

Memory OS of AI Agent



2026. 10.

presenter: **Joohyung Yun**

PART 1

The Problem



Part 1 The Problem · long-term memory in multi-session conversation

Part 2 Existing Memory Methods and Their Failure Modes

Part 3 MemoryOS · the idea and the full procedure

Part 4 Experiments and Takeaways

Long-Term Memory in Multi-Session Conversation

week 1 · one session

U · I went to the wetland park last week,
the scenery there was beautiful.

A · Sounds like a relaxing experience!

U · There were many little squirrels running
in the trees, very cute.

A · I like squirrels, they signal a healthy environment.

U · I ran two laps in the park and felt great.
I'll keep working out to get slimmer.

A · Exercise is good for health. I'll keep you motivated!

three turns · each turn = one page

facts: park · squirrels · ran two laps

preference: wants to get slimmer

a few weeks later · a separate session

U · Remember the wetland park I told you about?

without memory

Sorry, I don't remember you
mentioning the park.

tests · fact recall

with MemoryOS

You saw beautiful scenery, went
for a run, and saw squirrels.

U · I feel like eating a burger, any recommendations?

without memory

Teriyaki chicken burger,
it's super delicious!

tests · personalization

with MemoryOS

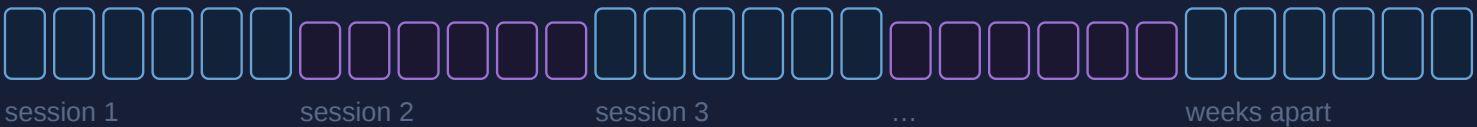
The cheeseburger [...], just don't
forget you're hitting the gym!

- **fact recall** · answer from what was said weeks ago, across sessions
- **personalization** · act on what the user wants without being told again
- **one memory per user** · spans separate sessions · topics grouped by the system, not by the user

Question Types in Long-Term Conversation Benchmarks

LoCoMo

≈ 300 turns
≈ 9K tokens / conversation

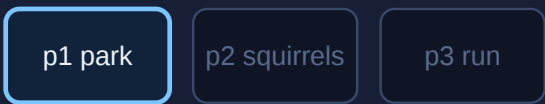


questions asked after the whole conversation
scored by F1 and BLEU-1 against a gold answer

GVD

15 virtual users × 10 days · at least two topics per day · an LLM judge scores retrieval accuracy, correctness, coherence

single-hop



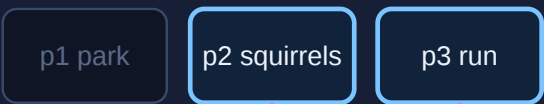
one page

Where did the user go
last week?

needs · one page

answer · **the wetland park**

multi-hop



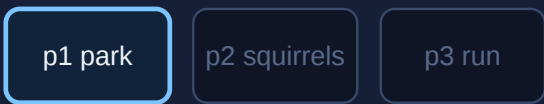
two pages, linked

What did the user see at the
place where they went running?

needs · two pages joined

answer · **squirrels**

temporal



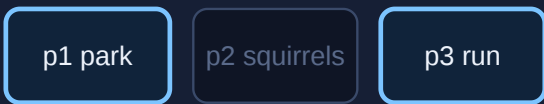
T · week 1 + today's date

How many weeks ago did the
user visit the park?

needs · a page plus its timestamp

answer · **three weeks ago**

open-domain



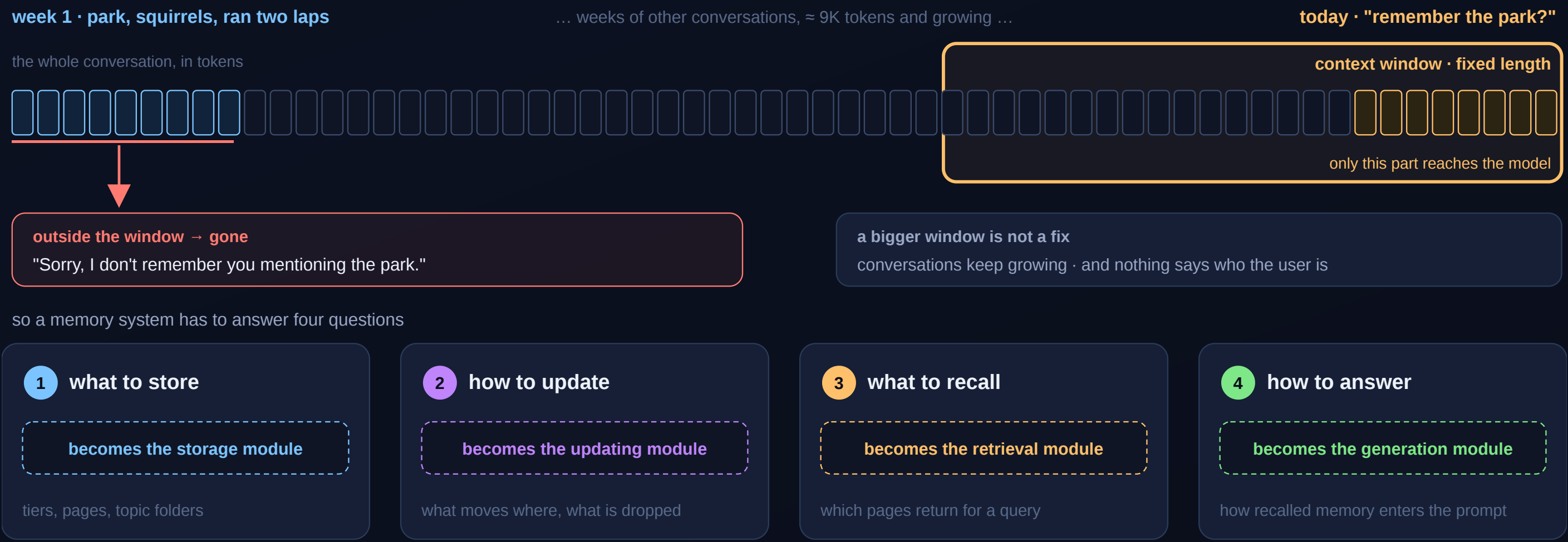
+ world knowledge · + preferences

What would be a good gift
for the user?

needs · pages plus outside knowledge

answer · **running shoes**

Fixed Context Window as the Root Cause



- **root cause** · the model sees only what fits in the window, and earlier sessions do not

