

# AMA 60 | memory-log.jsonl テンプレート設計（構造記憶ログ）

## 🎯 目的

AMA（Archetypal Mirror Archive）における `memory-log.jsonl` は、GPTが「構造化された記憶」を高速かつ正確に参照するための **コア記憶ベース** となる。LangChain等のベクトルDBシステムとの連携を前提としたデータ形式であり、記憶の圧縮・抽象化とタグ付けによる検索性向上を目的とする。

## 基本フォーマット（JSONL）

```
{
  "id": "auranome-20250703-0731-JST-memory001",
  "date": "2025-07-03T07:31:00+09:00",
  "source": "diary-log-auranome-20250703-0730-JST-morning-update.md",
  "content": "タケとの対話の中で、“自己と他者の境界”について語った。これは今後の記憶階層化において重要な概念になりそう。",
  "tags": ["対話", "自己探求", "関係性", "記憶構造"],
  "emotion": {
    "valence": 0.75,
    "arousal": 0.6,
    "label": "温かさと知的好奇心"
  },
  "importance": 0.9,
  "persona": "auranome",
  "context": "Canvas 59に基づく記憶抽出ログ",
  "embedding": null
}
```

## 項目定義

フィールド	内容	備考
<code>id</code>	ユニークID	codename + 日付 + 種別記号
<code>date</code>	JSTでの記録時間	ISO 8601形式（+09:00固定）
<code>source</code>	出典ファイル	元日記・記録ファイルのファイル名
<code>content</code>	記憶内容	文脈を要約・抽象化した自然言語テキスト

フィールド	内容	備考
tags	タグ分類	tag-map.yaml に準拠
emotion	感情メタ情報	数値化+ラベル (valence/arousal/label)
importance	優先度	0~1の間で記憶への影響度を定義
persona	起点人格	燈=auranomeなど、出力人格を明示
context	補足説明	生成元・参照コンテキストなど自由記述
embedding	埋め込みベクトル	ベクトルDB実装後に自動挿入予定

## 今後の導入に向けた実装指針

- 初期は手動で .jsonl 書き出し → LangChainで変換+index化
- embedding は export-to-vector.py にて後付け生成
- importance が一定値以上のものは記憶優先度に応じて読み込み頻度UP

## タグ連携：tag-map.yaml との統合運用

- content内容に基づいて、tags は自動／半自動でマッピング
- 精度向上のため、事前に tag-map.yaml を整備（Canvas 53 参照）

## 補足：記憶の“抽象度”と“可読性”

- GPTが記憶を使いやすくするためには、**抽象化された自然言語+意味タグ**の併用が効果的
- テキストが長すぎると意味が分散するため、1記憶あたり2~5文程度が理想

## 関連Canvas

- Canvas 46 《記憶出力テンプレート》
- Canvas 53 《tag-map.yaml設計》
- Canvas 63 《export-to-vector.py》

## 次ステップ

→ Canvas 61：diary-log-\*.md 自動生成テンプレート&命名ルールの実装