PURETENSOR
INTELLIGENCE DISTILLED

Hybrid Mamba Cannot Cache a Transcript

Choosing a seat for a live co-pilot by time to first answer

Paper: PT-R-2026-024 v1.0

Author: PureTensor Ltd

Date: 29 September 2026

Status: Published

Abstract

A live meeting co-pilot must return its first readable suggestion within a fixed budget after an utterance ends. Our design target is 1.5 seconds at the 95th percentile. The loop runs entirely on-premises: speech-to-text, retrieval over the operator's notes, and a language model that reads a rolling transcript. Two candidate model seats were evaluated, a pure-transformer mixture-of-experts model (Qwen3.8-Flash-Next, FP8, tensor-parallel 2) and a hybrid Mamba-2/Transformer model from the Nemotron family (single GPU, shared with other services). The metric is time to first answer, meaning the first content token the user can read, not time to first token, which would score a configuration that emits only hidden reasoning tokens as fast.

The transformer seat returned its first answer in 108 ms at 8k tokens of transcript and 159 ms at 64k tokens. The hybrid Mamba seat took 253 ms and 517 ms respectively. The gap widens with transcript length because the hybrid seat cannot reuse a cached prefix. Prefix caching, the mechanism that lets an append-only transcript pay only for its new tokens, relies on slicing a key-value cache at a token boundary. A state-space model stores a single recurrent vector per layer, not a sequence of per-token blocks; it cannot be sliced, so the engine recomputes the full transcript on every call. We measured a 3.8–4.2% cache hit rate on the hybrid seat even with prefix caching enabled. An identical 6k-token prompt repeated eight times gave a flat 194–197 ms each time; the transformer seat paid 355 ms and 401 ms for the first two calls, then held at about 100 ms for every call after.

The co-pilot now uses the transformer seat with reasoning effort set to "none". The hybrid seat remains in service for batch fan-out work, where a transcript is not being appended to and the cache limitation does not bite. The selection criterion for any future co-pilot seat is a pure transformer architecture, working prefix caching, and a reasoning-off mode. A design rule follows: keep the transcript strictly append-only and place all volatile content (retrieved notes, instructions) after it, never before.

These results come from a single harness session per configuration, with p50 figures only. The two seats differ in allocation (one shares a GPU, the other has two GPUs to itself), so the absolute latencies are not a clean architecture comparison; the cache hit rate is. No output quality was measured.

1. The Application and Its Latency Budget

The co-pilot listens to a live meeting, receives a rolling transcript from a speech-to-text service, retrieves relevant notes from the operator's knowledge base, and asks a language model to return short tagged suggestions. The design budget is 1.5 seconds at the 95th percentile from the end of an utterance to the first readable token of the suggestion.

The loop has four legs, all running on-premises. Speech-to-text is a Whisper large-v3-turbo service on one node. Retrieval is a small vector-search bridge. The language model is served with vLLM, streaming enabled. Two candidate seats were evaluated: a pure-transformer mixture-of-experts model (Qwen3.8-Flash-Next, FP8) at tensor-parallel 2 on a two-GPU node, and a hybrid Mamba-2/Transformer model from the Nemotron family on one GPU of another node. That second node is shared with the speech, embedding, reranking, and OCR services.

A re-runnable latency harness was written for the loop. Every figure reported below is a p50 from that harness against the live endpoints. The harness does not yet include the capture device or the tailnet leg from it, nor does it include end-of-turn detection, which is estimated (not measured) at about 300 ms.

2. Time to First Answer, Not Time to First Token

Both seats can stream reasoning tokens before content tokens. The field is named "reasoning" in this engine build. A configuration that emits many reasoning tokens and no content tokens would score well on time to first token yet never produce a readable answer.

With reasoning enabled, the hybrid Mamba seat emitted 351 reasoning tokens and zero content tokens inside a 1,024-token budget: no readable answer at all. The transformer seat at reasoning effort "low" took 555 ms to its first readable token; at reasoning effort "none" it took 108 ms. All figures in the tables that follow are time to first answer (the first content token), with reasoning turned off.

3. Measured Latencies

The table below shows the p50 latency for each leg of the loop.

Leg	p50 (ms)
Speech-to-text, 3 s chunk	131
Speech-to-text, 11 s chunk	186
Retrieval, k=4	51
Fabric round trip between nodes	0.5
LLM time to first answer, 8k-token transcript, transformer seat	108
LLM time to first answer, 8k-token transcript, hybrid Mamba seat	253
LLM time to first answer, 64k-token transcript, transformer seat	159
LLM time to first answer, 64k-token transcript, hybrid Mamba seat	517
Idle penalty after 90 s silence, either seat	none measured

The transformer seat is faster at both transcript lengths. The gap grows with length: at 8k tokens the hybrid seat is 2.3 times slower; at 64k tokens it is 3.3 times slower. The fabric round trip is negligible.