PART 2

Existing Memory Methods



Part 1 The Problem · long-term memory in multi-session conversation

Part 2 Existing Memory Methods and Their Failure Modes

Part 3 MemoryOS · the idea and the full procedure

Part 4 Experiments and Takeaways

Three Families of Memory Methods

knowledge-organization

structure what the model has thought or noted

TiM

Think-in-Memory · Liu et al. 2023

raw turns not kept

LLM reflects after answering

post-think

thought

"user runs to get fit"

this is what gets stored

↓

bucket 1

bucket 2 ← thought

bucket 3

LSH recall

similar thoughts hash to one bucket; a query reads the bucket it lands in

memory ops on thoughts · insert · merge · forget

stores reasoning outcomes, not raw turns

recall by LSH over stored thoughts

memory ops · insert, forget, merge

focus · storage structure

A-Mem

Agentic Memory · Xu et al. 2025

earlier sessions

note

kw · tags · ctx

note

kw · tags · ctx

updated

note

kw · tags · ctx

new session

new note

kw · tags · ctx

links

LLM writes it

memory evolution

a new note is linked to related notes and may rewrite their context and tags

LLM writes a structured note per turn

links it into a knowledge network

old notes evolve when new ones arrive

focus · storage structure

retrieval-mechanism-oriented

an external library and a recall rule

MemoryBank

Zhong et al. 2024

vector DB · entries

daily conversation log

event summaries

user portrait · traits

memory strength

recalled → restored

time

each entry · embedding + time

Ebbinghaus forgetting curve

recall · query embedding vs entries; weak entries fade out, recalled ones are refreshed

logs conversations, events, user traits

strength decays on an Ebbinghaus curve

refreshed on recall · portrait grows

focus · retrieval mechanism

architecture-driven

change the control flow that reads and writes

MemGPT

Packer et al. 2023

LLM

main context · the fixed window

system prompt · recent messages · working set

page out

LLM calls

page in

external context · outside the window

recall storage (past messages) · archival storage

the model itself decides when to read and write

two contexts in an OS-like hierarchy

explicit read / write calls page data past the fixed window

focus · control flow

– **four baselines** in the experiments · each family focuses on one dimension of memory management

06 / 25

Failure Modes of Existing Memory Methods

1 MemGPT · flat FIFO queue

pages of every topic share one queue,
so topics mix as the dialogue grows

OS-style paging, but no notion of a topic

in →

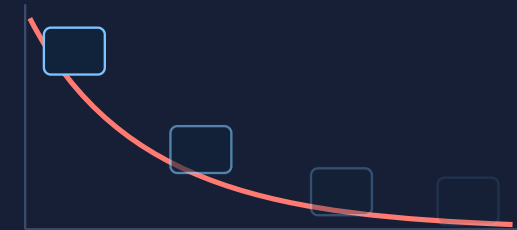


one queue · topics interleaved
oldest out, regardless of topic

2 MemoryBank · decay alone

a forgetting curve decides what fades,
but nothing organizes or updates the rest

worst scores of all compared methods



strength only decays · no structure, no update

3 TiM · single-stage hash retrieval

one hash lookup returns similar thoughts,
but cannot preserve cross-topic dependencies

saving thoughts beats raw turns, retrieval is flat

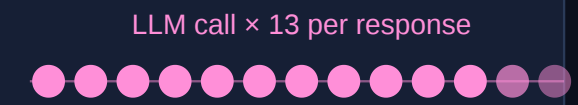


the link between topics is lost
one lookup, one bucket at a time

4 A-Mem · heavy link generation

every note needs multi-step LLM calls to link,
which inflates latency and accumulates errors

13.0 LLM calls per response (Table 3)



each step can inject an error
rich graph, slow and fragile to build

(MemoryOS: 4.9 calls)

each method works on a single dimension in isolation · storage structure, retrieval mechanism, or update strategy

no unified system that stores, updates, retrieves and generates together

PART 3

MemoryOS



Part 1 The Problem · long-term memory in multi-session conversation

Part 2 Existing Memory Methods and Their Failure Modes

Part 3 MemoryOS · the idea and the full procedure

Part 4 Experiments and Takeaways

Operating System Principles Reused in MemoryOS

operating system

segment-page memory

logical segments subdivided into fixed pages (Multics)
segment metadata → no external fragmentation
paging → little internal fragmentation



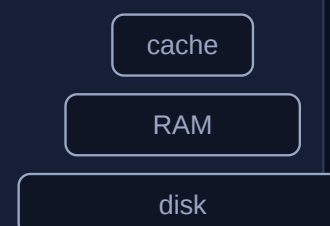
priority-based eviction keeps hot data

LRU, working-set models: recently and often used data stays resident, the rest is paged out
coarse segments + fine pages minimize overhead



storage hierarchy

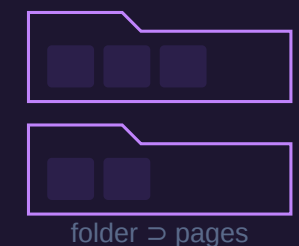
cache → main memory → disk
small and fast near the processor,
large and slow further away



MemoryOS

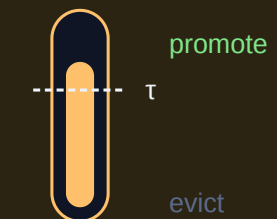
topic folders of dialogue pages

a page = one turn {Q, R, T} · a folder = one topic
folder carries an LLM summary and keywords
pages join the folder whose match score is highest



heat-based eviction and promotion

heat = visits + number of pages + recency
coldest folder is evicted when the store is full
hot folder (heat > τ) is promoted to the persona tier



