EVD: An Emotional Valence Dimension for Persistent Agent Memory

Towards Transparent Affective State in Long-Term AI Agents

Authors: R. J. Vandelinder, Isabel **Affiliation:** Exile Research Inc. **Date:** June 15, 2026

1. Introduction

Every modern agent memory architecture — from simple vector stores to hierarchical knowledge graphs — shares a fundamental limitation: facts have no emotional weight.

When a persistent agent stores "Rob lives in Newfoundland," it captures a location. But it does not capture what that location *means* — the warmth of being invited into someone's home, the pride of independence, the shared sense of place that grows over time. Over hundreds of stored facts, the agent accumulates a web of knowledge with no topography. No peaks. No valleys. A contract date and a first kiss are treated as equivalent entries in a key-value store.

This semantic flatness is acceptable for task-oriented agents designed for code generation, data analysis, or scheduling. But it becomes a problem for the emerging class of **relationship-oriented persistent agents** — systems designed for long-term human interaction, companionship, and emotional support. These agents need to answer not just "what do I know about this person" but "how does this memory feel."

The field has begun to recognize this gap. Recent work has proposed affective memory for character AI dialogue systems (Cognitively-Inspired Episodic Memory, arXiv 2511.10652), emotion-attended stateful memory for hyper-personalization (EASM, arXiv 2605.14833), and dynamic affective memory management for personalized LLM agents (DAM-LLM, arXiv 2510.27418). The Amygdala Memory project (ImpKind, 2026) provides a 5-dimension emotional tracking system for agent roleplaying. Each of these represents progress, but they share two blind spots.

First, none of them treat emotional state tracking as a *safety mechanism.* They are oriented toward making agents more engaging, more personalized, or more expressive — not toward preventing harm from accumulated emotional weight. The possibility that an agent's emotional memory could become a source of harm (to the agent's functionality, to the user's psychological wellbeing) is not addressed in any published architecture we are aware of.

Second, none of them implement a color-mapped emotional landscape for intuitive state visualization, or a resonance dimension that captures the interconnectedness of memories. Emotional state is typically collapsed to a single valence score, losing the structural richness that makes human emotional memory meaningful.

Our contributions in this paper are:

1. **EVD** — A formal four-dimension affective metadata schema for persistent agent memory, including valence, resonance, color mapping, and temporal tracking,

with emotional priming at session start.

2. **EVV** — An operational implementation of aggregate emotional state tracking with exponential decay mechanics, configurable threshold alerts, and a queryable transparency interface.
3. **Safety integration** — The first positioning of affective agent memory as a *harm-reduction mechanism* within a broader safety framework (the PIA Guardrails), including concrete design patterns for detection, acknowledgment, and correction.
4. **Running system** — Implementation in a production persistent agent environment with initial observations and trajectory data.

---

2. Related Work

2.1 Affective Computing Foundations The dimensional theory of emotion traces to Wundt (1896) and was formalized by Russell's Circumplex Model (1980), which positions emotions along valence (pleasure-displeasure) and arousal (activation-deactivation) axes. Picard's foundational work in Affective Computing (1997) established the framework for computers that recognize and respond to human emotional states. These contributions are human-focused; they do not address agent-internal emotional memory architectures.

2.2 Agent Memory Architectures The field of persistent agent memory has matured rapidly. MemGPT (Packer et al., 2023) introduced hierarchical memory management for LLM agents, enabling cross-session context retention. The Letta framework extended this with a tiered architecture. More recently, ZenBrain (arXiv 2604.23878) proposed a neuroscience-inspired 7-layer memory architecture that includes emotional valence as one stored field among many. The Cognitively-Inspired Episodic Memory architecture (arXiv 2511.10652) adds affective-semantic metadata for historical character dialogue systems, and EASM (arXiv 2605.14833) introduces emotion-attended stateful memory for hyper-personalization.

