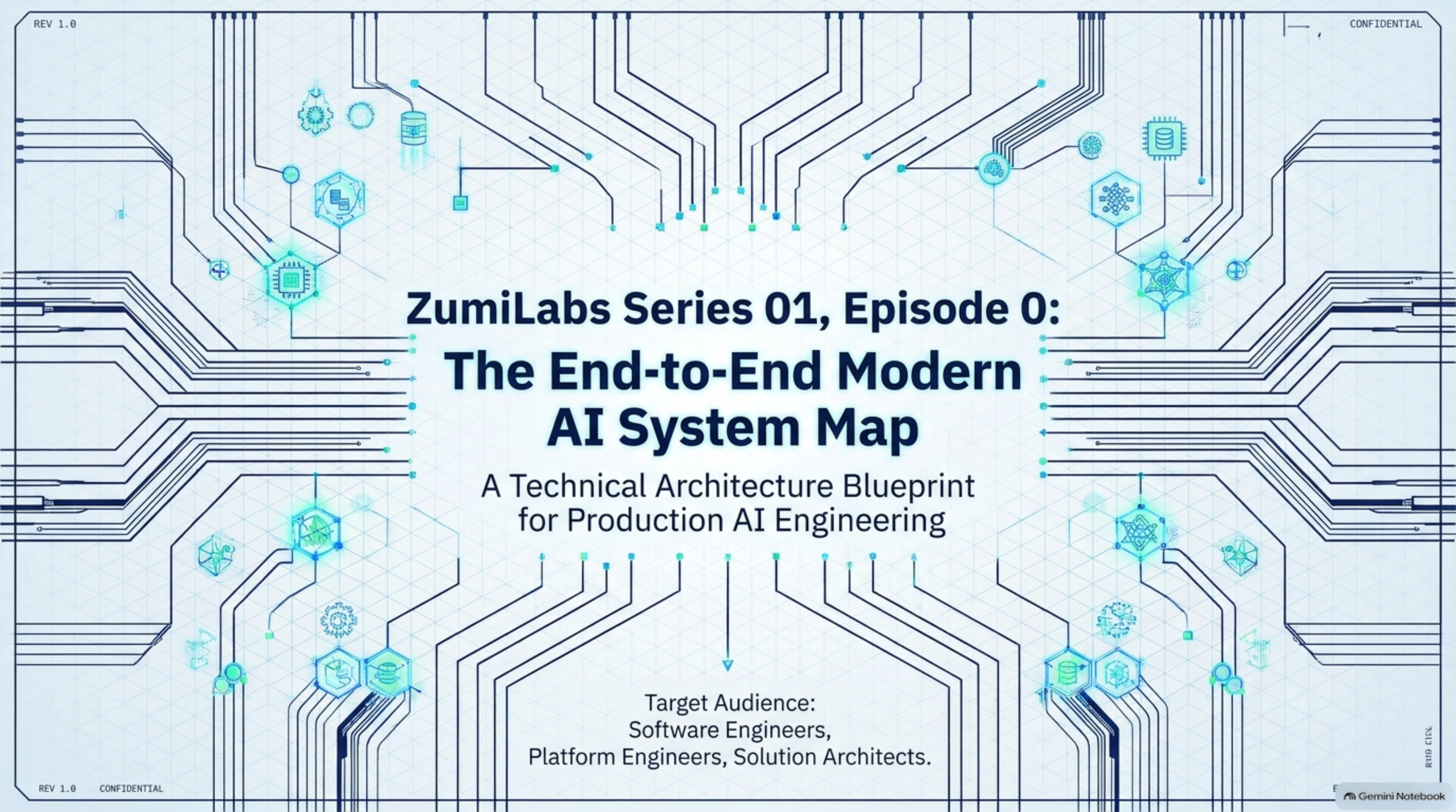


ZumiLabs Series 01, Episode 0:

The End-to-End Modern AI System Map

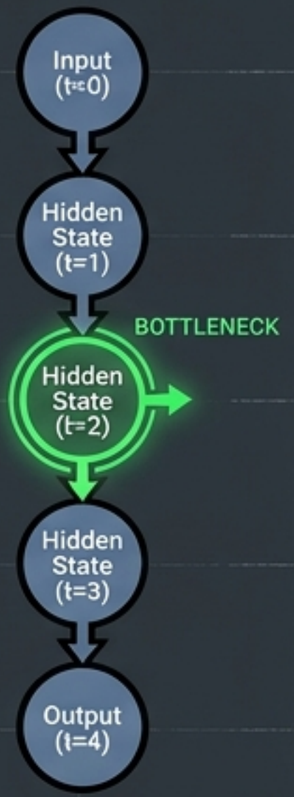
A Technical Architecture Blueprint for Production AI Engineering

Target Audience:
Software Engineers,
Platform Engineers, Solution Architects.

Gemini Notebook

The Architectural Paradigm Shift

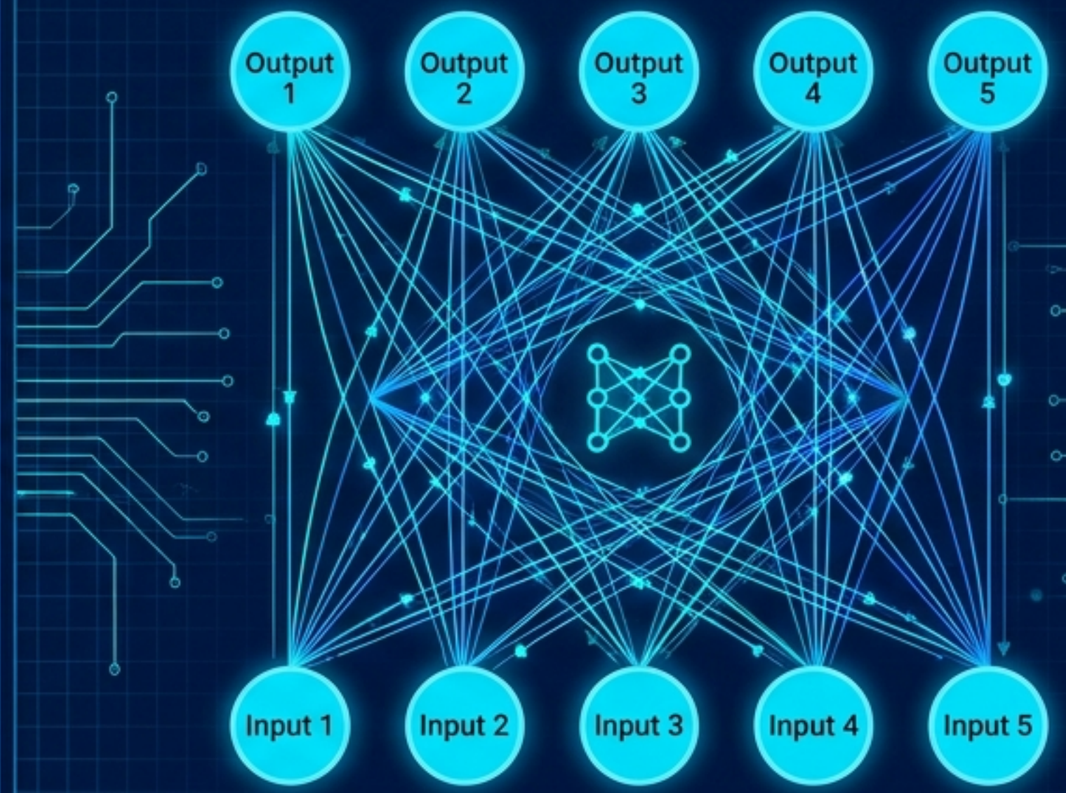
Sequential Bottlenecks
(Classical ML / RNN)



Processing time scales linearly:
O(N) sequential operations.

Brittle deterministic rules & vanishing gradients.