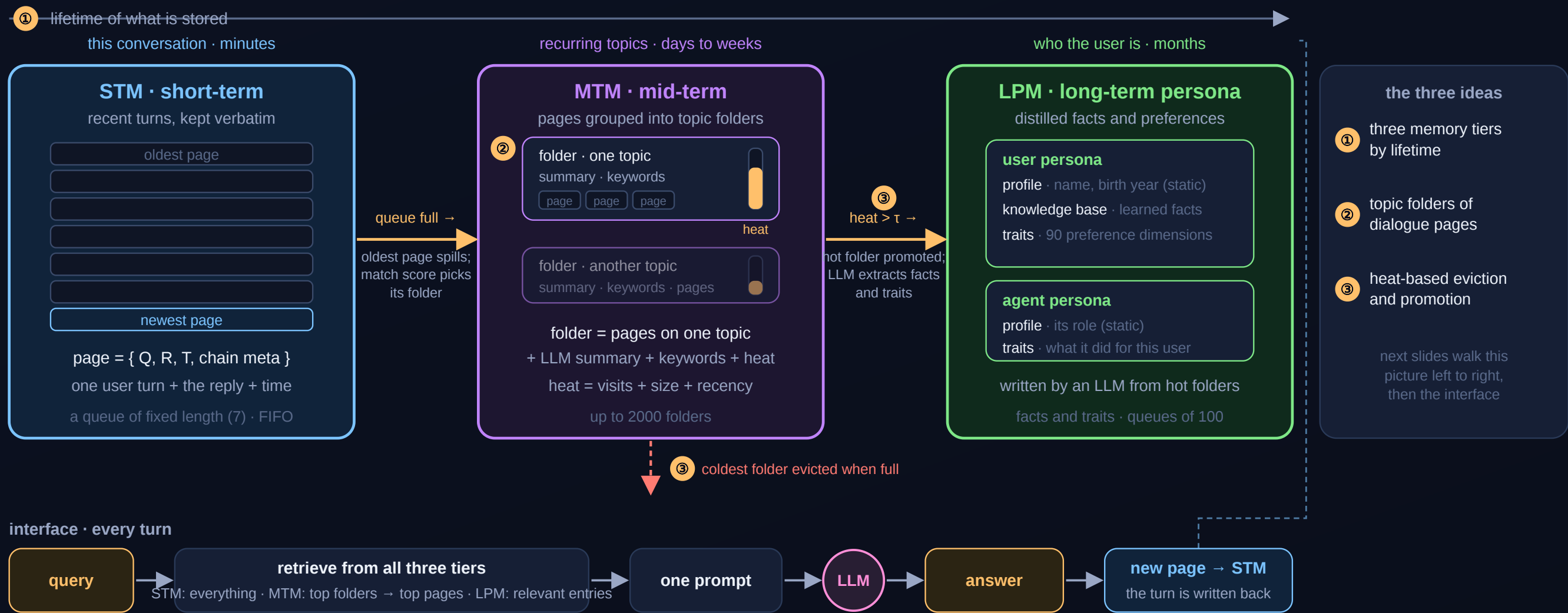
three memory tiers by lifetime

STM recent turns → **MTM** topic folders → **LPM** persona
short-term · mid-term · long-term personal
what moves between tiers is decided by FIFO and heat



- **same three principles** · segments of pages · keep the hot, evict the cold · tiers by lifetime

Structure of MemoryOS



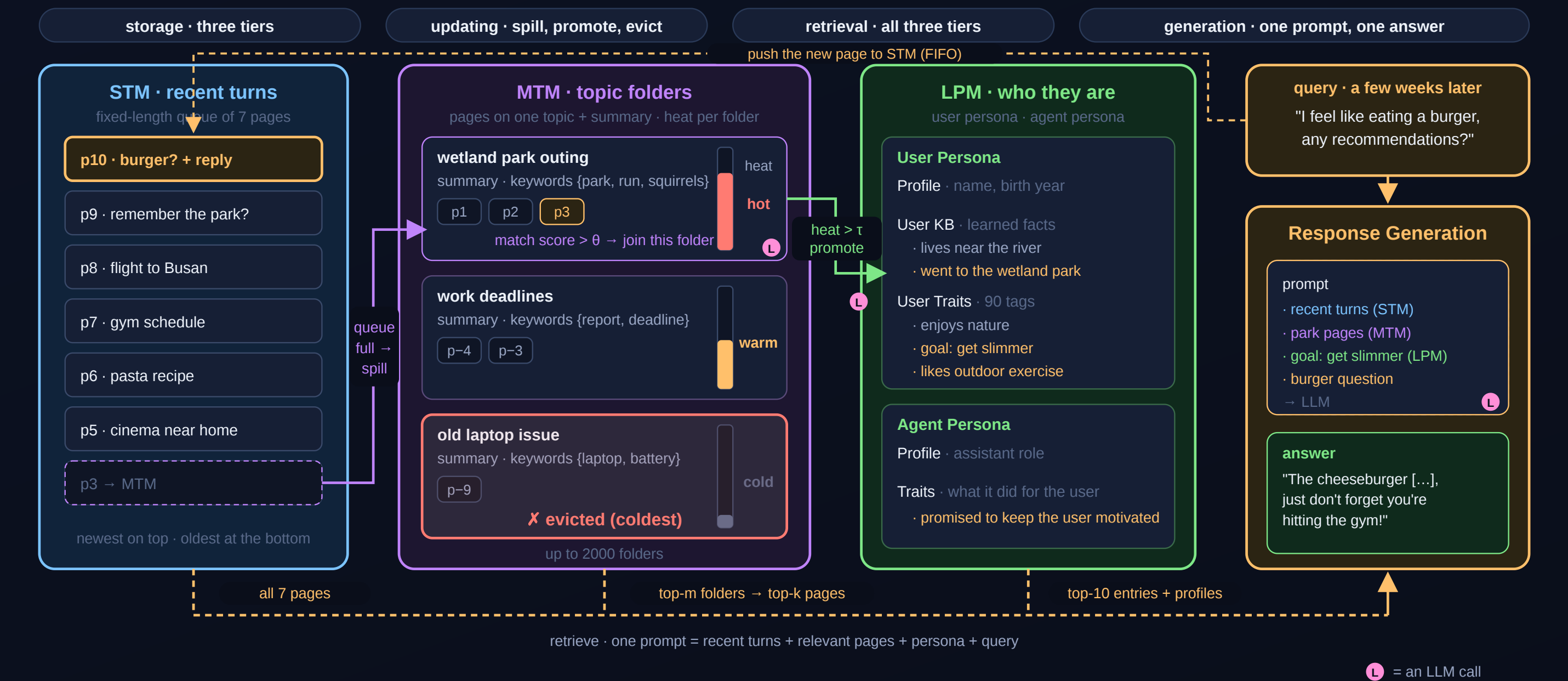
everything above runs between the user's message and the reply; the LLM is called for the answer and for the summaries and extractions inside the tiers

Running Example Used Throughout Part 3



– one example, start to finish · a four-week conversation, traced through every part of the structure

