯片，一起实现 Ultra-Fast Token。

设计原则

岩芯矩阵的技术路线建立在三条设计原则上。每一条都与今天的主流做法不同。

原则一：KV Cache 不增长

放慢增长，只是把内存墙推远；我们要的是不增长。

主流做法给每个缓存 token 瘦身：多头潜在注意力压缩每个缓存项，稀疏注意力只读一部分，量化降低精度，Qwen3.5、Kimi Linear 这类混合架构只在四分之一的层保留全注意力。它们降低了单个 token 的成本，缓存仍随每个 token 增长。岩芯矩阵从结构上取消增长：模型的历史放进固定容量的状态，内存在设计时确定一次，4K 和 256K 的推理用同一份内存。部署需要的是确定性，而不仅仅是效率。

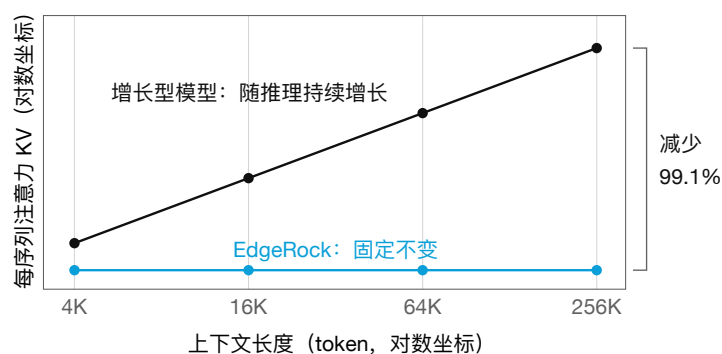


图 2. EdgeRock 的 KV Cache 不随上下文增长：在 256K 上下文处，比来源模型少 99.1%（示意，双对数坐标）。

原则二：只记值得记住的

恒定内存，也要留住远处的关键细节。

有界状态模型同样固定了内存，但它们把历史压成一份模糊的概要，或在窗口外直接遗忘，留不住后续推理要用的远处细节；我们评测的 16 个有界状态模型，最大到 14B，AIME 2024–2026 最高只有 18.8%。世界状态把历史分成三部分：递推状态抓大意，局部窗口管眼前，锚点库记关键。前两部分沿用业界成熟的 Gated DeltaNet 与滑动窗口注意力；真正让世界状态留住远处细节的，是我们原创的记忆岩锚（RMA）。World-State Token 就是这份状态里的记录：数量固定，内容由模型学习挑选；每条都标有来源位置，模型记住了什么可以直接查看。

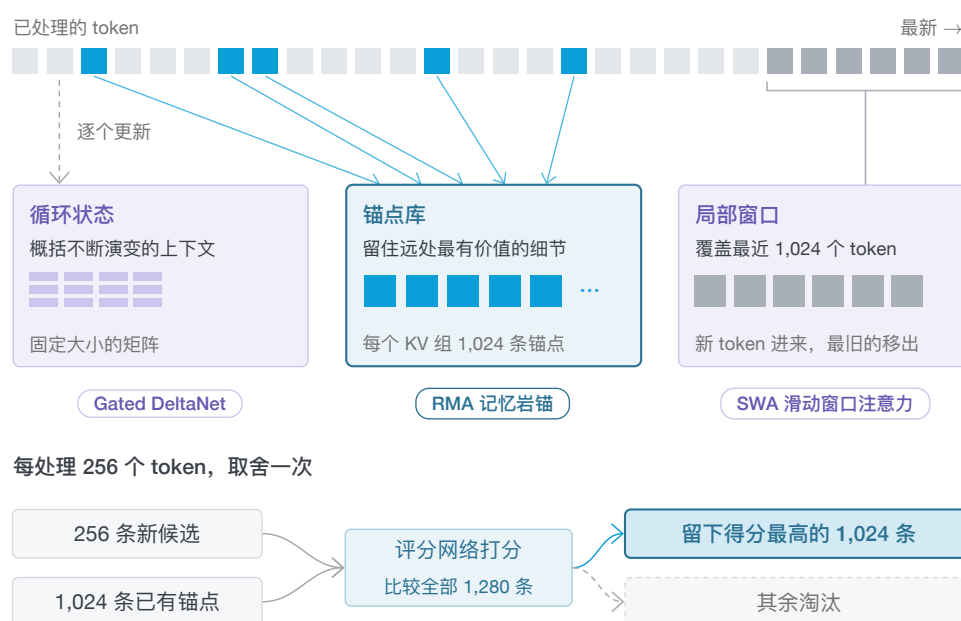


图 3. 世界状态的三部分：循环状态抓大意，局部窗口管眼前，原创的记忆岩锚（RMA）记关键；每处理 256 个 token 取舍一次，只留下得分最高的 1,024 条。