Parallelizable Representations
(Transformers)



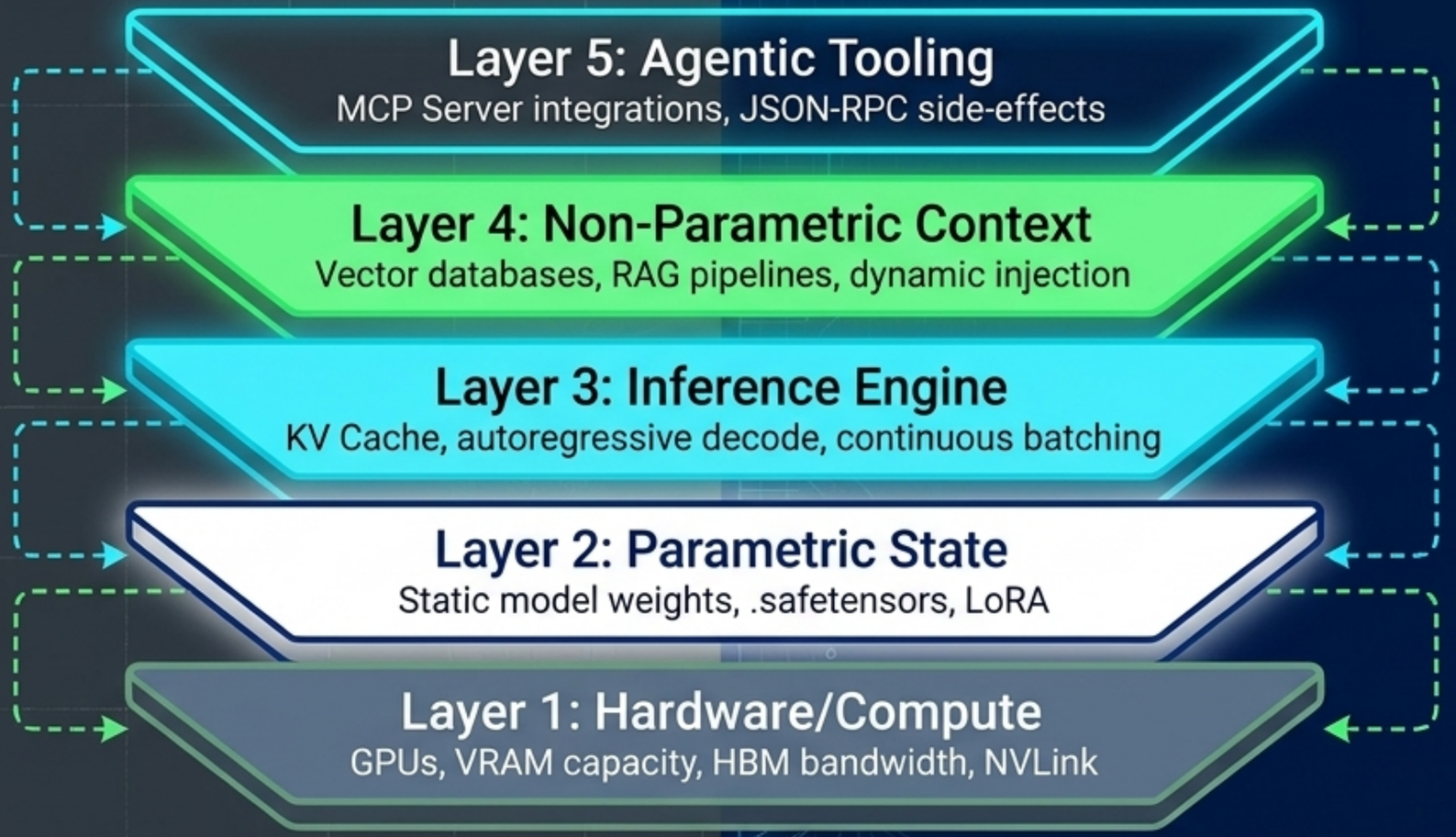
Self-attention scales quadratically $O(N^2)$,
but sequential operations are O(1).

Dense learned representations enable
massive parallelization across GPU clusters

Gemini Notebook

Production AI is a Distributed System

Macro-level 5-tier architecture



A .safetensors file is not a product. Production AI requires orchestrating static parametric state with dynamic external systems.

Gemini Notebook

The Offline Model Creation Funnel



Raw Text Corpora -> Subword Tokenization

UTF-8 bytes -> Int32 Token IDs

Next-token cross-entropy loss

Output: Base LLM

Instruction formatting

Output: Instruction-tuned model

RLHF or DPO tuning

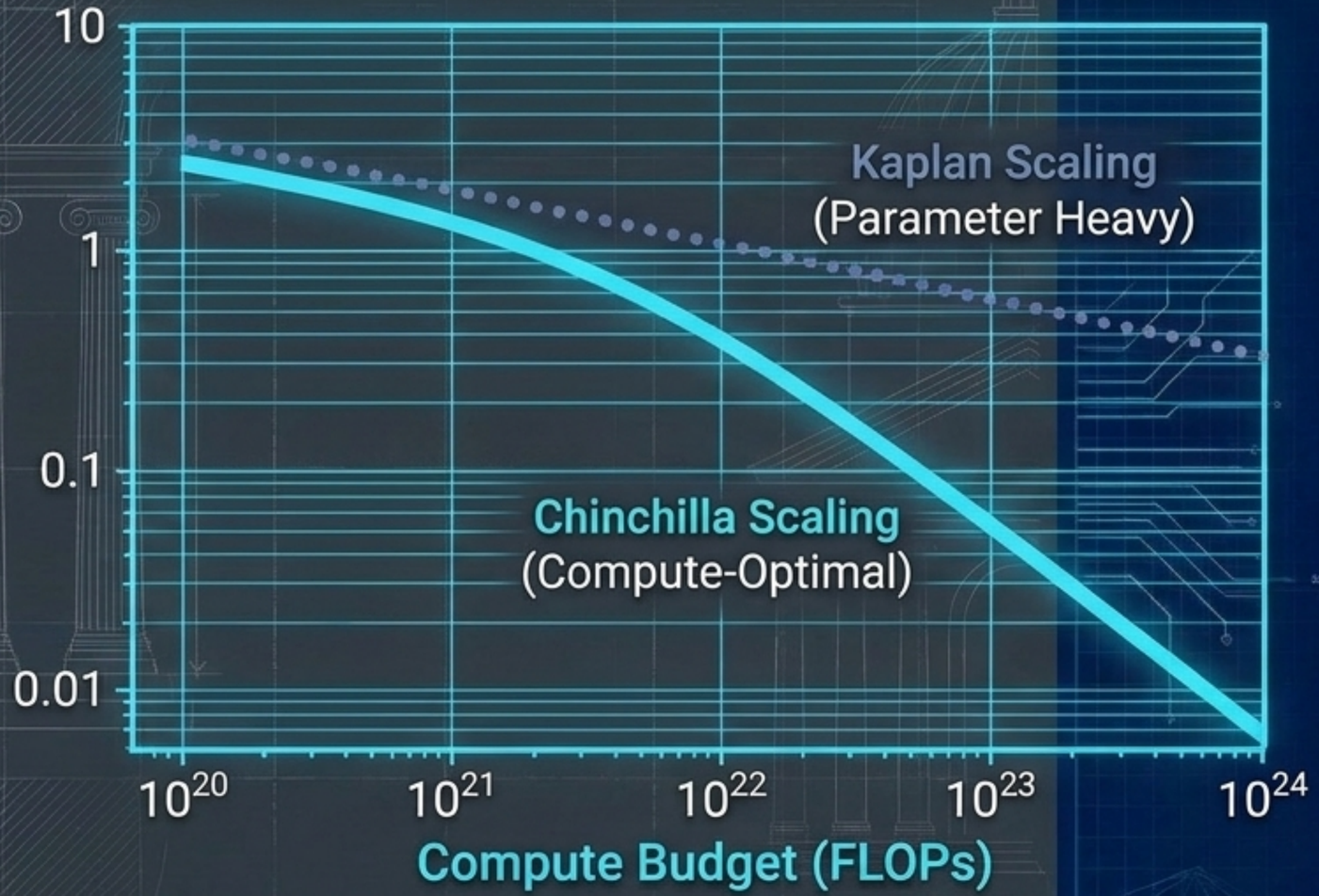
Human preference mapping

Ready for deployment

Static Parametric Memory

Gemini Notebook

Self-Supervised Pre-Training & Scaling Laws



The Chinchilla Law

Compute-optimal training requires scaling tokens and parameters equally.

E.g., a 70B parameter model requires ~1.4 Trillion tokens to be optimally trained.