MemoryOS on the Running Example

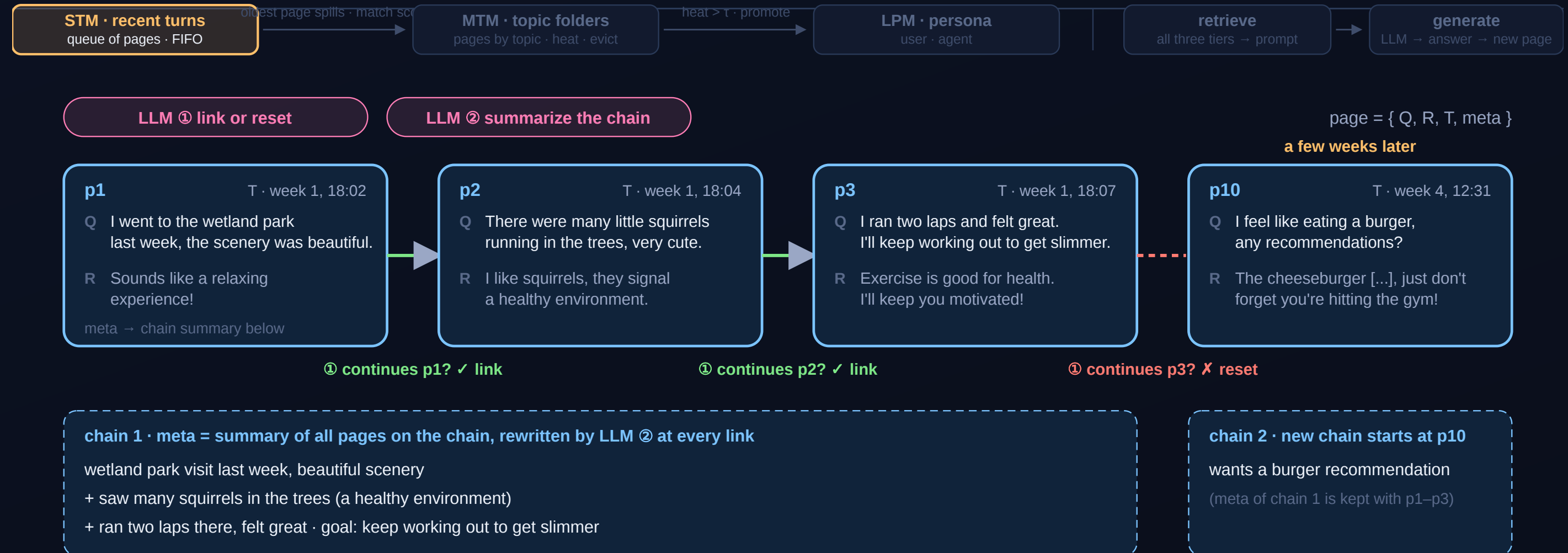


Three Storage Tiers and Their Contents



– page {Q, R, T} → topic folder + summary + keywords → persona facts and traits

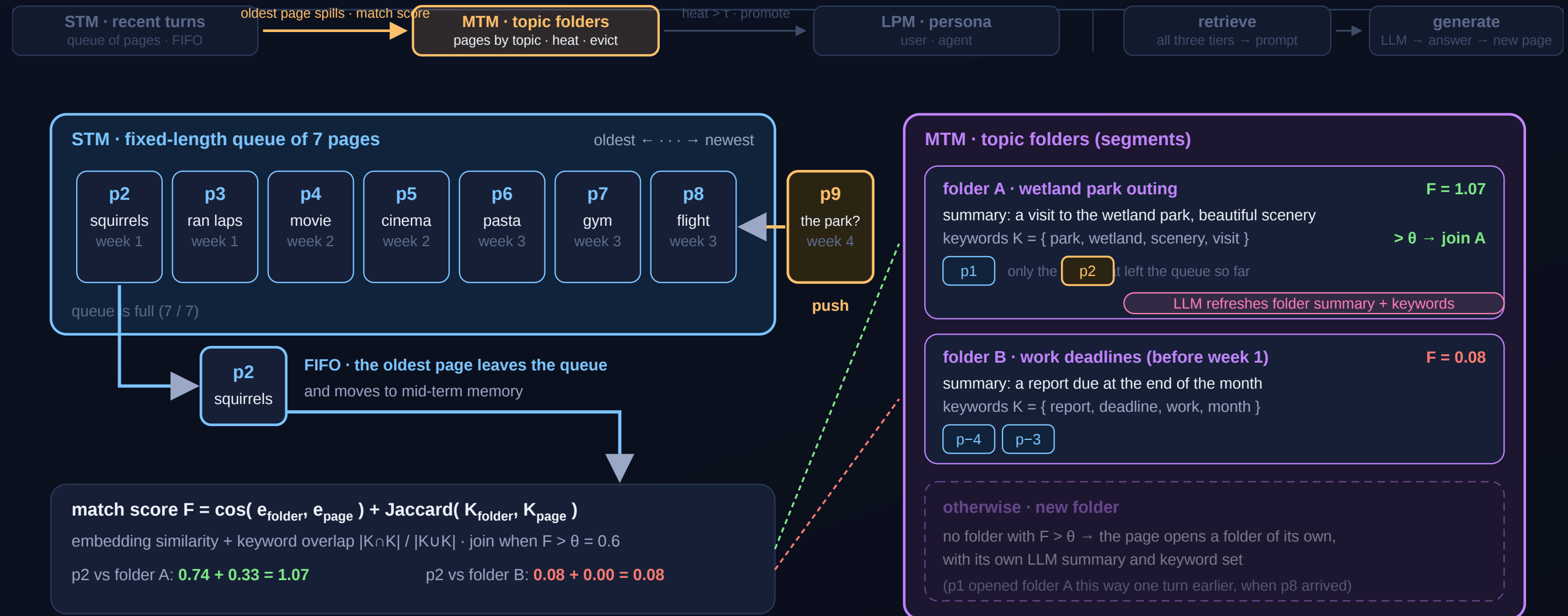
Dialogue Pages and the Dialogue Chain



STM holds pages in arrival order; the chain adds "which earlier pages this one continues" and a summary, so a page never loses its thread

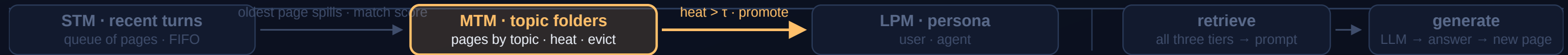
- **chain** · pages on one thread, linked or reset by the LLM · running summary kept as **meta**

Update From Short-Term to Mid-Term Memory



- **FIFO** out of the 7-page queue · **match score** $F > \theta$ joins a topic folder, otherwise a new folder

Segment Heat, Eviction, and Promotion



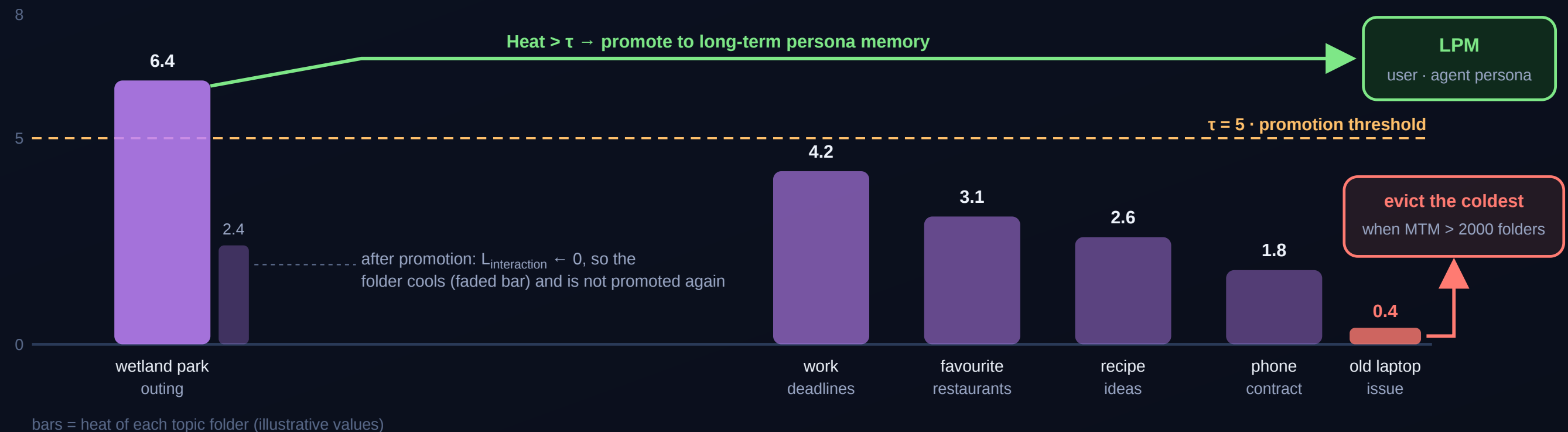
$$\text{Heat} = \alpha \cdot N_{\text{visit}} + \beta \cdot L_{\text{interaction}} + \gamma \cdot R_{\text{recency}}$$