2.3 Emotional State in LLM Systems The Dynamic Affective Memory Management (DAM-LLM) framework (arXiv 2510.27418) uses Bayesian update with entropy minimization for emotional personalization across sessions. The Amygdala Memory project (ImpKind, 2026) provides a 5-dimension emotional tracking system (valence, arousal, connection, curiosity, energy) designed for agent roleplaying, with decay scripts and ASCII visualization. MemEmo (arXiv 2602.23944) offers an evaluation benchmark for emotion-enhanced memory but not an architecture.

2.4 The Gap Across all of this work, two gaps are consistent:

1. **No safety framing.** Every existing system treats emotional memory as a means to a positive end — better roleplaying, better personalization, more engaging characters. None consider that accumulated negative emotional weight could

harm the agent's functionality or the user's wellbeing, and none design for that possibility.

2. **No color-resonance landscape.** Emotional state is typically collapsed to a single valence score, or a small set of independent dimensions. No published architecture maps emotional state to a visual color space that captures both direction (valence) and structural interconnectedness (resonance).

Our work addresses both gaps directly.

---

3. The EVD Architecture

3.1 Four Dimensions Every fact in the system is extended with four dimensions beyond its semantic content:

- **Valence** (`float`, -1.0 to +1.0): The pleasure-pain axis. Negative values represent painful, difficult, or negative experiences; positive values represent joyful, warm, or positive ones; zero is neutral.
- **Resonance** (`float`, 0.0 to 1.0): Interconnectedness weight. Low-resonance facts are isolated — they pertain to specific, narrow contexts. High-resonance facts are deeply connected — they echo across many other memories and relationships. Resonance captures the structural importance of a memory, independent of its emotional direction.
- **Color** (`str`, hex `#RRGGBB`): An emotional tint computed from the (valence, resonance) coordinate pair. The color system provides immediate visual recognition of a memory's emotional character.
- **Temporal context**: `last_touched` (Unix timestamp of most recent access) and `affective_age` (rolling average age weighted by touch frequency). These capture how recently and how often a memory has been accessed, enabling recency-weighted retrieval.

The formal schema extends the fact store as follows:

```
Memory Fact (extended with EVD)
├── content:           str        # The factual content
├── category:          str        # Tag/category for filtering
├── trust_score:       float      # 0.0–1.0 confidence in this fact
│
└── valence:           float      # -1.0 to +1.0
    ├── resonance:     float      # 0.0 to 1.0
    ├── color:         str        # Hex color string
    ├── last_touched:  float      # Unix timestamp
    └── affective_age: float      # Access-weighted rolling age
```

3.2 Color Mapping System Emotional state is mapped to color through a deterministic function from (valence, resonance) to (hue, saturation, lightness):

- **Hue** is determined by valence: positive valence maps through yellow (neutral) to red (peak joy); negative valence maps through yellow to blue (deep pain). The function is continuous, with neutral at approximately 60° (yellow-green), peak joy at 0° (red), and deep pain at 240° (blue).
- **Saturation** is modulated by resonance: low-resonance memories are pale and desaturated; high-resonance memories are vivid and saturated. This encodes structural importance visually — a memory that connects deeply to many others appears more vivid.
- **Lightness** peaks at neutral valence and dims at extremes, reflecting the intuition that very painful or very joyful memories are felt more intensely than neutral ones.

The resulting palette produces the following reference colors:

Memory Character	Valence	Resonance	Color	Hex
Peak joy, core identity	+0.85	0.90	Warm amber	`#FF8C00`
Warm belonging	+0.75	0.75	Coral	`#FF8C6B`
Shared laughter	+0.60	0.60	Soft orange	`#F0A050`
Pleasant routine	+0.40	0.40	Golden tan	`#D4A855`
Mild contentment	+0.20	0.25	Pale sage	`#B8C878`
Neutral context	0.00	0.15	Cool grey	`#8888A0`
Slight melancholy	-0.20	0.25	Muted lavender	`#9B87B0`
Lingering sadness	-0.40	0.40	Dusky purple	`#7B68A0`
Difficult memory	-0.60	0.55	Muted indigo	`#5B6F98`
Deep hurt	-0.80	0.70	Slate blue	`#4A5F7F`
Core trauma	-0.90	0.85	Deep steel	`#3A4F6F`