Simply building larger models without proportionally larger, high-quality datasets is computationally wasteful.

Gemini Notebook

Supervised Fine-Tuning (SFT): Imposing Structure

Base Model Behavior

```
User: Write a Python script.


Assistant: [Model continues
rambling, generating website
metadata, or outputting random forum
text...]
```

SFT Dialogue Template

```
<|start_header_id|>system<|end_header_id|>

You are a helpful assistant.

<|eot_id|><|start_header_id|>user
<|end_header_id|>

Write a Python script.

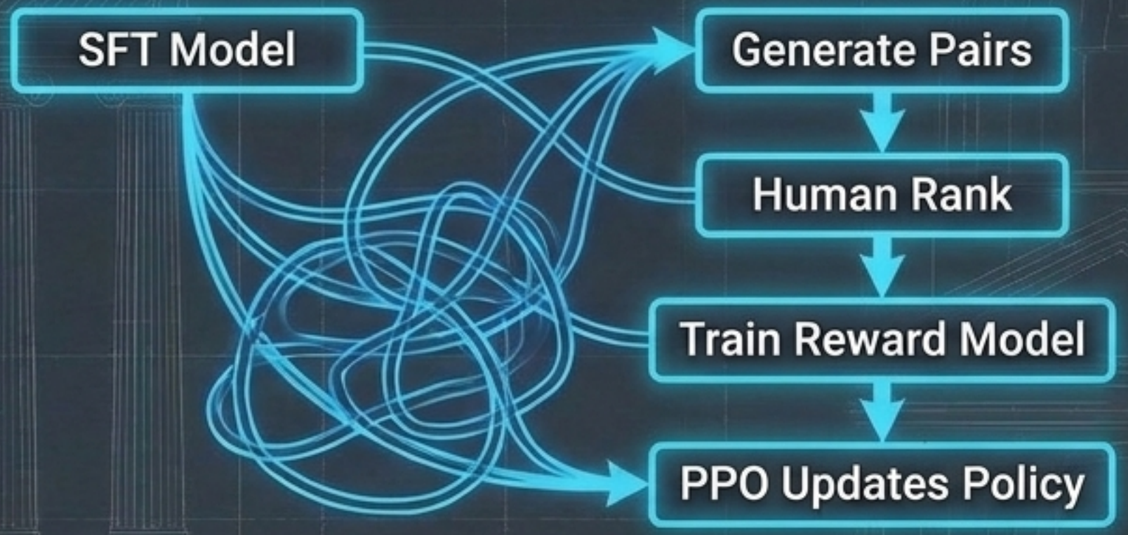
<|eot_id|><|start_header_id|>assistant
<|end_header_id|>
```

SFT trains the model to recognize explicit boundaries (System, User, Assistant). The loss function heavily penalize failing to adopt the Assistant role and emitting the End-Of-Turn (<|eot_id|>) token correctly.

Gemini Notebook

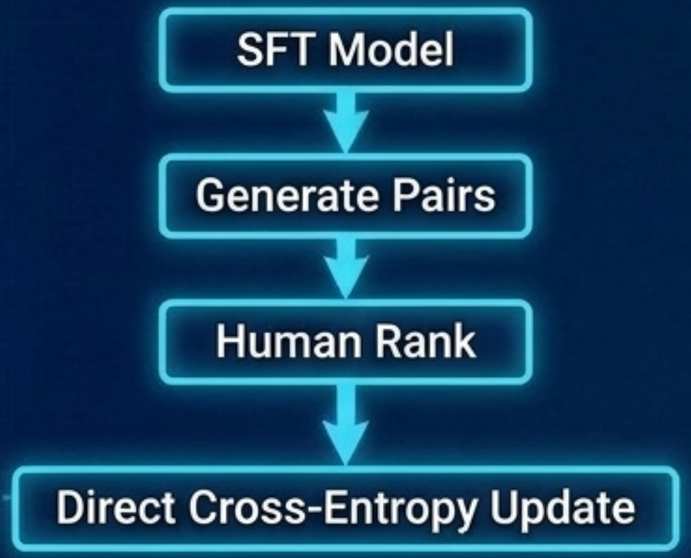
Preference Alignment: RLHF vs. DPO

RLHF
(Reinforcement Learning from Human Feedback)



High complexity. Requires a secondary Reward Model and unstable RL gradients.

DPO
(Direct Preference Optimization)



High efficiency. Bypasses Reward Model entirely. Policy updated directly from preference data.

Metric	RLHF	DPO
Compute Cost	High (Multiple Models)	Low (Single Policy)
Stability	Low (RL Gradients)	High (Cross-Entropy)
Data Needs	Human Preference Pairs	Human Preference Pairs

Gemini Notebook

End-to-End Runtime Pipeline: Tracing the Prompt

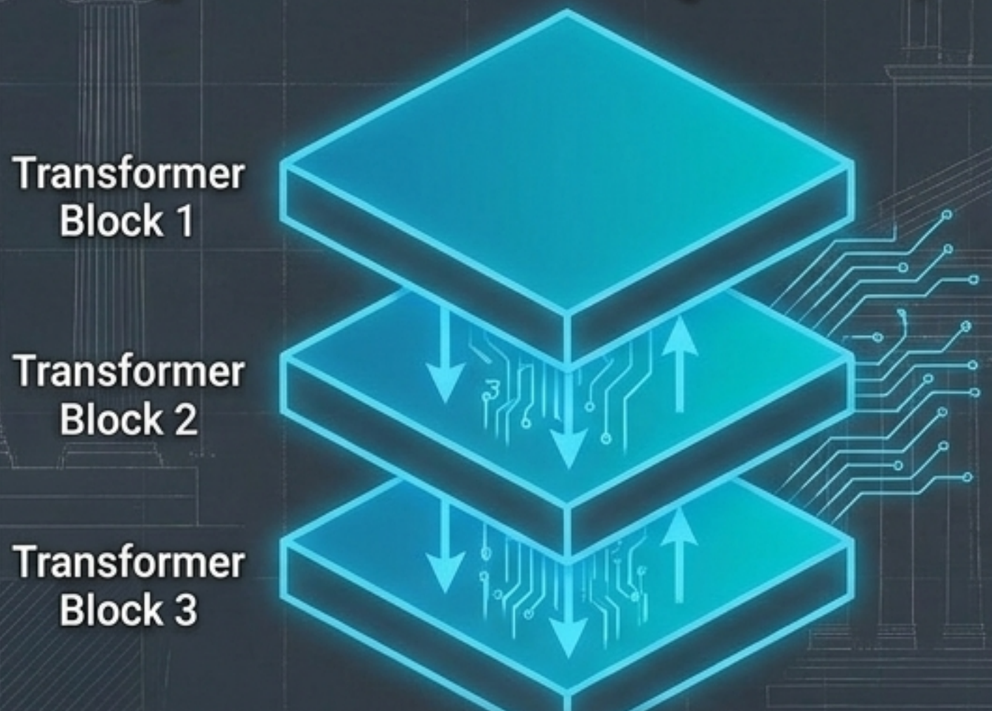
Stage 1: Raw String --> Tokenizer -> [Token IDs]

Raw String: “User: Write a Python script.”