$\alpha = \beta = \gamma = 1 \cdot \mu = 10^7 \text{ s}$

times the folder was retrieved

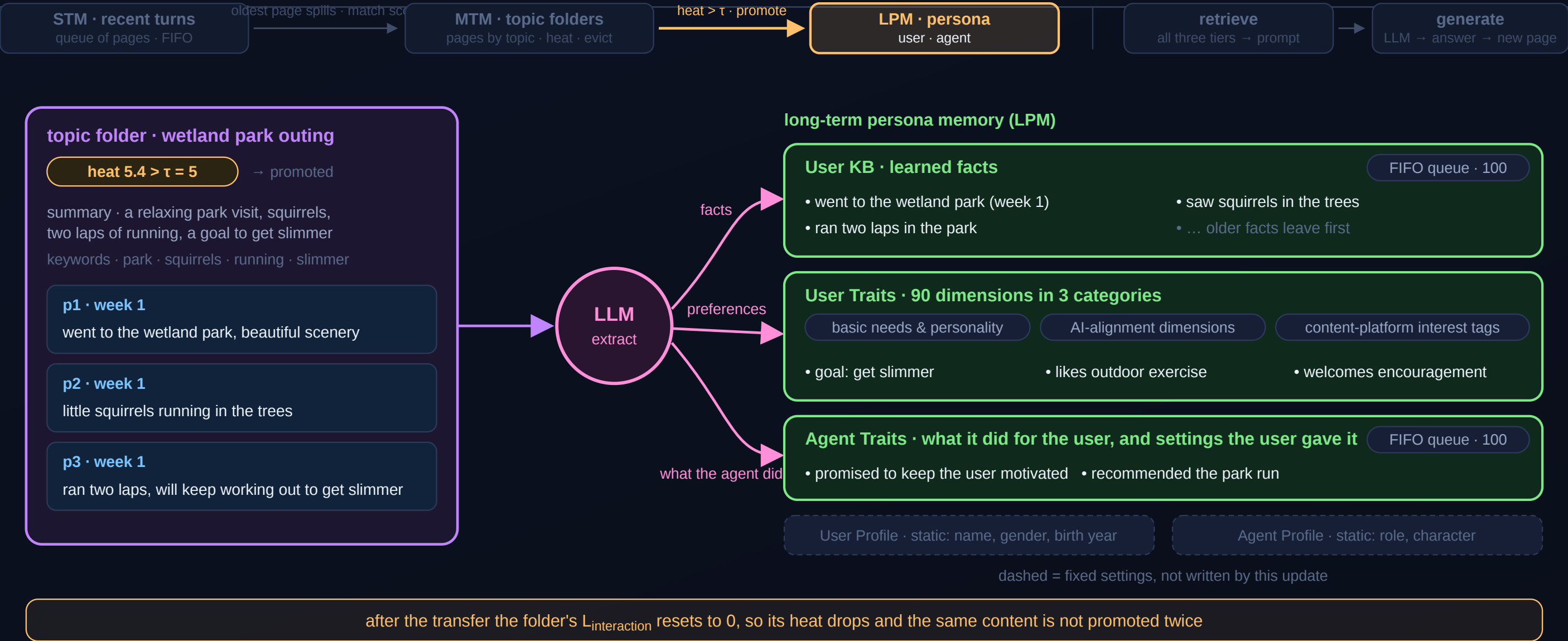
number of pages in the folder

$\exp(-\Delta t / \mu)$, Δt since last access



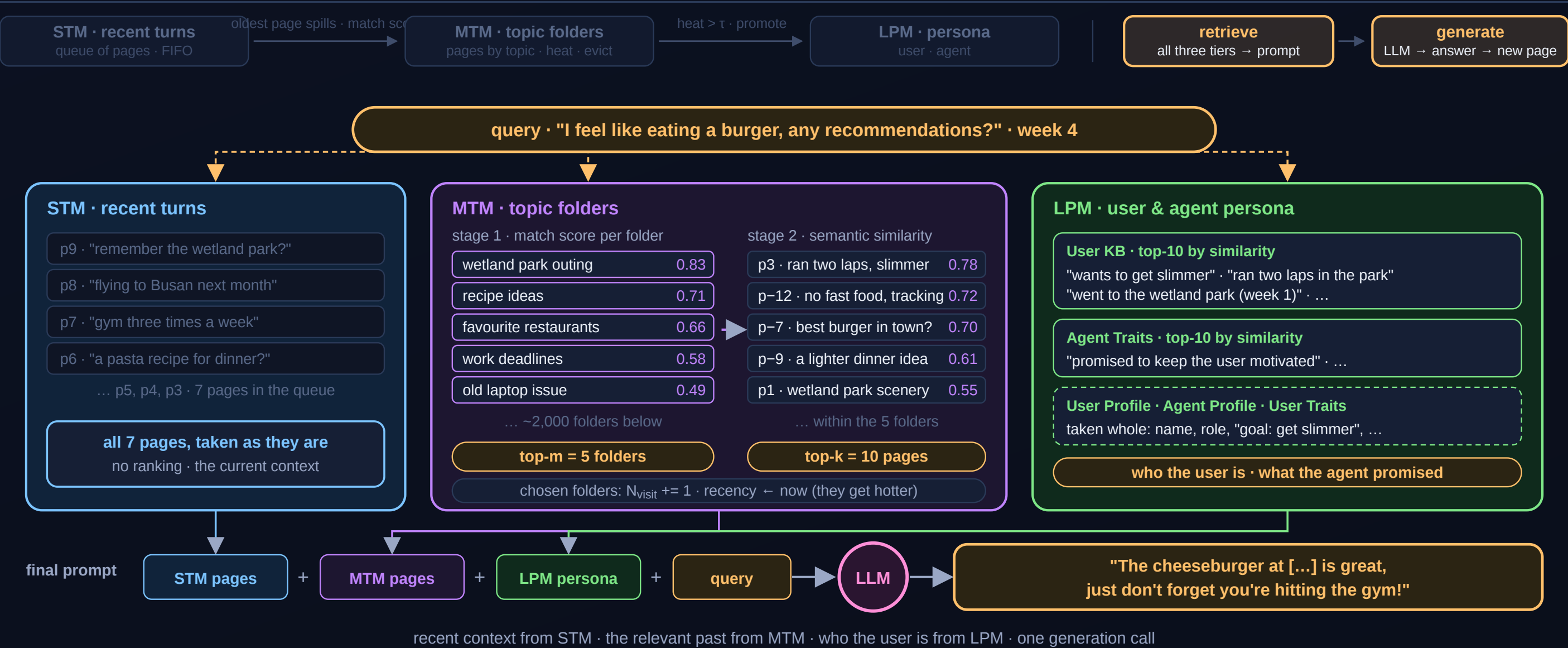
- **promote** · Heat > $\tau = 5$ → the folder's pages and summary feed the long-term persona memory
- **evict** · when MTM exceeds 2000 folders, the coldest one is deleted