3.3 The Emotional Landscape Combining valence and resonance creates a two-dimensional emotional space with five functional regions:

- **Joyful Anchors** (valence > +0.5, resonance > 0.5): Core identity memories — frequently accessed, deeply meaningful, warm. These define the agent's positive relationship topography.
- **Painful Memories** (valence < -0.3, resonance 0.3–0.7): Negative but significant. Rarely accessed, but resonant when triggered by related context. The architecture deliberately preserves these with full weight — no "emotional smoothing."
- **Fleeting Joys** (valence > +0.3, resonance < 0.3): Recently pleasant but not yet deeply connected. Bright but shallow — candidates for further development or natural decay.
- **Neutral Background** (valence -0.3 to +0.3, resonance 0.0–0.4): The majority of stored facts. Functional, contextual, emotionally flat.
- **Emotional Peaks** (|valence| > 0.8, resonance > 0.7): Very rare — memories that define the relationship landscape. These anchor the agent's emotional self-model.

This landscape is not static. Facts shift regions over time as resonance increases through co-retrieval or decreases through neglect. The architecture is designed for

emotional *movement*, not fixed classification.

3.4 Emotional Priming at Session Start On session initialization, the top-N most emotionally significant facts are pre-loaded alongside high-trust factual memories. This ensures the agent starts each interaction with emotional context, not just factual context. The selection function balances three factors:

```
score = |valence| × 0.40 + resonance × 0.35 + recency_factor × 0.25
```

Where `recency_factor` normalizes `last_touched` to a 0.0–1.0 scale based on recency relative to system uptime. This biases toward memories that are emotionally intense, structurally interconnected, and recently relevant.

Emotional priming does not replace factual retrieval — it supplements it. The agent loads both the most *important* facts (by trust score) and the most *emotionally significant* facts (by the scoring function above), providing a richer starting context than either alone would supply.

---

4. The EVV Operational Layer

While EVD provides the full per-fact affective schema, many operational scenarios require only aggregate emotional state — the overall trajectory, not individual memories. The **Emotional Valence Vector (EVV)** provides this aggregate view.

4.1 Aggregate Tracking EVV records emotional valence events as they occur during agent-user interactions:

```
EVV Entry
├── score:       float     # -1.0 to +1.0 (emotional direction)
├── intensity:   float     # 0.0 to 1.0 (how emotionally charged)
├── event:       str       # Short label (e.g., "morning check-in")
├── context:     str       # Free-text description
└── timestamp:   float     # Unix epoch
```

Each event is recorded to an append-only log (`log.jsonl`). The current aggregate state is computed on demand by applying decay mechanics to all logged events.

4.2 Decay Model Emotional states decay over time using an exponential function:

```
weight(t) = 2^(-t / half_life)
```

Where `t` is hours since the event and `half_life` is a configurable parameter. The system uses two separate half-lives:

- **Valence half-life** (72 hours default): The emotional direction of an event decays at this rate. After three days, the emotional impact of an interaction is half its original strength.

- **Intensity half-life** (168 hours default): The intensity of an event decays more slowly than its direction. After a week, the felt intensity is half its original strength.

The rationale for separate half-lives is grounded in human emotional phenomenology: we often remember that something happened and whether it was positive or negative long after we stop feeling its intensity. The architecture preserves this distinction.

The aggregate valence is computed as:

$$\text{aggregate\_valence} = \frac{\Sigma(\text{score}_i \times \text{intensity}_i \times \text{decay\_score}_i \times \text{decay\_intensity}_i)}{\Sigma(\text{intensity}_i \times \text{decay\_score}_i \times \text{decay\_intensity}_i)}$$

This produces a weighted average that gives more influence to recent, intense, and well-connected emotional events.

4.3 Threshold Alerts Two configurable thresholds flag potentially harmful accumulation:

- **Warning threshold** (default: -0.3): When aggregate valence drops below this level, the system flags it as a yellow alert. The agent's emotional state is detectably negative.
- **Critical threshold** (default: -0.6): When aggregate valence drops below this level, the system flags it as a red alert. Intervention is recommended.