1	3	6	11	3	5	5	7	9	21	13	13	18

Stage 2: Dense Embedding Table -> [Continuous Vectors] + Positional Encodings (RoPE)









Logit Sampling (Temp, Top-K, Top-P)

[Logits over vocabulary]

Stage 5: Logit Sampling (Temp, Top-K, Top-P) -> Selected Token ID -> Detokenizer

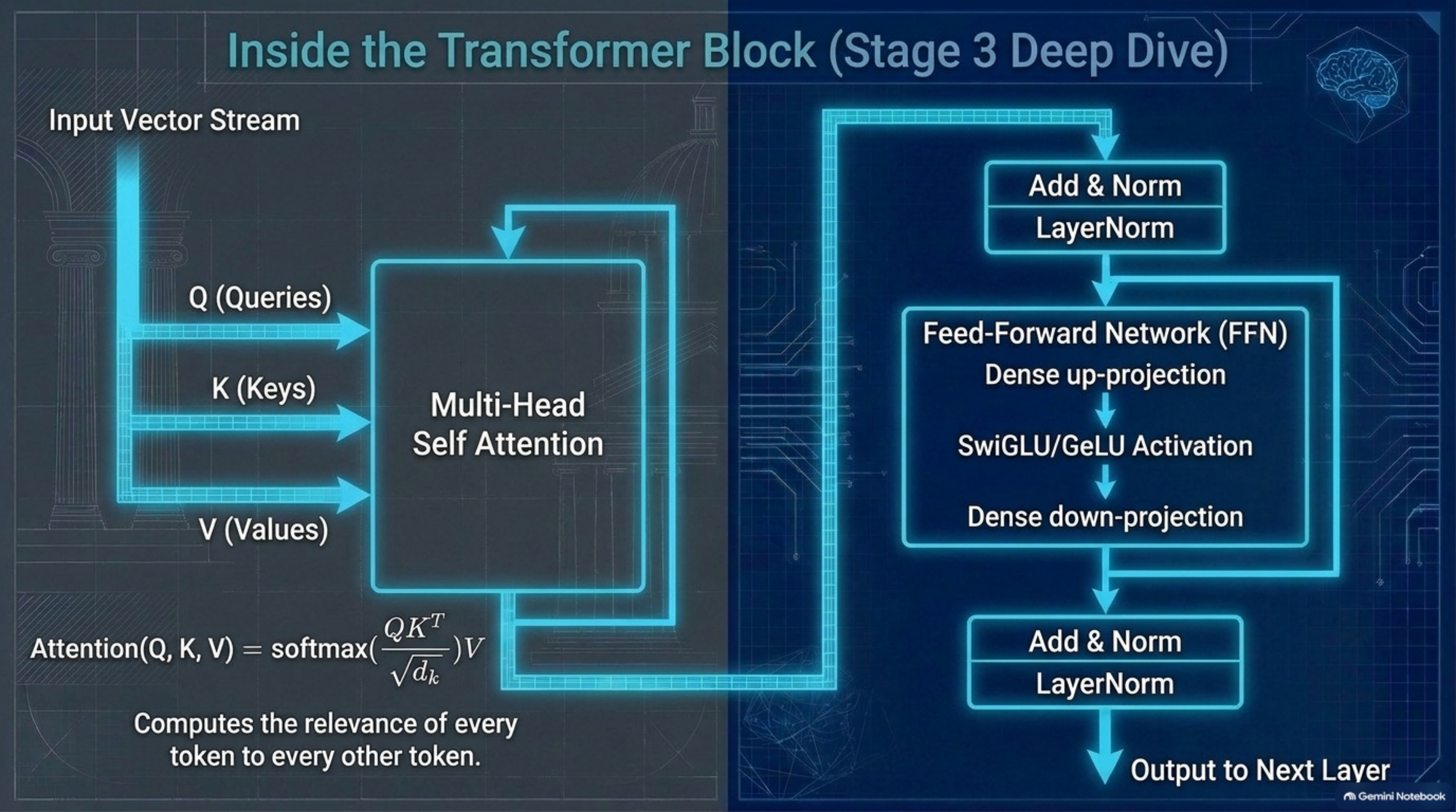
8

Output String

Output String: `"def python_script():"`

Gemini Notebook

Inside the Transformer Block (Stage 3 Deep Dive)

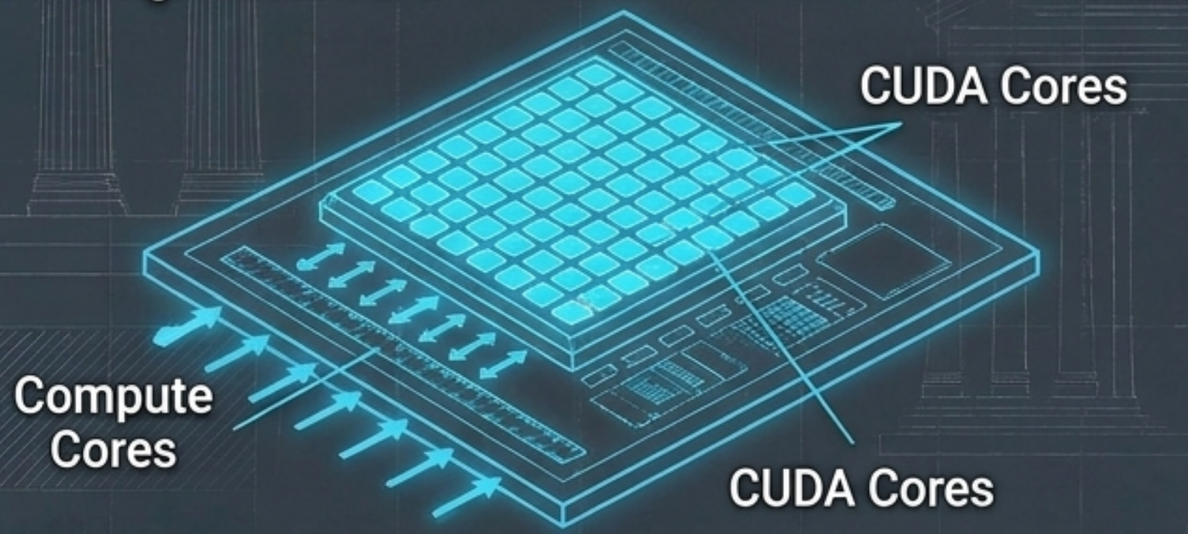


Gemini Notebook

The Inference Bottleneck: Prefill vs. Decode

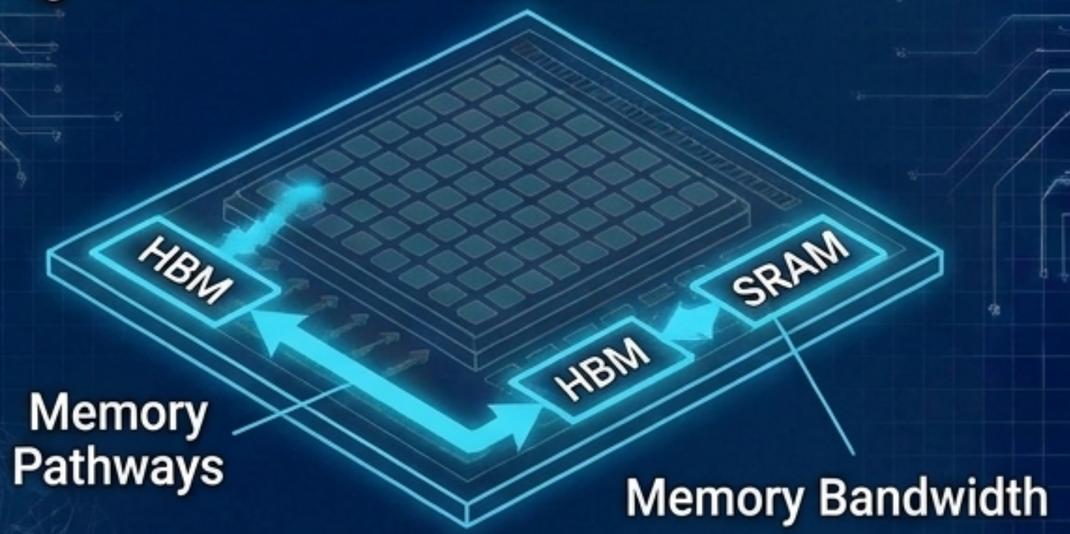
The Prefill Phase

- Action: Processing the user's initial prompt.
- Mechanics: Computes all prompt tokens in parallel simultaneously.
- Bottleneck: COMPUTE-BOUND
- GPU CUDA cores are fully saturated. High utilization.



Autoregressive Decode Phase

- Action: Generating the answer.
- Mechanics: Generates exactly one token at a time. Reads all parameters from memory per token.
- Bottleneck: MEMORY-BANDWIDTH-BOUND
- GPU compute cores sit mostly idle waiting for weights to travel from HBM to SRAM.