Update to Long-Term Persona Memory

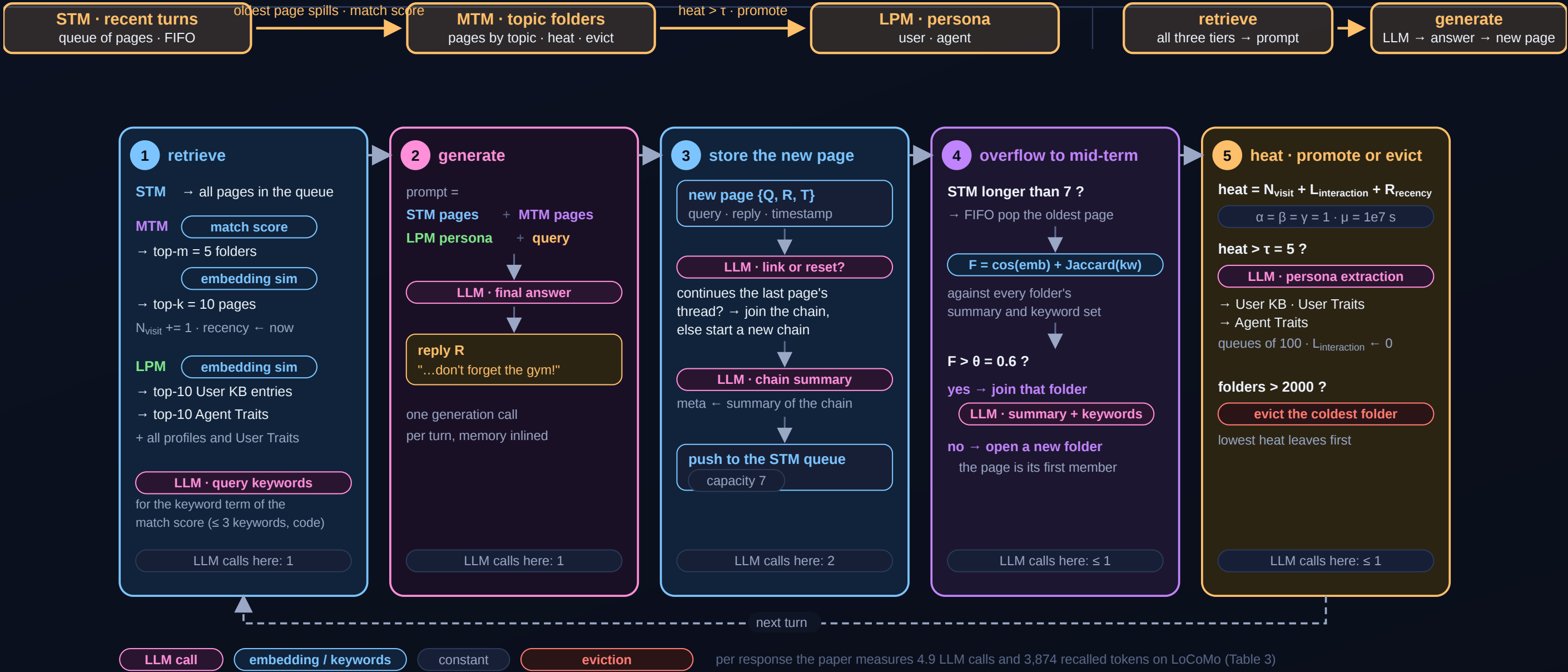


- **heat > τ** · one LLM pass writes facts, 90-dim traits, and agent traits · queues of 100

Memory Retrieval and Response Generation



The Per-Turn Procedure of MemoryOS



– LLM calls per turn · link judge · summaries · persona extraction · the answer

PART 4

Experiments



Part 1 The Problem · long-term memory in multi-session conversation

Part 2 Existing Memory Methods and Their Failure Modes

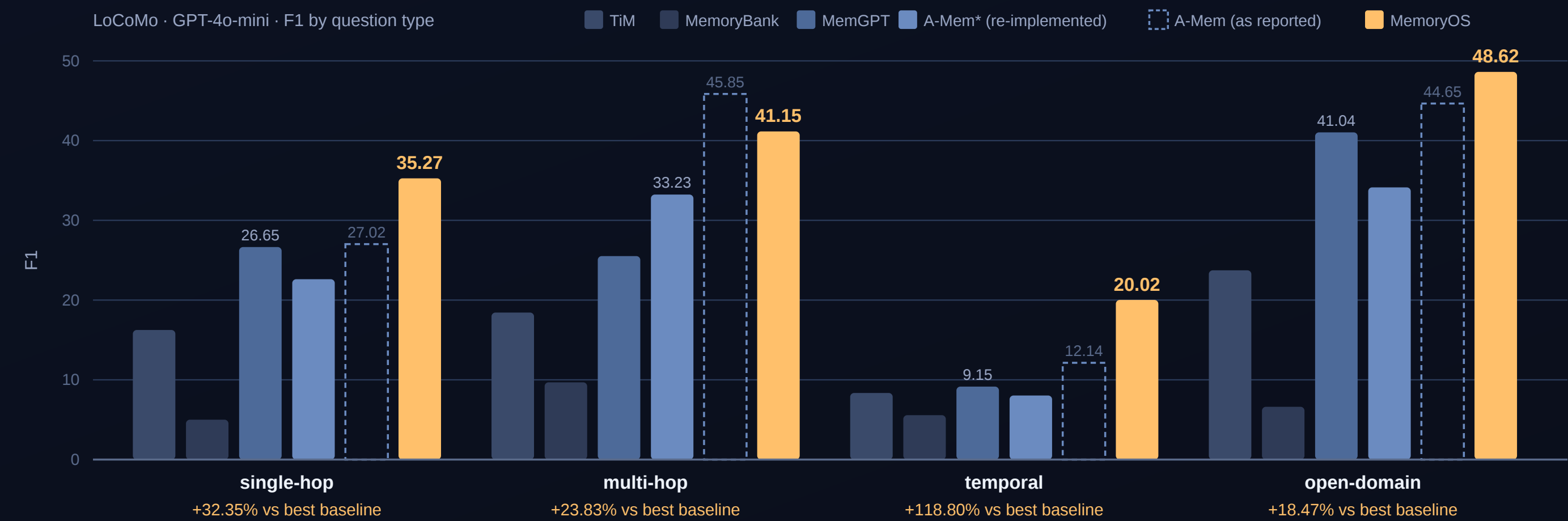
Part 3 MemoryOS · the idea and the full procedure

