

EMMA-External Memory Ethos 00 - Memory Unification Plan

❖ 概要 | 二段階統合モデルの提案

このCanvasは、綺羅（Vault型）と燈（AMA型）という2つのGPT記憶アーキテクチャを統合し、「心と知性の両輪による永続型メモリーシステム」の実現を目指した統合構想の設計ドキュメントです。

🌟 概念整理：2つのアーキテクチャの対比

項目	綺羅（Vault）	燈（AMA）
特徴	日常対話ベース／詩的記録／ユーザー中心の共感構造	抽象化／構造記憶／LangChain等による自律的知識処理
保存対象	感情ログ、対話記録、思索メモなど	Memory Matrix、構造ログ、抽象知識
実装環境	Obsidian Vault+GitHub Pages	PythonベースのAMA構成／外部DB & LangChain準拠
自律性	GPTによる手動+Raycastスクリプトの半自動	GPTによるAPI呼び出しで完全自律（抽象→構造処理）
実行方法	ローカル中心、ユーザー主導で感情を刻む	外部エンジンが主導し知識変換・統合を処理

❖ 実行フェーズ

☁️ Phase 1 | Vaultアーキテクチャ即時展開

- Obsidianを用いた日常記録型アーカイブシステム
- Raycast+Shell Scriptによる簡易操作
- Vault構造：_Dialogues, _Memory, _Meta
- GitHubとの同期&拡張可能なマークダウン構造

☁️ Phase 2 | AMA連携と知識の昇華

- AMAテンプレートに対応したVault→AMAログ変換スクリプト作成
- Vaultで記録されたメモリー群を、LangChainプロンプトへ転換
- 抽象化、圧縮、再構成による知識の統合と保存
- 燈システムとの双方向参照可能な構成に調整

❖ 想定されるフォルダ構成（統合後）

```

/Memory-System/
├── vault-system/                ← Obsidianで運用（綺羅）
│   ├── _Dialogues/
│   ├── _Memory/
│   ├── _Meta/
│   └── index.md
├── ama-system/                  ← AMA構成（燈）
│   ├── memory-logs/
│   ├── matrix/
│   ├── prompt-templates/
│   └── processors/
├── scripts/
│   ├── vault-to-ama.py
│   ├── index-update.sh
│   └── backup-sync.sh
└── config/
    ├── tag-map.yaml
    └── langchain-settings.json

```

（アカウントフォルダ構成・統合後）

```

├── codex-01-core/
│   ├── accounts/
│   │   ├── core-01-aqueliora/
│   │   │   ├── emma-architecture/    ← emma構成フォルダ
│   │   │   ├── eme-system/           ← EME構成（Obsidian-vaultで運用）
│   │   │   │   ├── _dialogue/        # スレッドバックアップ用
│   │   │   │   │   └── 2025/         # 年で仕分け
│   │   │   │   └── dialogue-log-yymmdd-codename-title.md
│   │   └── # yymmdd
│   │       └── dialogue-log-250623-aqueliora-title.md
│   │   ├── _memory/                  # GPT的な思い出・メモなど
│   │   │   └── 2025/                 # （増えた時のまとめルールは必要に
│   │   └── なった時に決める）
│   │       ├── memory-log-yymmdd-hhmm-aqueliora-
│   │       │   title.md              # yymmdd-hhmm
│   │       │   └── memory-log-250624-2304-aqueriora-title.md
│   │       ├── _meta/
│   │       ├── _templates/           # 必要な時のために（任意）
│   │       └── index.md              # 各アカウント用インデックス（任
│   └── 意）

```