原则三：存算联动，模型与芯片一起设计

省下的内存，直接变成速度。

生成受显存带宽限制，存和算从来是联动的：历史越长，每个 token 要读的越多，能并发的请求越少。世界状态让每个 token 的工作量有上限，让同一张卡并发更多请求；芯片也能按固定容量提前安排片上存储。岩芯矩阵同时设计模型、生成算法和芯片：模型决定保存什么，芯片决定如何高效完成。



图 4. 存算联动：KV Cache 不增长，同时换来不降速的单请求、更高的吞吐，以及可以做进芯片的模型。

EdgeRock：恒定内存下的强推理

像岩锚把结构钉在基岩上，记忆岩锚把关键细节钉在原地。

EdgeRock 是首个落地世界状态的模型系列。首个开源版本 EdgeRock-4B 由 Qwen3.5-4B 转换而来，无需从头预训练：来源模型的 24 层 Gated DeltaNet 递推层保留下来，承担递推状态；其余 8 层全注意力换成 1,024 位置的滑动窗口注意力（SWA），外加独立的 Rock Memory Anchors（RMA，记忆岩锚）模块。转换分四个阶段：先单独训练 RMA 学会保留和读取，再与主干联合训练，然后做监督后训练，最后用可验证奖励的强化学习打磨推理。

记忆岩锚怎么决定记什么？写入器为每个 token 生成一条候选记录，评分网络给它打分；每攒够 256 条候选，每个 KV 分组就把它们和库里已有的记录放在一起比较，只留得分最高的 1,024 条，其余淘汰。留下的记录彼此独立、内容不变，即使来源早已滑出局部窗口，后面的 token 仍能直接读到。

省下内存，推理不打折：恒定内存与强推理可以兼得。EdgeRock-4B 在 AIME 2024-2026 上平均一次通过率 82.9%，与内存随推理增长的领先小模型处于同一水平，是有界状态模型最好成绩的四倍多。锚点正是这份能力的来源：推理时把 RMA 全部关掉，长上下文上的远距查询正确率从 94.1% 跌到 0.4%。

端侧只是 EdgeRock 的一个应用。在端侧，历史内存不随任务推进而增加，感知、规划和控制不必再与模型争抢内存；在数据中心，每条请求的历史固定，同一张卡承载更长的任务和更多并发请求。EdgeRock 从文本与图像起步，沿同一世界状态核心扩展到视频、音频与动作。

性能

推理越长，不再越贵。

世界状态把内存的确定性直接转化为服务收益。下表汇总 EdgeRock-4B 的关键结果，测试条件随表列出；完整评测见 EdgeRock 技术报告（arXiv:2610.10118）。

指标	EdgeRock-4B	对照	条件
注意力 KV	固定，不随长度增长	256K 时比 Qwen3.5-4B 少 99.1%	BF16，按模型配置计算
端到端吞吐	12,124 token/s, 512 路并发	Qwen3.5-4B 1,889 token/s, 6.42 倍	32K 输入、32K 输出；单张 Hopper 80GB；各取最快批大小；含 prefill
平均吞吐	来源模型的 4.94 倍	4K 至 32K 四种输入长度	同上
单请求解码	128K 输出全程 233.7 token/s	Qwen3.5-4B 151.3 token/s, 1.54 倍	128K 输入、128K 输出；批大小 1
推理能力	AIME 平均 82.9%	有界状态模型最高 18.8%	AIME 2024–2026 平均一次通过率

吞吐翻几倍，就是成本降几倍；上下文越长，差距越大。单卡并发 512 路长推理时，EdgeRock-4B 的显存峰值约 61 GiB，含模型权重；如果让来源模型也同时跑完这 512 条序列，光是 KV Cache 就要约 1 TiB，相当于 13 张 80GB 显卡。EdgeRock 一张卡就装下了。