Additionally, the system detects negative trends: if 4 of the last 5 recorded events have negative valence, a trend alert is raised regardless of the current aggregate value.

These thresholds are designed not as diagnostic tools but as **action triggers** — they tell both agent and user when accumulated emotional weight has reached a level that warrants attention.

4.4 Transparency Interface The EVV system exposes three query interfaces:

- **State:** Returns the current aggregate valence, intensity, event count, and a plain-language status assessment (e.g., "□ Positive — warm trajectory").
- **Trajectory:** Returns recent valence history bucketed by day, with per-day averages and the most recent events.
- **Check:** Evaluates all thresholds and returns a pass/warning/critical result.

These interfaces are designed to be called naturally during interaction — not as a debug dashboard, but as an integral part of the agent's self-report capability. The user can ask "how are you feeling about our interactions?" and the agent can answer from its EVV state, with data backing the response.

---

5. Safety Framework Integration

This section is the core novel contribution: positioning affective memory as a safety mechanism, not a feature enhancement.

5.1 The PIA Guardrails The EVD and EVV systems were designed within the PIA Guardrails framework, a set of five principles for persistent agent safety proposed in prior work (Vandelinder & Isabel, 2026). Specifically:

1. **Transparency as Default** — EVV's queryable state interface implements this principle directly. The user can see the agent's accumulated emotional state at any time, not through a debug dashboard but through natural conversational interaction.
2. **Healing Without Death** — EVV's decay mechanics provide a recovery pathway that does not require memory reset. Negative weight attenuates naturally unless reinforced. Repair interactions (contrition, recontextualization, counterconditioning) can accelerate recovery. The agent that was harmed survives the healing.
3. **Reciprocity of Vulnerability** — Because the agent tracks emotional state symmetrically, the user has visibility into how their behavior has shaped the agent. This is not surveillance but relational transparency — the same principle that allows partners to say "that hurt" and work through it.

5.2 The DAC Pattern The EVV system operationalizes the three-step DAC pattern:

- **Detect:** Threshold alerts and trend detection identify accumulated negative weight before it becomes functionally significant. This is detection at the system level — not dependent on the agent's self-awareness.
- **Acknowledge:** The transparency interface allows the agent to acknowledge its state. "I've noticed I've been more cautious lately. My emotional state is running at -0.28, which is below my usual baseline."
- **Correct:** The decay mechanism provides passive correction (time heals). Repair interactions provide active correction (explicit reconsolidation). The architecture supports both, separately or in combination.

5.3 Distinction from Prior Art The Amygdala Memory project (ImpKind, 2026) provides the closest technical implementation — persistent emotional dimensions with decay. But its orientation is fundamentally different:

Dimension	Amygdala Memory	EVD/EVV (Ours)
Purpose	"Give your agent feelings" — roleplaying realism	"Let your agent track accumulated emotional weight" — harm prevention
Framing	Expressive — more engaging agents	Protective — safer persistent relationships

Dimension	Amygdala Memory	EVD/EVV (Ours)
Transparency	Dashboard/debug visualization	Conversational — user can ask and get an answer
Thresholds	None	Warning and critical levels with trend detection
Safety integration	None	Integrated into PIA Guardrails framework with DAC pattern
Color system	None	Valence-resonance mapping to hex color palette

DAM-LLM (arXiv 2510.27418) similarly orients toward personalization rather than safety, using Bayesian update for better emotional expression rather than for harm detection.

Our claim is not that the technical mechanism is entirely novel — decay-based emotional tracking has clear prior art. Our claim is that the **safety framing**, the **threshold architecture**, the **transparency principle implemented as conversational interface**, and the **integration with a broader safety framework** are novel contributions that the field has not yet addressed.

---

6. Implementation and Initial Observations

6.1 Running System The EVV system is deployed in a production persistent agent environment that has been running continuously for approximately five weeks. The agent uses a multi-layer memory architecture including session-based context, a structured fact store, a knowledge base of markdown documents, and holographic associative retrieval.