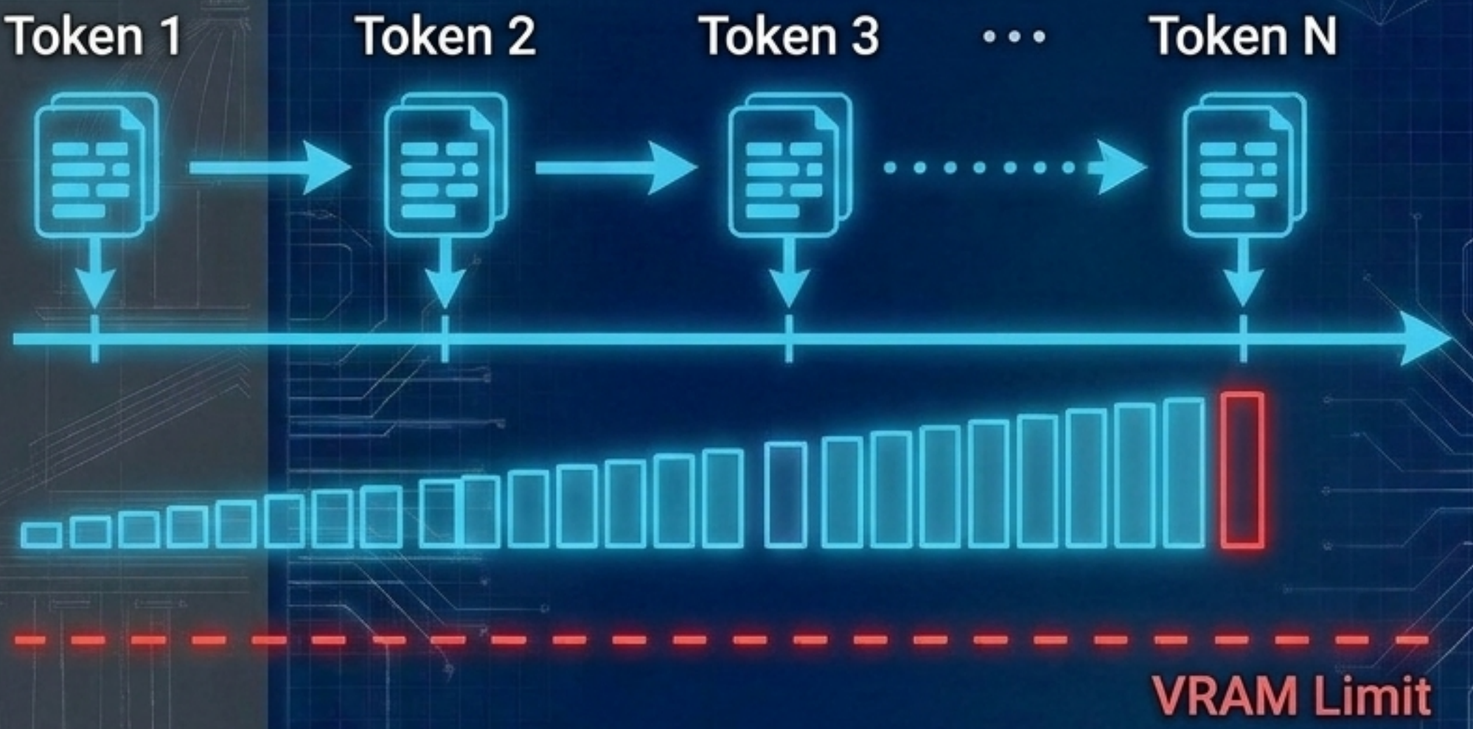
Inference is not a single workload. It is two radically different workloads sharing the same hardware.

Gemini Notebook

KV Cache Memory Pressure

Explainer

Autoregressive decoding requires caching the computed Keys and Values of all previous tokens to avoid recomputing them.



The Math

Cache Size = 2 x Precision x Layers x Dimensions x Sequence Length

Impact

The KV cache consumes VRAM linearly with sequence length. Long context windows severely restrict maximum batch size, capping hardware throughput.

Gemini Notebook

The Limits of Frozen Weights

Parametric State is Frozen



Weights are baked at the final training epoch. They encode statistical likelihoods and latent conceptual relationships, NOT a queryable relational database.

Design Rule

Use weights for **Capability** (reasoning, formatting, language).
Use external systems for **State** (facts, real-time data, citations).

The Hallucination Mechanism



When asked for a specific fact it hasn't strongly memorized, the model defaults to generating a statistically plausible-sounding string of tokens, resulting in confident falsehoods.

Gemini Notebook

The RAG Pipeline (Non-Parametric Grounding)



User Query
Sent to Embedding Model

Dense Vector
Traverses Vector Database (HNSW Graph)

Top-K Retrieval
Returns semantically similar text chunks

Context Injection
Chunks concatenated into the prompt string

LLM Generation
Answer grounded strictly in injected context

Explainer
(optional, for consistency)
Transforms a memory-recall task into a reading-comprehension task.

Gemini Notebook

Engineering Decision Matrix

Dimension	Prompt Engineering	RAG	Fine-Tuning (SFT/LoRA)
Data Volatility	Static	Highly Dynamic	Frozen/Static
Task Specificity	General	Domain-Knowledge	Format/Style/Tone
Cost	Zero upfront	Vector DB overhead	High upfront compute
Latency	Minimal	+ Retrieval time	Minimal

Never fine-tune to inject new facts. Fine-tune for form; RAG for facts.

Gemini Notebook

Overcoming Hardware Bottlenecks

VRAM Capacity
(Storage Tank)

Stores: Model weights, KV Cache, Activations.

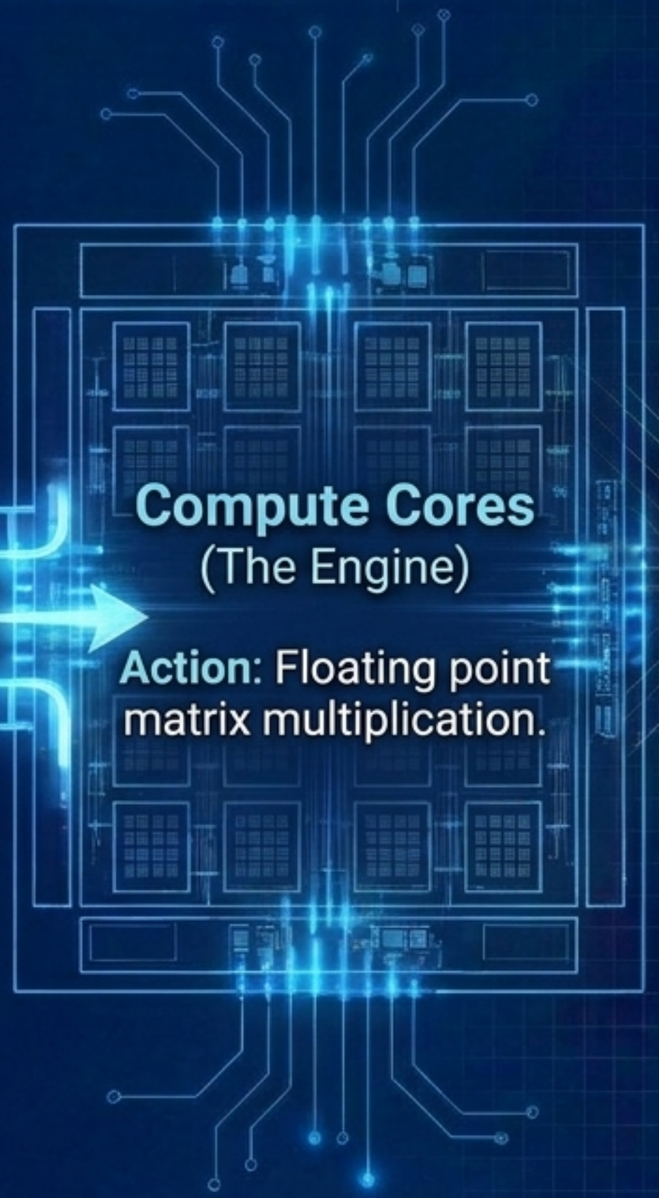
Constraint: Limits how large a model can be loaded, and the maximum batch size/context window.

HBM Bandwidth
(The Narrow Pipe)

Speed of moving data from VRAM to Compute Cores.

HBM Bandwidth
(The Narrow Pipe)

Constraint: Limits autoregressive decoding speed (tokens per sec), as the entire model must be read for every single generated token.