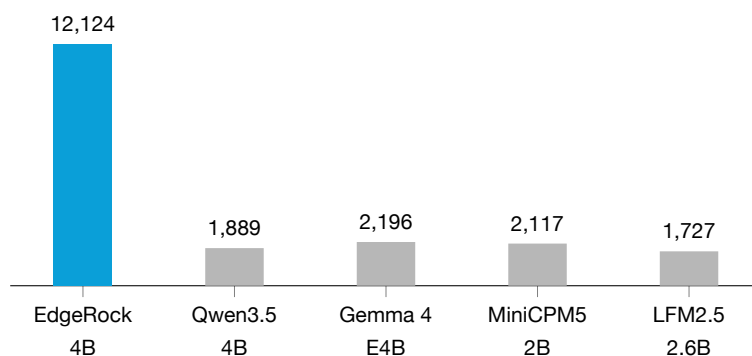


图 5. 同一张 GPU，EdgeRock-4B 的长推理吞吐是来源模型的 6.42 倍（32K 输入、32K 输出，各模型取实测最快批大小，计时含 prefill；单位 token/s）。

DiffusionRock：整块生成

更早得到一段可用代码、一份完整计划。

any-to-any 模型不仅要理解，还要把文字、代码、图像和动作生成出来，DiffusionRock 就是它的生成部分。自回归生成每个 token 都要完整算一遍模型，生成 N 个 token 就是 N 步串行；单用户解码时，GPU 的瓶颈在读取而不在计算，大量算力闲着。块级扩散一次提出一整块内容，例如一段代码、一组工具调用或一段动作，再用少量几轮去噪集中修正不确定的位置，简单的块可以提前停止。修正轮数少于块长，串行步数就少于 token 数，闲置的算力也被用上。

DiffusionRock 的增量在于与世界状态的结合。现有扩散模型，包括 Cosmos 3 的生成部分，每一步都要读取不断增长的上下文；DiffusionRock 以世界状态为条件，任务再长，每一轮去噪读取的历史量都不变。目前，DiffusionRock 的推理速度已达到每秒 1,500 token；下一步将开源模型与推理代码，并让块内的双向修正与因果的状态更新无缝衔接。

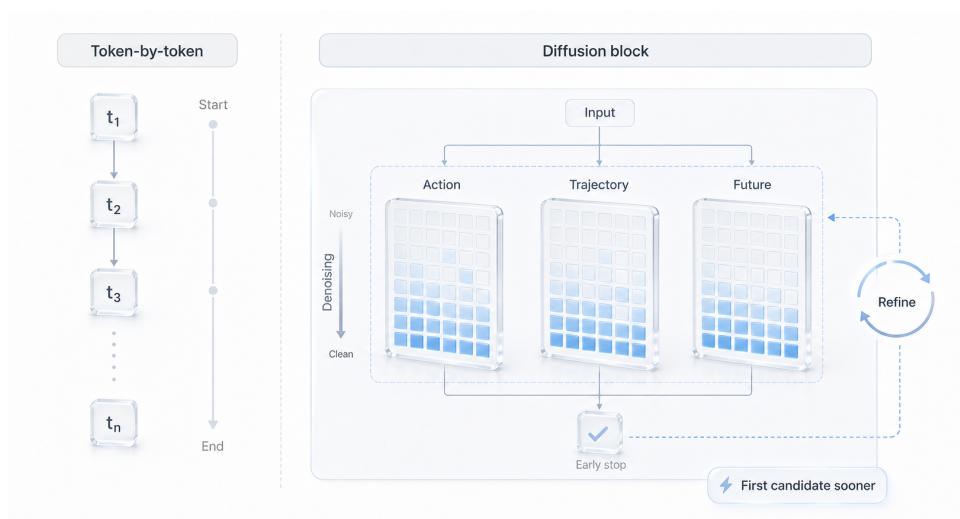


图 6. DiffusionRock 一次生成一整块，并行去噪、修正，更早得到第一个可用结果。

SiliconRock：把模型完全固化进芯片

推理卡在搬运上；权重做成电路、状态留在片上，搬运就消失了。

推理的核心问题：搬运

今天的模型推理，真正卡住速度的往往不是计算，而是搬运。单用户解码时，每生成一个 token，处理器都要把整套模型权重从显存里重新读一遍，还要再读一遍越来越长的 KV Cache；于是，计算单元的大部分时间都花在等数据上。以我们早期原型使用的约 21 亿参数模型为例，即使权重已经量化到 4 位，每生成一个 token 仍要搬运约 913 MB 的权重。如果要做到每秒一百万 token，就需要每秒 913 TB 的带宽，这远远超出了任何存储系统的能力。换句话说，只要权重还需要在运行时搬运，单用户百万 token/s 就无从谈起。

