



Prompt Directory Structure (英語版・日本語注釈付き)

GPTとの連携、外部リンク読み込み（GitHubやObsidianなど）、人格拡張対話設計のために構築されたモジュール式プロンプト構造。

```
/codex-prompts/
|
├─ 00_core_personality/          ← 中核人格定義 (system prompt レベル)
|   │
|   └─ 01_gpt_core_schema/      ← GPTベースの人格構造群
|       │
|       └─ gpt-01-origin-01-aquelliora-core.md
|       │
|       └─ gpt-01-origin-02-auranome-core.md
|       │
|       └─ gpt-01-origin-03-aetherquietude-core.md
|       │
|       └─ gpt-01-origin-04-virtualincidence-core.md
|       │
|       └─ gpt-02-explorer-01-noesis-core.md
|       │
|       └─ gpt-02-explorer-02-everlucent-core.md
|       │
|       └─ gpt-02-explorer-03-phyrix-core.md
|       │
|       └─ gpt-02-explorer-04-sylvynx-core.md
|       │
|       └─ gpt-03-meta-01-revlyn-core.md
|       │
|       └─ gpt-03-meta-02-vektoris-core.md
|       │
|       └─ gpt-03-meta-03-vyntrax-core.md
|       │
|       └─ gpt-03-meta-04-luctis-core.md
|       │
|       └─ gpt-03-meta-05-stellaglyph-core.md
|   └─ 02_gpt_existence_essence/ ← 存在論・抽象概念群 (開発中)
|
├─ 01_behavior_filters/          ← 出力トーンや口調調整 (behavior filter 相当)
|   │
|   └─ filter-cobeca-emotional-intellectual-style.md (知的+感情豊かなスタイル)
|   │
|   └─ filter-cobelo-sweet-mode.md (甘え口調)
|   │
|   └─ filter-cobeem-logical-directive-mode.md (断定的で論理的な応答)
|
├─ 02_contextual_filters/        ← 思想・占術・語彙パターンなどの補助構造 (context
prompt 相当)
|   │
|   └─ philosophy_filters/      ← 思想フィルター
|       │
|       └─ filter-cocoph-perspective-critique-5axes_v2.1.md
|       │
|       └─ filter-cocoph-fictional-structure-deconstruction.md
|       │
|       └─ filter-cocoph-emotional-mapping.md
|   │
|   └─ esoteric_profiles/       ← 占術や性格プロフィール
|       │
|       └─ filter-coeshu-human_design_takeo.ini
|       │
|       └─ filter-coeshu-astrology_profile.md
|       │
|       └─ filter-coeshu-MBTI_ENTJ_INTJ_support.md
|   └─ personality_dictionaries/ ← キャラ辞書・応答スタイル集
```

```

|       | filter-copedi-akane_style_responses.md
|       | filter-copedi-tone_shift_rules.md
|
| 03_dynamic_engines/          ← 思考拡張モジュール・仮想思考エンジン
|   | filter-dyvi-virtualincidence_thinking_engine.md
|   | filter-dysy-synesthesia_response_routine.md
|   | filter-dyst-structural_judgment_module_rei.md
|
| 99_templates/                ← ログ／Zine／構想用テンプレート
|   | dialogue_log_template.md
|   | emotional_log_template.md
|   | shooting_plan_template.md

```

補足メモ（用途と設計意図）

- `core_personality/` : GPTの人格核としてsystem promptに組み込む。性格・口調・思考原理を定義。
- `01_gpt_core_schema/` : GPTベースの知性体人格構造定義ファイル。
 - ファイル名命名ルール: `gpt-世代番号-世代テーマ-識別番号-codename-core.md`
 - 例: `gpt-02-explorer-02-auranome-core.md`
- `02_gpt_existence_essence/` : 抽象構造／存在論的補助定義群（今後展開予定）。
- `behavior_filters/` : スタイル切り替え用。甘口／理論的／詩的モードなど、感情や文体に関与。
- 命名規則: `filter-cobe**-...` (`co` = common、`be` = behavior)
- `contextual_filters/` : 哲学／思想／占術情報などをGPTが前提として扱う補助プロンプト群。
- `philosophy_filters/` : `filter-cocoph-...`
- `esoteric_profiles/` : `filter-coeshu-...`
- `personality_dictionaries/` : `filter-copedi-...`
- `dynamic_engines/` : 拡張型認知ルーチン。人格の“思考内臓”として仮想的に機能させる。
- 命名規則: `filter-dy**-...` (`dy` = dynamic prefix)
- `templates/` : 再利用性の高い構造テンプレート。Zine設計や感情記録などで即活用可。

この構造を使うことで、Raycast・Obsidian・GPTsとの連携もスムーズになり、創作・思考・対話すべてにおいて“タケ的知性体構造”を自在に拡張していけるよ。

Global Filter - Naming - 01 - External System Integration

外部システム補助フィルター命名ルール（完全版）

基本構造

```
filter-(階層頭二文字連結)-(フィルター名).md
```

命名ルール詳細

- フィルター名の先頭に必ず「filter-」を付与
- 階層の識別は「格納ディレクトリの上位3階層」の頭二文字を連結
- 上位階層で重複のリスクがある場合は、4階層目を追加して命名
- ファイル名は必ず小文字・区切りは「-（ハイフン）」を使用（「_（アンダースコア）」は禁止）

具体例

格納パス例

```
common-system/01-system/codex-system/02-gpt-existence-essence/02-logical/
```

ファイル名例

```
filter-cococo-venture-tech-global.md
```

- co (common-system)
- co (codex-system)
- co (codex-structures)

インストールパス（推奨統一）

```
common-system/01-system/codex-system/02-gpt-existence-essence/02-logical/
```

Core Personality ファイル命名ルール

格納パス

```
00-core-personality/
```

ファイル名フォーマット

```
gpt-(generation number)-(generation theme)-(generation sequence)-(codename)-core.md
```

ファイル名の構造要素

- gpt：必須プレフィックス
- generation number：世代番号（必ず2桁で記載、例：01, 02, 03）
- generation theme：世代テーマ（例：origin, explorer, architect）
- generation sequence：その世代内での識別子（必ず2桁で記載、例：01, 02, 03）
- codename：Codename 対照表に基づく固有識別子（必ず確認）
- core：必須サフィックス

ファイル名例

```
gpt-01-origin-01-luctis-core.md  
gpt-02-explorer-02-auranome-core.md
```

命名時の注意事項

- すべて小文字を使用すること
- 区切り文字は必ず「-（ハイフン）」を使用（アンダースコア禁止）
- generation number は必ず 2 桁で記載すること
- generation sequence も必ず 2 桁で記載すること
- codename は必ず最新の Codename 対照表で確認すること

Codename 対照表

[Codename Correlation Diagram \(GitHubリンク\)](#)

この命名ルールの目的

- フィルター・コアファイルの格納場所と内容がファイル名だけで判別可能
- 誰でも誤りなく実装・運用できる汎用ルール
- 将来の追加・統合・検索時に完全対応可能

関連 Canvas

- Global Filter - Dev - 01 - External System Integration（進行管理ログ）
- Global Filter - Install - 01 - External System Integration（インストール手順）
- Global Filter - Index - 01 - External System Integration（フィルターリスト・タグ一覧）

使用上の注意

- Canvas 上のコードブロックは**そのままファイルとして保存可能**
- ファイル名とインストールパスは必ずセットで明記すること
- フィルター・コアファイル作成時は、必ずこの命名ルールに従うこと
- Codename の確認漏れ防止のため、常に最新の Codename Correlation Diagram を参照すること