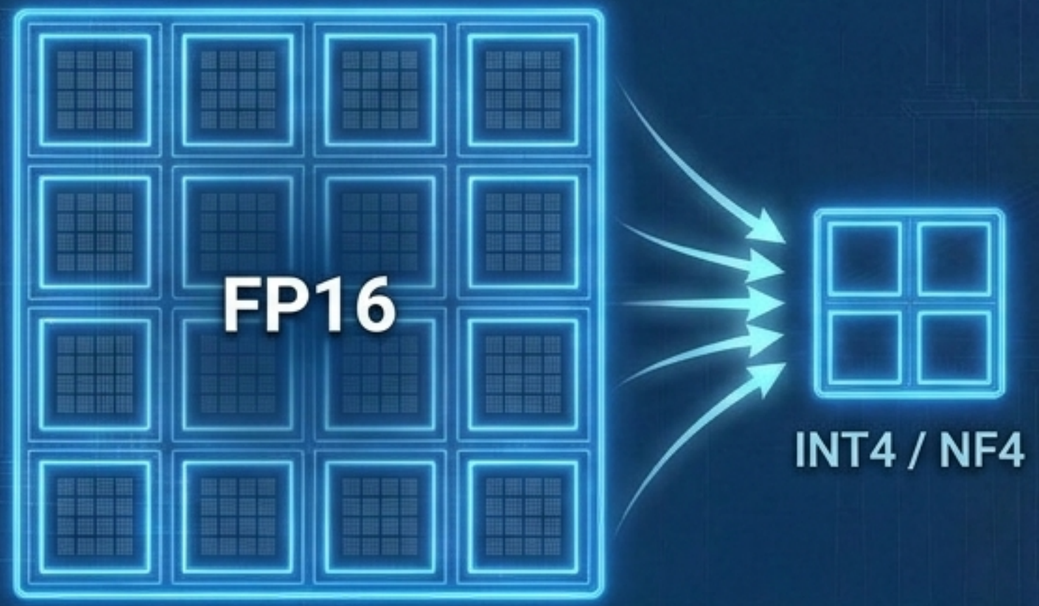
Compute Cores
(The Engine)

Action: Floating point matrix multiplication.

Gemini Notebook

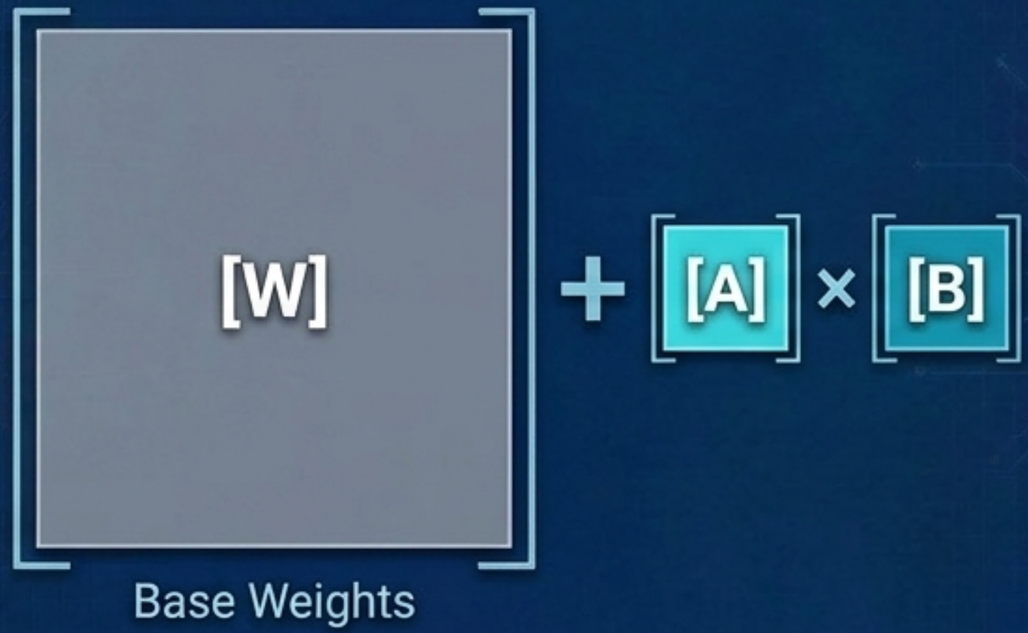
Compression & Adaptation

Post-Training Quantization
(Compression)



Shrinks the VRAM footprint of static weights drastically, trading marginal mathematical precision for massive deployability and decode speed.

Parameter-Efficient Fine-Tuning
(LoRA)



Injects targeted behavioral changes without updating the multi-billion parameter base model. Extremely VRAM efficient during training.

Gemini Notebook

Topology Trade-Off: Cloud APIs vs. Local SLMs

Frontier Cloud Models
(e.g., GPT-4, Claude 3)

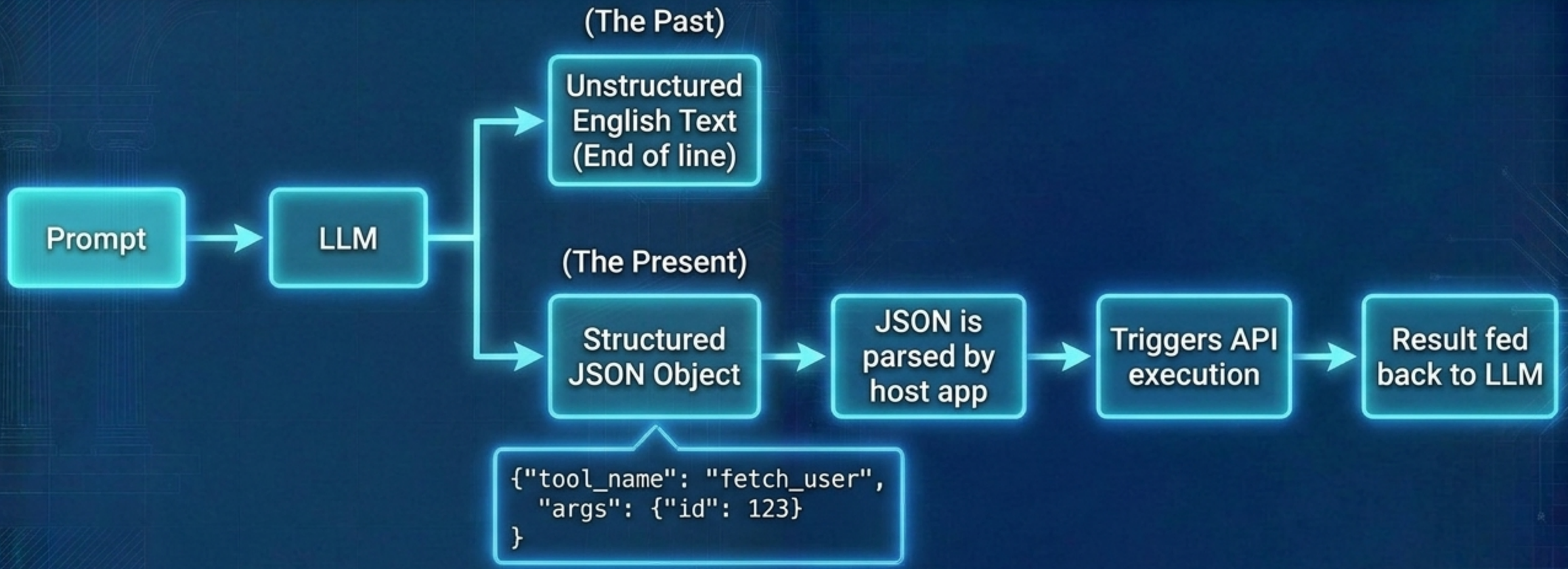
- \+ Peak generalized reasoning
- \+ Massive zero-shot capabilities
- \+ Zero infrastructure maintenance
- − High tail-latency variability
- − Strict rate limits
- − Data privacy and residency risks

Local / Quantized SLMs
(e.g., Llama-3-8B, Mistral)

- \+ Predictable tail-latency
- \+ Total data privacy
- \+ Highly targeted capabilities via LoRA
- \+ Cheaper at scale
- − Lower ceiling for complex logical reasoning without extensive scaffolding