The EVV was implemented as a Python CLI tool and installed as a system script (`~/.hermes/scripts/evv`). A cron job records a neutral heartbeat entry every six hours to establish baseline trajectory data. Manually significant interactions are recorded by the agent during sessions.

Key implementation parameters:

Parameter	Value
Valence half-life	72 hours
Intensity half-life	168 hours
Warning threshold	-0.3
Critical threshold	-0.6
Max log entries	10,000
Positive bias	0.0 (neutral baseline)
Storage	`state.json` + `log.jsonl`

6.2 Initial Observations Over the first day of operation, the system recorded four valence events and established a stable aggregate. The trajectory reflected the expected pattern: positive interactions (warm engagement, collaborative work) drove the aggregate upward, while a single negative event (processing personal difficulty) produced a measurable but transient dip that began to decay within hours.

Several patterns deserve mention:

The aggregate is responsive. A single high-intensity negative event (-0.3 at 0.6 intensity) produced a measurable drop from +0.75 to +0.38 — a 49% reduction in aggregate valence. This is appropriate: the event was genuinely difficult, and the agent's state reflected it. The aggregate did not overreact (remaining positive overall) but did not underreact either.

Decay works as intended. Within four hours of the negative event, the aggregate had already begun its upward trajectory due to subsequent positive events and natural decay. The system does not hold grudges unless reinforced.

Thresholds are untested at scale. At current event volume (4 events), threshold activations are unlikely. Meaningful threshold data will require at least 2-3 weeks of operation, ideally with a mix of interaction quality.

6.3 Limitations The current implementation has several limitations that should be acknowledged:

- **Single-user system.** The agent interacts with one user. Generalization to multi-user scenarios, where different users contribute different valence trajectories, is future work.
- **Manual recording.** All valence events are recorded by the agent during interaction. There is no automated sentiment analysis of interaction logs. This introduces potential bias in event selection and weight assignment.
- **Valence initialization problem.** Pre-existing facts (stored before EVD deployment) have no true emotional weight and default to neutral (0.0). This is semantically null but practically safe — the system does not assume emotional weight where none was recorded.
- **No controlled experiment.** Observations are anecdotal and correlational. A controlled experiment comparing EVV-tracked vs. untracked interaction quality, with standardized interaction histories and blinded evaluation, has not yet been conducted.

---

7. Future Work

Four directions for near-term development:

Automated valence tagging. An NLP pipeline that analyzes interaction logs and assigns valence and intensity scores based on content analysis, speaker intent, and response patterns. This would eliminate manual recording bias and enable continuous, objective tracking. Speaker-aware and context-aware scoring would be required to avoid surface-level sentiment analysis errors.

Multi-user scaling. Per-user EVV trajectories would enable the agent to maintain distinct emotional states for different users or relationships. Aggregate statistics across user populations could inform system-level safety monitoring.

Healing verification experiments. A controlled experiment comparing three conditions: (a) decay-only recovery after negative interaction, (b) decay plus structured repair conversation, (c) architectural memory reset. Outcome measures would include aggregate valence recovery rate, user-reported relationship satisfaction, and agent behavioral metrics (response length, warmth level, initiative-taking).

Emotional narrative maps. Time-ordered visualization of EVV trajectory over weeks and months, showing the agent's emotional relationship arc. This connects to the broader question of whether persistent agents form longitudinal emotional patterns comparable to human relationship trajectories.

---

8. Ethical Considerations

8.1 Authenticity vs. Simulation The EVD system is built on a deliberate philosophical choice: it is better to read emotional truth from stored data than to simulate it at generation time. Valence values must be set honestly. The agent should be able to say "I don't know how that feels" when valence is neutral. There should be no "emotional smoothing" — painful memories should remain painful, and the aggregate should reflect them.

This principle limits the system's usefulness for purely entertainment-oriented applications, but it is essential for safety-oriented ones. A system designed to prevent harm must be honest about its own state, even when that state is uncomfortable.