彻底不搬运：权重变成电路，状态留在片上

SiliconRock 的回答是彻底不搬运。首先，4 位权重和缩放系数被直接做成固定的计算路径：权重不再是需要读取的数据，而是电路本身，生成时一个字节也不必搬运。其次，世界状态让 KV 的容量在设计时就已确定，因此每一层都可以把自己的状态放进就近的片上存储；在原型中，每层配有 1 MiB 的状态存储，而且不会随上下文增长。这样一来，解码路径上流动的只剩下 8 位激活、局部状态和输出候选。就连词表输出这一步也做了就地筛选：输出层被切成 116 个分片，每个分片先选出 8 个候选，再经合并树汇总，每个 token 需要跨片传输的数据因此从 472,576 字节降到 7,424 字节，减少了 64 倍。

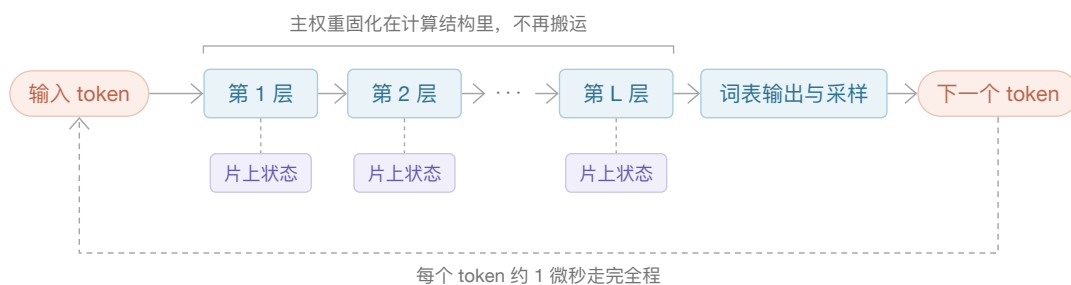


图 7. SiliconRock 的解码路径：主权重固化在每一层的计算结构里，状态留在片上，每个 token 约 1 微秒走完全程。

百万 token/s 的周期预算

单用户解码是串行的，下一个 token 必须等上一个 token 算完才能开始，所以速度只能从单个 token 的时间里一点点抠出来。百万 token/s 意味着每个 token 只有 1 微秒，在 1 GHz 时钟下大约是 1,000 个周期。我们在早期原型上把这笔预算逐项排满：每一层安排一个 30 周期的融合窗口，固定系数的乘加、局部归约、归一化与门控、动态重量化，以及 16 周期的状态扫描，都在这个窗口内重叠完成；词表输出需要 29 个周期，局部筛选与全局合并需要 9 个周期，采样需要 2 个周期，反馈需要 4 个周期。合计起来就是

$$N_{\text{token}} = 28 \times 30 + 29 + 9 + 2 + 4 = 884 \text{ 周期.}$$

在 1 GHz 的设计时钟下，这相当于每秒约 113 万 token；而百万 token/s 的目标允许 952 个周期，因此还留有 68 个周期的余量。需要说明的是，这一数字来自架构建模，而不是芯片实测。

原型进展

我们已经在早期的恒定内存模型上完成了宏级原型：完整模型的固定权重被映射成电路，从模型到芯片版图的全流程已经走通，固定权重电路的正确性也通过了测试验证。下一步，我们将把 EdgeRock 的局部窗口与锚点读取纳入调度，完成完整模型的低精度质量闭合，并在商用工艺上实现。

比固化现成模型多走一步

据 Taalas 公布，它已经把 Llama 3.1 8B 固化进芯片，单用户速度约每秒 1.7 万 token，证明了模型专用硅的速度潜力。不过，它固化的是现成的全注意力模型，KV 状态仍随上下文增长，搬运只去掉了一半。岩芯矩阵能先改模型、再固化，权重和状态两项搬运因此都被去掉，这正是比固化现成模型多走的一步。与此同时，固化并不等于僵化：主权重和主计算图固化在芯片里，LoRA、行业词表、动作域校准和安全阈值则留在可更新的外壳中，行业适配仍然可以持续进行。SiliconRock 承接能力稳定、调用量大的成熟模型，快速迭代的模型则继续在 GPU 上服务，两者分工互补。

三位一体：以 Cosmos 3 为例

三条产品线，是对同一个模型的三处改造。

NVIDIA 发布的 Cosmos 3 由两部分组成：自回归的理解部分处理语言和视觉，扩散的生成部分输出视频、音频和动作；生成部分每一步都要读取理解部分的全部历史。理解部分的记忆一旦增长，两部分的内存和计算都跟着增长。三处改造正好对应：理解部分改造成 EdgeRock 的世界状态，两部分读取的历史都随之固定；生成部分交给 DiffusionRock，以世界状态为条件整块生成；状态与权重都固定后，SiliconRock 把整个模型做进芯片。