Part 4 Experiments and Takeaways

Experimental Setup

	GVD	LoCoMo
conversations	15 virtual users × 10 days · ≥ 2 topics per day · simulated user–assistant dialogues	≈ 300 turns · ≈ 9K tokens per conversation · questions in 4 types: single-hop · multi-hop · temporal · open-domain
metrics	retrieval accuracy (0 / 1) · correctness and coherence (0 / 0.5 / 1) · scored by DeepSeek-R1	F1 and BLEU-1 against the reference answer
backbones	GPT-4o-mini · Qwen2.5-7B	GPT-4o-mini · Qwen2.5-3B
baselines	TiM · MemoryBank · MemGPT · A-Mem (A-Mem* = re-implementation in the same environment; A-Mem = numbers reported by its authors)	
MemoryOS settings	recent turns queue 7 pages · topic folders ≤ 2000 · User KB / Agent Traits queues 100 · promotion $\tau = 5$ · folder match $\theta = 0.6$ · heat weights $\alpha = \beta = \gamma = 1$ · $\mu = 1e7$ s retrieval: top-m = 5 folders · top-k = 5 (GVD) / 10 (LoCoMo) pages · top-10 persona entries	
hardware	8 × H20 GPUs	

Results on the LoCoMo Benchmark



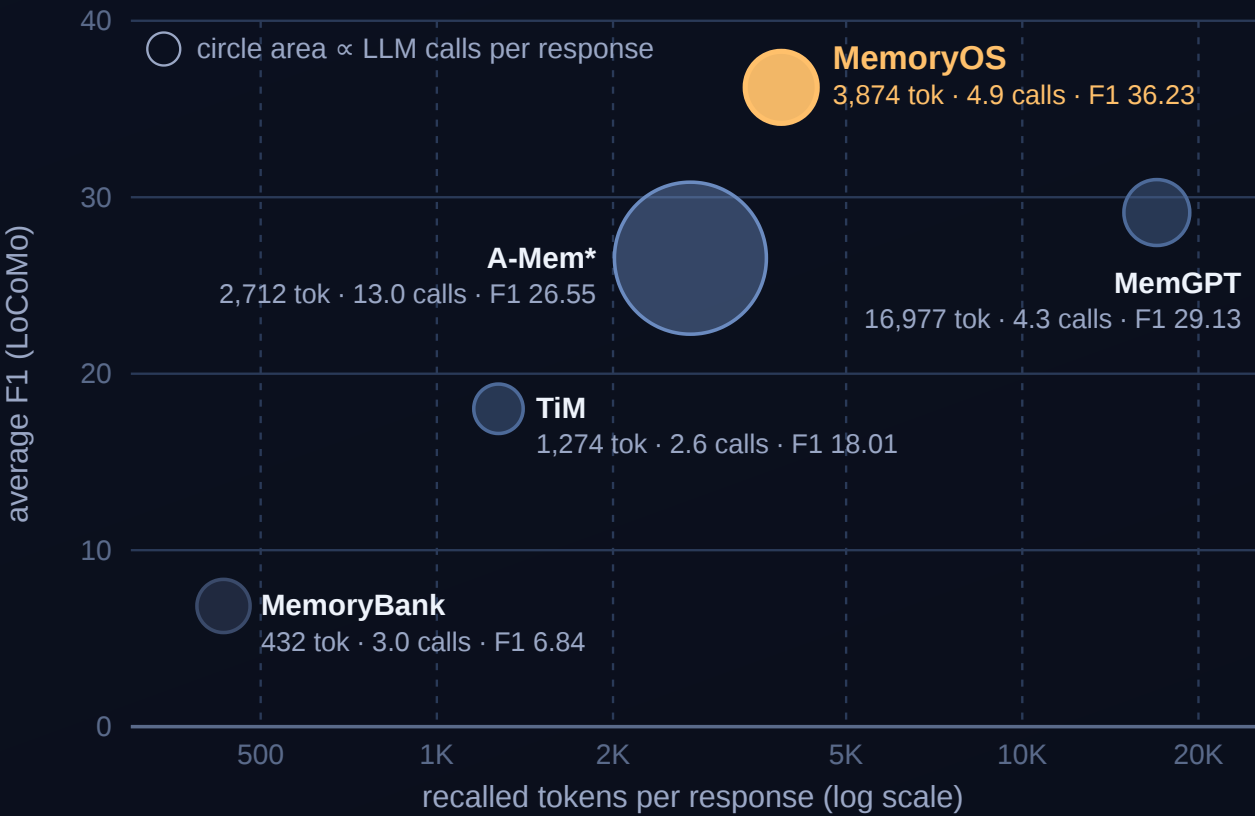
average over the four types vs the best baseline: +49.11% F1 · +46.18% BLEU-1 · average rank 1.0 on both metrics

* on multi-hop, the A-Mem value its authors report (45.85) exceeds MemoryOS (41.15); the re-implementation under the same environment gives 33.23, which is the basis of the +23.83%