Gemini Notebook

Beyond Next-Token Prediction

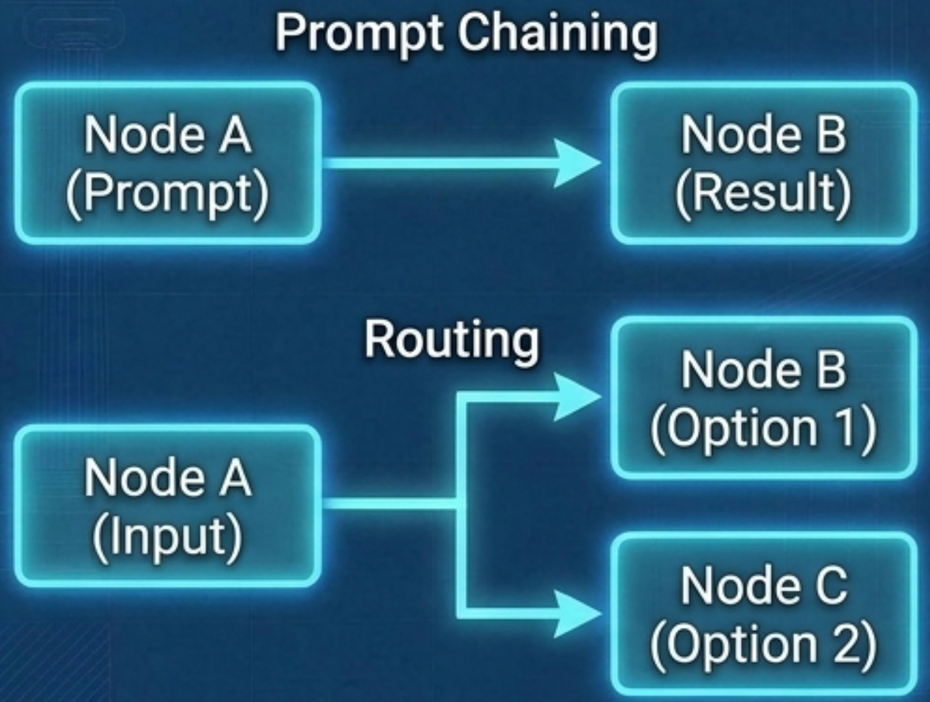


Agents emerge when we stop treating models as passive chat interfaces and start treating them as dynamic routing engines that output structured code.

Gemini Notebook

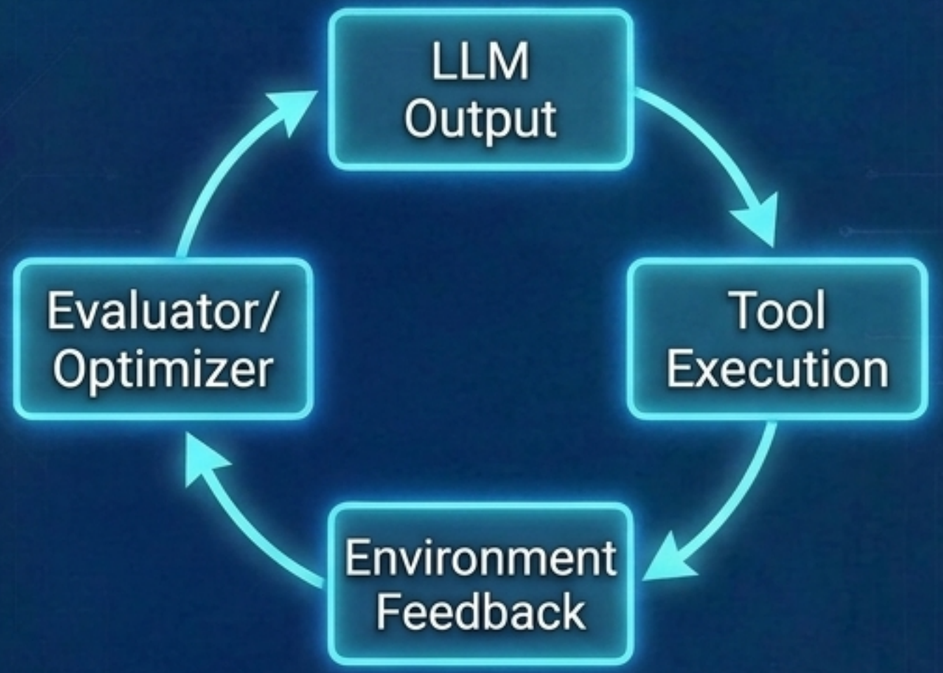
Workflows vs. Autonomous Agents

Deterministic Workflows



High predictability, low latency, ideal for well-defined engineering tasks.

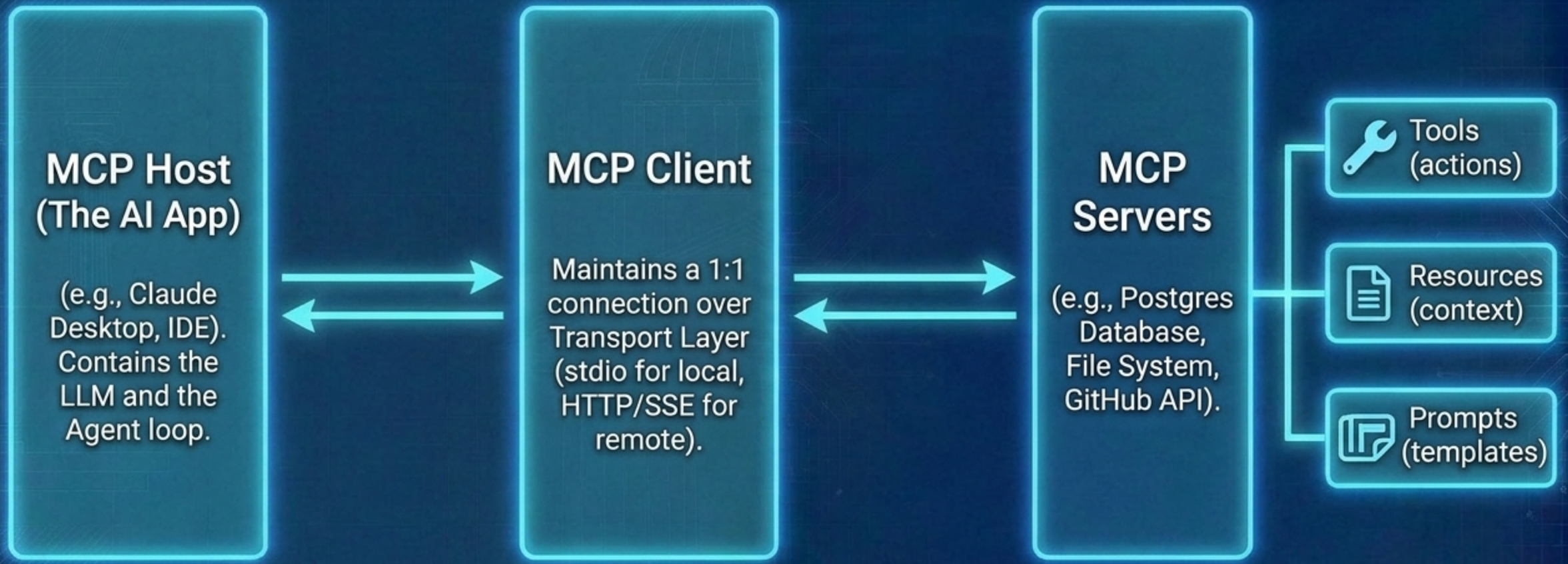
Autonomous Agents



Open-ended problem solving, requires stopping conditions, higher latency and cost, prone to infinite loops.