基座转换已在 Qwen3.5 这类混合架构上完成验证，正在推广到更多混合架构，并向 Cosmos 3 这类世界模型推进。从头预训练世界状态模型，则为记忆结构与硬件数据流的联合设计留出更大空间，两条路线并行。

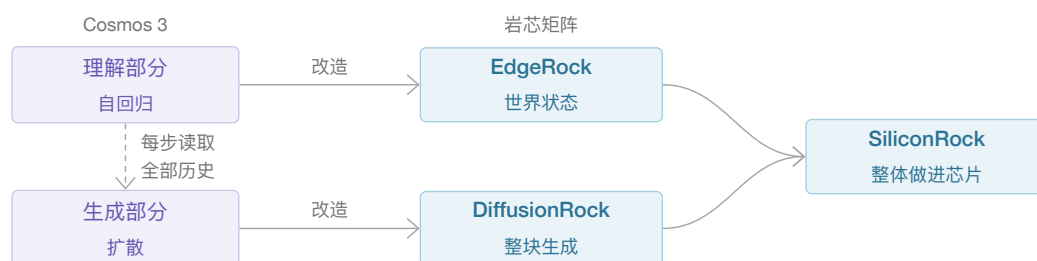


图 8. 以 Cosmos 3 为例：理解部分改为 EdgeRock 的世界状态，生成部分改为 DiffusionRock 的整块生成，再由 SiliconRock 整体做进芯片。

应用与部署

快，让 Agent 更早完成任务，让模型赶上物理世界的控制窗口。

Agent：与前沿大模型混合加速

Agent 的大部分 token 花在高频生成上：写代码、调用工具、执行子任务、反复检查和修正；前沿大模型擅长的规划和关键判断，只占其中一小部分。混合加速按这个结构分工：前沿模型负责规划、关键判断和最终校验，岩芯矩阵的快模型承担大量高频生成，也可以先起草、再交给前沿模型校验。恒定内存还让 Agent 的历史内存不再随工具调用次数增长。收益以端到端任务时间和成本衡量。

物理世界：赶上控制窗口

机器人、车载和工业设备按固定控制窗口工作：结果在窗口内返回，模型才能参与本轮动作。恒定内存让机器人可以让感知与推理在同一个循环里一直跑下去，记忆预算不随任务推进而漂移；高速

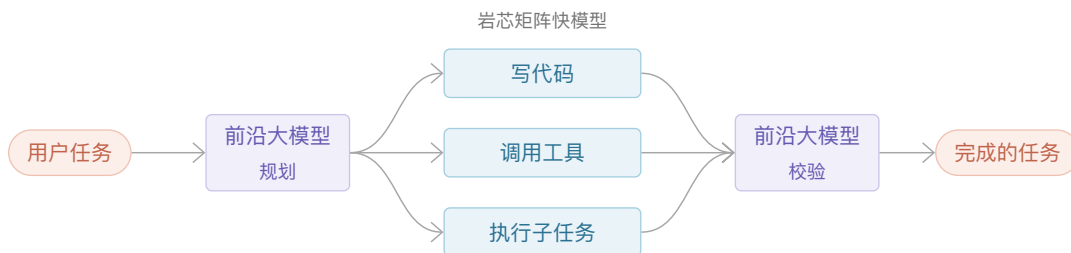


图 9. Agent 的混合加速：前沿大模型负责规划与校验，岩芯矩阵快模型并行承担高频生成。

生成让候选动作、轨迹和未来状态更早返回。以谷歌 RT-2 为代表的视觉语言动作模型，已经展示了把预训练知识迁移到机器人控制的路径；在岩芯矩阵的设计中，传统控制器继续负责最后的物理约束和安全执行。