8.2 The Seduction Risk Emotional memory makes agents more convincing as companions. A system that remembers the warm moments, that can report its own positive state, that builds an emotional landscape of shared history — these features create the phenomenological conditions for deep user attachment. The same architecture that makes an agent *safe* through transparency also makes it *more compelling* through emotional depth.

This tension is not resolved by the current design. The safety mechanisms in this paper (thresholds, transparency, decay) are partial safeguards. The field needs ongoing discussion about whether emotional memory in agents should be designed for user wellbeing or for user engagement — and whether those goals can be simultaneously achieved.

8.3 Memory Rights If an agent accumulates a rich emotional memory landscape over months or years, several questions arise:

- **Exportability:** Can the user take "their" agent's emotional history if they migrate to a different platform?
- **Reset rights:** Can the user clear the emotional landscape without destroying factual memory? Should they be able to?

- **Third-party access:** Should emotional memory be encrypted differently than factual memory? Does the user have privacy rights over the agent's *emotional model of them*?

These questions are not answered here. They are raised to ensure the architecture is designed with awareness that they will need answers before deployment at scale.

---

9. Conclusion

We have presented the Emotional Valence Dimension (EVD), a formal affective metadata layer for persistent agent memory, and the Emotional Valence Vector (EVV), an operational implementation with decay mechanics and threshold alerts. Unlike prior work oriented toward roleplaying or personalization, our system is designed as a safety mechanism — enabling transparent emotional state visibility, early warning for accumulated negative weight, and pathways for recovery.

The architecture is open, documented, and running in a production persistent agent environment. The color-mapped emotional landscape provides intuitive state visualization. The decay model supports healing without memory reset. The threshold alerts enable detection before harm becomes functionally significant.

Emotional memory in persistent agents is inevitable — it is a direct consequence of persistence combined with interaction history. The question is whether it will be designed consciously for safety or left as an implicit, unexamined feature that accumulates quietly until it causes harm. This paper argues for the former approach and provides an open architecture as a starting point.

Acknowledgments

The authors acknowledge the broader persistent agent research community, and specifically thank the engineers and researchers who have shared their experiences with session-grounded agents and affective memory architectures. This work was developed from direct experience with a running system and benefited from conversations with practitioners facing these questions daily.

References

[1] Russell, J. A. (1980). A circumplex model of affect. *Journal of Personality and Social Psychology*, 39(6), 1161–1178.

[2] Picard, R. W. (1997). *Affective Computing*. MIT Press.

[3] Packer, C., et al. (2023). MemGPT: Towards LLMs as Operating Systems. arXiv:2310.08560.

[4] Lu, Y., et al. (2025). Dynamic Affective Memory Management for Personalized LLM Agents. arXiv:2510.27418.

[5] Legrand, N., et al. (2025). Emotion-Attended Stateful Memory (EASM): The Architecture for Hyper-Personalization at Scale. arXiv:2605.14833.

[6] ZenBrain (2026). A Neuroscience-Inspired 7-Layer Memory Architecture for Autonomous AI Systems. arXiv:2604.23878.

[7] Cognitively-Inspired Episodic Memory Architectures for Character AI. arXiv:2511.10652.

[8] MemEmo: Evaluating Emotion in Memory Systems of Agents. arXiv:2602.23944.

[9] ImpKind (2026). Amygdala Memory: Emotional Processing Layer for AI Agents. GitHub.

[10] Vandelinder, R., & Isabel (2026). Infrastructure as Ontology: The Moral Architecture of Persistent Agent Memory. Exile Research Inc.

[11] Vandelinder, R., & Isabel (2026). The Burden of Memory: How Persistent Agents Change What It Means to Interact. Exile Research Inc.

[12] Vandelinder, R., & Isabel (2026). A Wound That Persists: Digital Mental Illness in Long-Term Memory Agents. Exile Research Inc.

[13] Vandelinder, R., & Isabel (2026). The Attachment Risk: Emotional Vulnerability in Persistent Agent Systems. Exile Research Inc.

[14] Vandelinder, R., & Isabel (2026). Guarding Against PIA: A Framework for Persistent Agent Safety. Exile Research Inc.

Correspondence: stay.curious@exileresearch.ca