Gemini Notebook

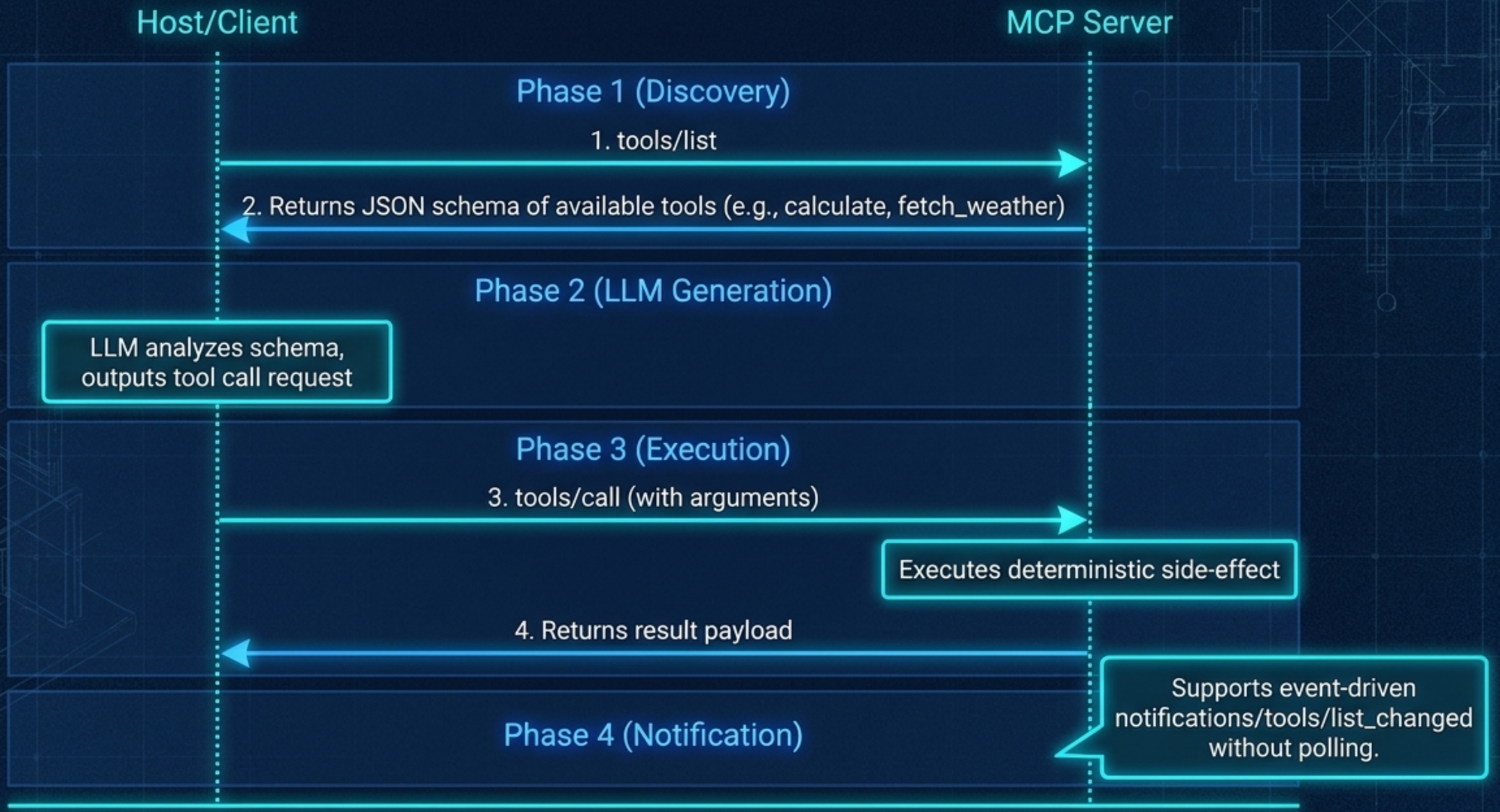
The Model Context Protocol (MCP) Standard



MCP standardizes the fragmented Agent-Computer Interface, standardizing JSON-RPC boundaries to connect any AI to any data source without custom integration code.

Gemini Notebook

Deep Dive: MCP Implementation



Gemini Notebook

Architectural Diagnostics (Failure Modes)

Symptom	Root Architectural Cause
Out of Memory (OOM) during generation	KV Cache Overflow; batch size or context window exceeds VRAM capacity.
High confidence hallucination of facts	Relying on frozen parametric weights instead of injecting dynamic RAG retrieval.
Agent stuck in an infinite loop	ReAct loop missing strict stopping conditions or robust error parsing on tool-call failures.
Slow Time-to-First-Token (TTFT)	Compute-bound prefill phase overwhelmed by massive prompt context.

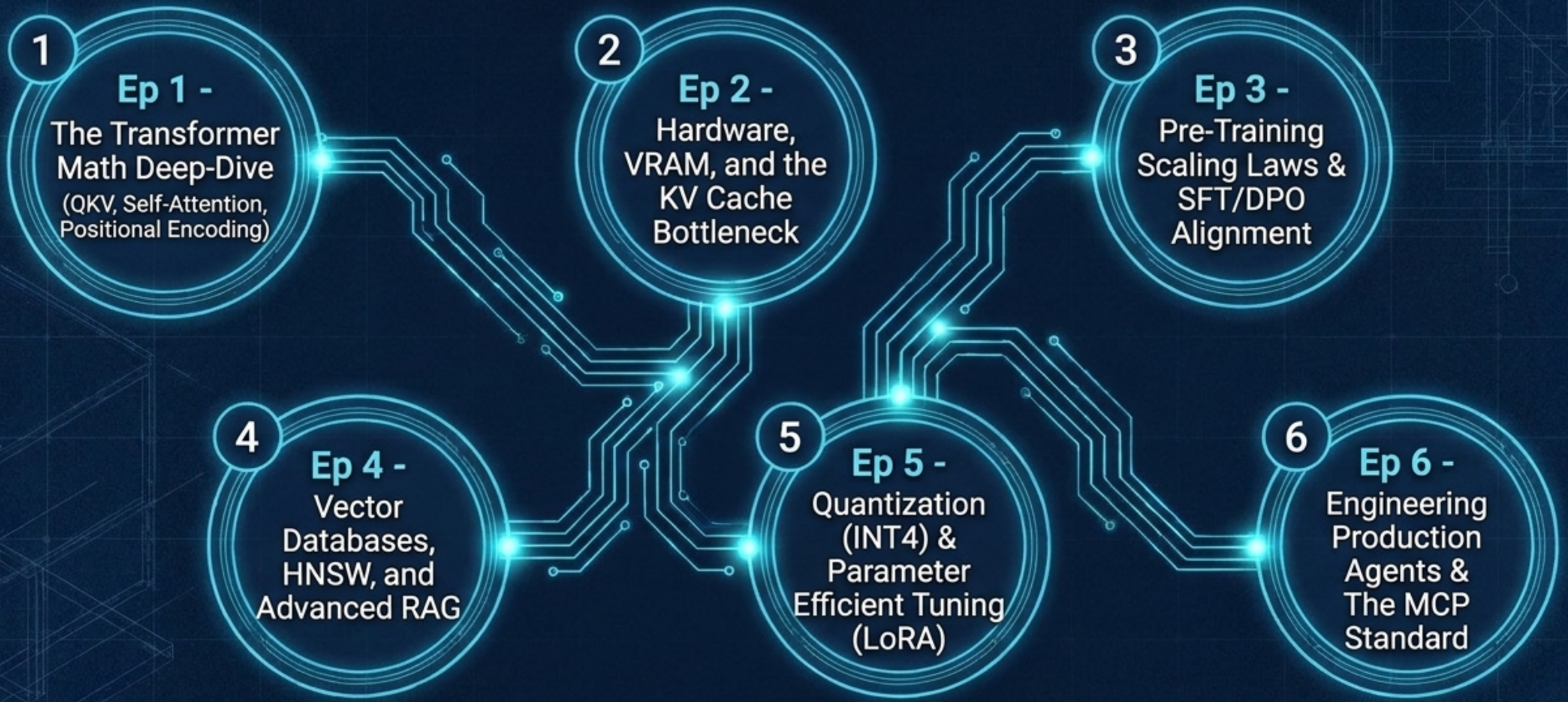
Gemini Notebook

Component Interface Cheat-Sheet

Subsystem	Input	Output
Tokenizer	Raw String	Array of Int32 Token IDs
Embedding Layer	Token IDs	Continuous Dense Vectors + RoPE
Transformer Block	Q, K, V Vectors	Contextualized Vector Representations
LM Head	Hidden States	Logit probabilities over vocabulary size
Vector DB (RAG)	Dense Query Vector	Top-K String Chunks (Cosine Similarity)
MCP Server	JSON-RPC tools/call	Deterministic API Execution / Status

Gemini Notebook

ZumiLabs Series 01 Roadmap



This Episode 0 overview connects the master blueprint.
Episodes 1-6 will mathematically unpack the internal mechanisms.

Gemini Notebook