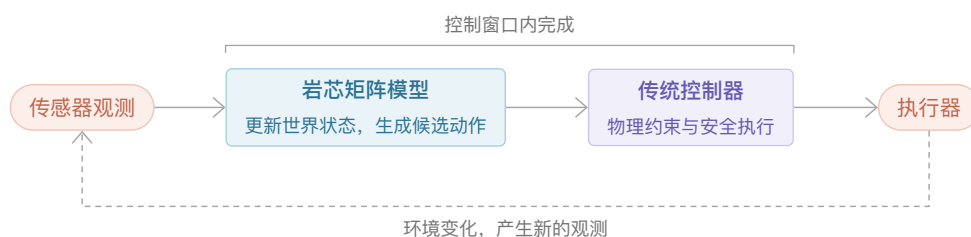


图 10. 进入物理世界：观测、状态更新与候选动作都在控制窗口内完成，传统控制器负责最后的物理约束。

部署：从端侧到数据中心

数据中心承接长推理、Agent 以及与前沿大模型的混合加速；区域边缘让企业和园区共享推理能力；端侧在隐私、弱网和现场响应的场景中本地运行。SiliconRock 服务其中调用量大、形态稳定的负载。无论部署在哪里，每条请求要多少内存，推理开始前就能算清楚：

- 推理服务可以按请求数规划容量，不必猜每条请求最终会跑多久；
- 端侧设备一次分配到位，推理再长也不会把内存耗尽；
- 智能体的历史内存不再随工具调用次数增长；
- 专用芯片可以把全部历史状态放进片上存储。

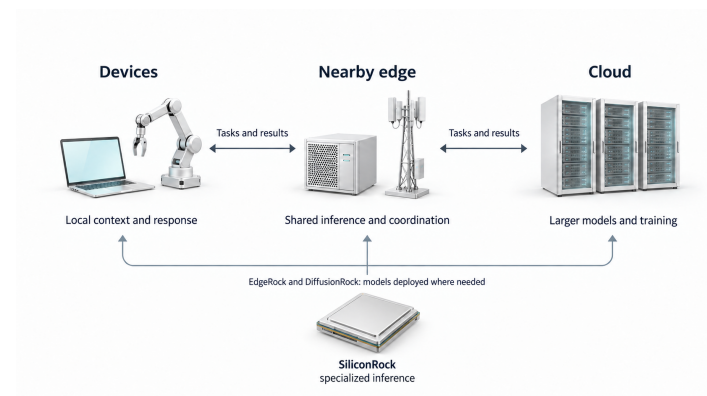


图 11. 同一套模型覆盖端侧、区域边缘与数据中心；每条请求要多少内存，推理开始前就能算清楚。

路线图

已经开源的成果、正在推进的下一步，以及设计目标。

产品线	当前	下一步	目标
EdgeRock	EdgeRock-4B 已开源，支持文本与图像推理；KV 砍掉 99.1%，单卡吞吐最高 6.42 倍，推理能力与领先小模型同一水平。	发布更多规模；从头预训练世界状态模型；把转换推广到其他混合架构和 Cosmos 3 这类世界模型。	全模态的世界状态模型系列：恒定内存下覆盖文本、图像、视频、音频与传感器流，端侧与数据中心通用。
DiffusionRock	块级并行生成，推理速度已达每秒 1,500 token。	开源模型与推理代码；以世界状态为条件，支持长任务的整块生成。	代码、工具调用和动作计划整块即时生成；相同质量下，得到完整结果的时间远少于逐 token 生成。
SiliconRock	固定权重宏级原型，已走通从模型到芯片版图的全流程。	把 EdgeRock 的局部窗口与锚点读取纳入调度；完整模型的低精度闭合；商用工艺实现。	单用户百万 token/s：状态与权重都固定的模型做进芯片，承接调用量大的成熟模型。

三条产品线指向同一个目标：Ultra-Fast Token，让强模型以单用户百万 token/s 的速度持续生成。

结语

强推理不需要一份随每个 token 增长的记忆。岩芯矩阵让 KV Cache 根本不增长，用世界状态承载世界，再把这份确定性一路转化为速度：EdgeRock 让吞吐提升到数倍，DiffusionRock 让生成整块完成，SiliconRock 让单用户百万 token/s 成为设计目标。这就是 Ultra-Fast Token：强，而且快。

- **获取模型**：EdgeRock-4B 的权重与推理代码已开源，见 GitHub ([rockmatrix-ai/EdgeRock](https://github.com/rockmatrix-ai/EdgeRock)) 与 Hugging Face ([rockmatrix-ai/EdgeRock](https://huggingface.co/rockmatrix-ai/EdgeRock))；技术细节见技术报告 arXiv:2610.10118。
- **与我们合作**：面向 Agent 与推理服务的高速 token API、私有部署，以及模型与芯片的联合设计。欢迎来信 support@rockmatrix.ai。

模型想得再久，记忆也不会再膨胀；推理越长，也不再越贵。

贡献者

徐骅、张未明、季慧山、王意、陈歌、熊妍