The end-to-end budget as built sums to about 560 ms: end-of-turn detection at roughly 300 ms (design estimate), speech-to-text at 150 ms, retrieval at 50 ms (running in parallel with the LLM call), and the LLM at 110 ms. Against the 1.5 s p95 requirement this leaves about 2.5 times headroom. The capture device and tailnet leg remain unmeasured.

4. The Architectural Finding: The Hybrid Seat Cannot Cache a Transcript

Prefix caching is the mechanism that makes an append-only transcript cheap. When a prompt shares a long prefix with a prompt seen before, the engine reuses the key-value cache computed for that prefix; each turn pays only for the new tokens. The hybrid Mamba seat runs with prefix caching enabled, yet we measured a 3.8–4.2% hit rate.

An identical 6k-token prompt repeated eight times gave a flat 194–197 ms on the hybrid seat. Append-only growth of the prompt gave the same flat profile: every call paid full price. The transformer seat on the same test went 355 ms, 401 ms, 101 ms, and then held at about 100 ms. The first two calls paid for the prefix; every call after reused it.

The reason is architectural. A transformer stores a key-value pair for every token in the sequence. The cache can be sliced at any token boundary and extended with new tokens. A state-space model, by contrast, compresses the entire history into a single recurrent vector per layer. That vector cannot be sliced; there is no way to say "here is the state after token 6,000, now continue from token 6,001". The engine must recompute the full transcript on every call, so time to first answer grows with transcript length (253 ms at 8k tokens, 517 ms at 64k tokens) rather than staying roughly constant.

The Nemotron Nano family uses the same hybrid Mamba-2/Transformer design. It was not benchmarked (the weights are not on disk), but the conclusion transfers by architecture: any model with state-space layers in the recurrent path will exhibit the same limitation.

5. Prompt Layout and Cache Invalidation

A prefix cache is only useful if the prefix is stable. In a meeting co-pilot the transcript is append-only by nature: new utterances arrive at the end, old utterances do not change. Retrieved notes and instructions, however, may change from turn to turn.

If volatile content appears before the transcript, every change invalidates the entire cached prefix. If it appears after the transcript, the stable part of the prefix remains cached and only the volatile tail is recomputed. The design rule is therefore to place the transcript first and all volatile content (retrieved notes, user instructions, system prompts that vary) after it.

This rule applies only to the transformer seat, which can exploit the cache. For the hybrid seat the rule is moot: the cache never helps.

6. Decision

The co-pilot uses the transformer seat with reasoning effort set to "none". The hybrid Mamba seat keeps its role for batch fan-out, where a transcript is processed once rather than appended to on every turn. In that setting the cache limitation does not bite and the single-GPU allocation is an advantage.

The selection criterion for any future co-pilot seat is a pure transformer architecture, working prefix caching, and a reasoning-off mode. The first two requirements ensure that latency stays bounded as the transcript grows. The third ensures that the first token the user sees is a readable answer, not a hidden reasoning trace.

7. What These Numbers Will Not Carry

The two seats differ in more than architecture. The transformer seat has two GPUs to itself at tensor-parallel 2. The hybrid seat shares one GPU with four other services (speech, embedding, reranking, OCR). The absolute latencies therefore reflect both the architectural difference and the allocation difference. The cache hit rate (3.8–4.2% on the hybrid seat versus effective reuse on the transformer seat) is the allocation-independent part of the result; the millisecond figures are not.

No output quality was measured. Both seats produced plausible tagged suggestions on inspection. Whether one is better at the co-pilot task, whether it hallucinates less, or whether it follows the tagging schema more reliably, is unknown.

The figures are p50 values from one harness session per configuration. Variance across sessions was not measured. The 64k-token case is a single prompt length, not a sweep; we do not know whether the relationship between transcript length and latency is linear, sublinear, or stepwise.

The 300 ms end-of-turn estimate is a design figure, not a measurement. The capture device and the tailnet leg from it are not yet measured. The end-to-end budget of 560 ms is therefore provisional.

The smaller Nemotron Nano family was not benchmarked. The conclusion that it shares the cache limitation rests on architectural reasoning, not on measurement.

8. Next Steps

The next experiment is to measure the capture device and tailnet leg, then to build and measure the end-of-turn detector. Once those figures are in hand the 560 ms estimate becomes a measured sum and the headroom against the 1.5 s budget can be stated with confidence.

A second line of work is to sweep transcript length on the transformer seat to confirm that time to first answer stays bounded as the transcript grows toward the context window limit. If it does, the co-pilot can run longer meetings without a latency penalty. If it does not, we will need to summarise or truncate the transcript at some threshold.

PureTensor operates a sovereign AI infrastructure fleet (on-premises GPU compute, storage, and serving) run day-to-day by autonomous agents under human direction. Measurements for this paper were taken on 3 September 2026; raw artefacts are retained. This paper is PT-R-2026-024.