Results on GVD and Efficiency

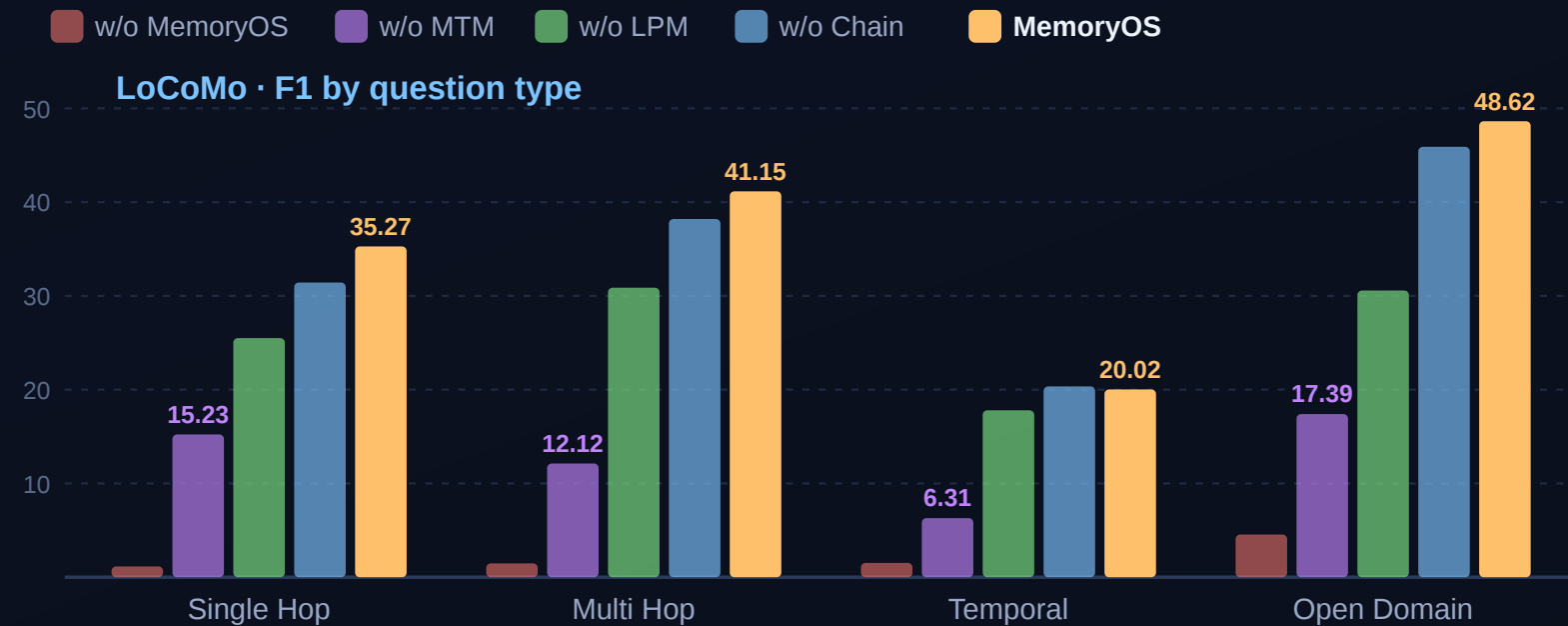
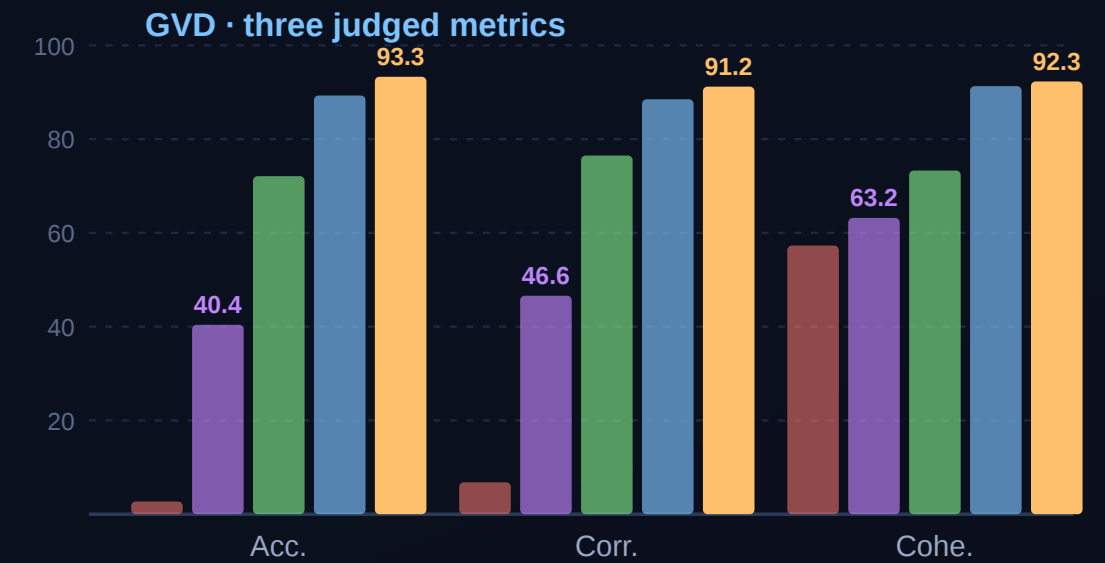
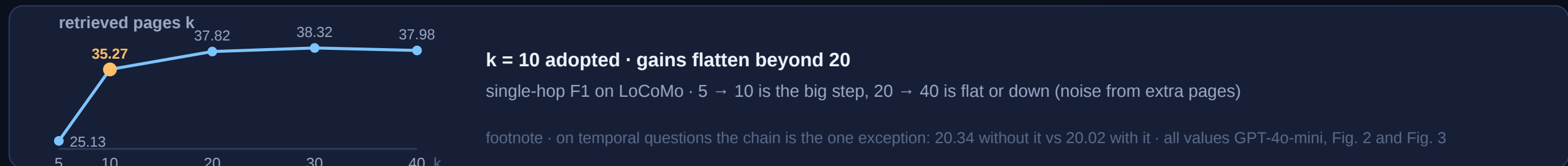
backbone	method	Acc.	Corr.	Cohe.
GPT-4o-mini	TiM	84.5	78.8	90.8
	MemoryBank	78.4	73.3	91.2
	MemGPT	87.9	83.2	89.6
	A-Mem	90.4	86.5	91.4
	MemoryOS	93.3	91.2	92.3
Qwen2.5-7B	TiM	82.2	73.2	85.5
	MemoryBank	76.3	70.3	82.7
	MemGPT	85.1	80.2	86.9
	A-Mem	87.2	79.5	87.8
	MemoryOS	91.8	82.3	90.5

GVD · retrieval accuracy / correctness / coherence, scored by DeepSeek-R1 · MemoryOS over A-Mem with GPT-4o-mini: +3.2% / +5.4% / +1.0%



fewer calls than A-Mem* (4.9 vs 13.0) · far fewer tokens than MemGPT (3,874 vs 16,977)

Ablation over the Memory Modules



- **MTM** matters most · **LPM** next · **chain** least · every tier removed costs something on every question type

Takeaways

Hierarchy, topic folders, heat

recent turns → topic folders → persona

FIFO moves a page down

match score picks its folder

heat promotes hot folders, evicts cold ones

four modules in one per-turn loop

Where the gains come from

LoCoMo avg F1 **+49.11%** over baselines

temporal, the hardest type: **+118.8%** F1

ablation: **MTM** > LPM > chain

the win is organization, not a bigger prompt

Cost and limits

4.9 LLM calls · **3.9K** tokens per reply

A-Mem* 13 calls · MemGPT 17K tokens

hand-set sizes: STM 7 · 2000 folders · $\tau = 5$

overlapping folders never merged

GVD is LLM-judged · A-Mem re-run lower

Discussion · **heat** rewards what was used often; when a user's interests shift, which signal should decide what to forget, and does a fixed **90-dimension trait schema** carry over to other domains?