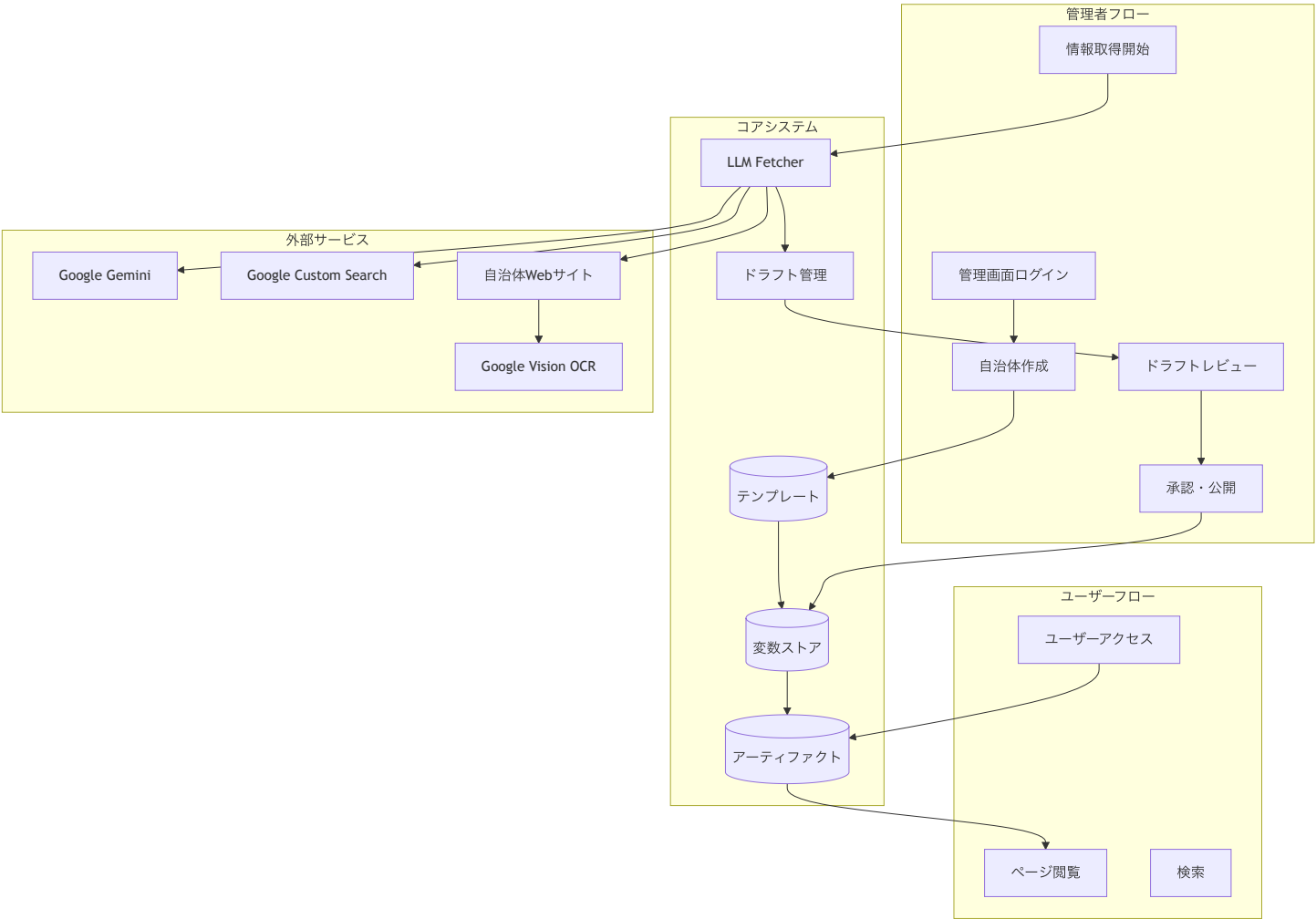


INNOMA 実装フローチャート

目次

1. メインデータフロー概要
2. ページレンダリングフロー
3. LLM情報取得フロー
4. ドラフト管理フロー
5. テンプレート・変数システム
6. 管理画面フロー
7. ストレージ抽象化層
8. コンポーネント階層
9. 認証・ミドルウェア
10. 未使用・参照用コード

1. メインデータフロー概要



2. ページレンダリングフロー

2.1 メインレンダリングパイプライン

エントリーポイント

app/[municipality]/[...path]/page.tsx
ArtifactPage()

アーティファクト読込

buildArtifactKey()
URLからキー生成

loadArtifact()
lib/artifact/index.ts

loadArtifactInternal()
lib/artifact/loader.ts

getFromCache()
lib/artifact/cache.ts

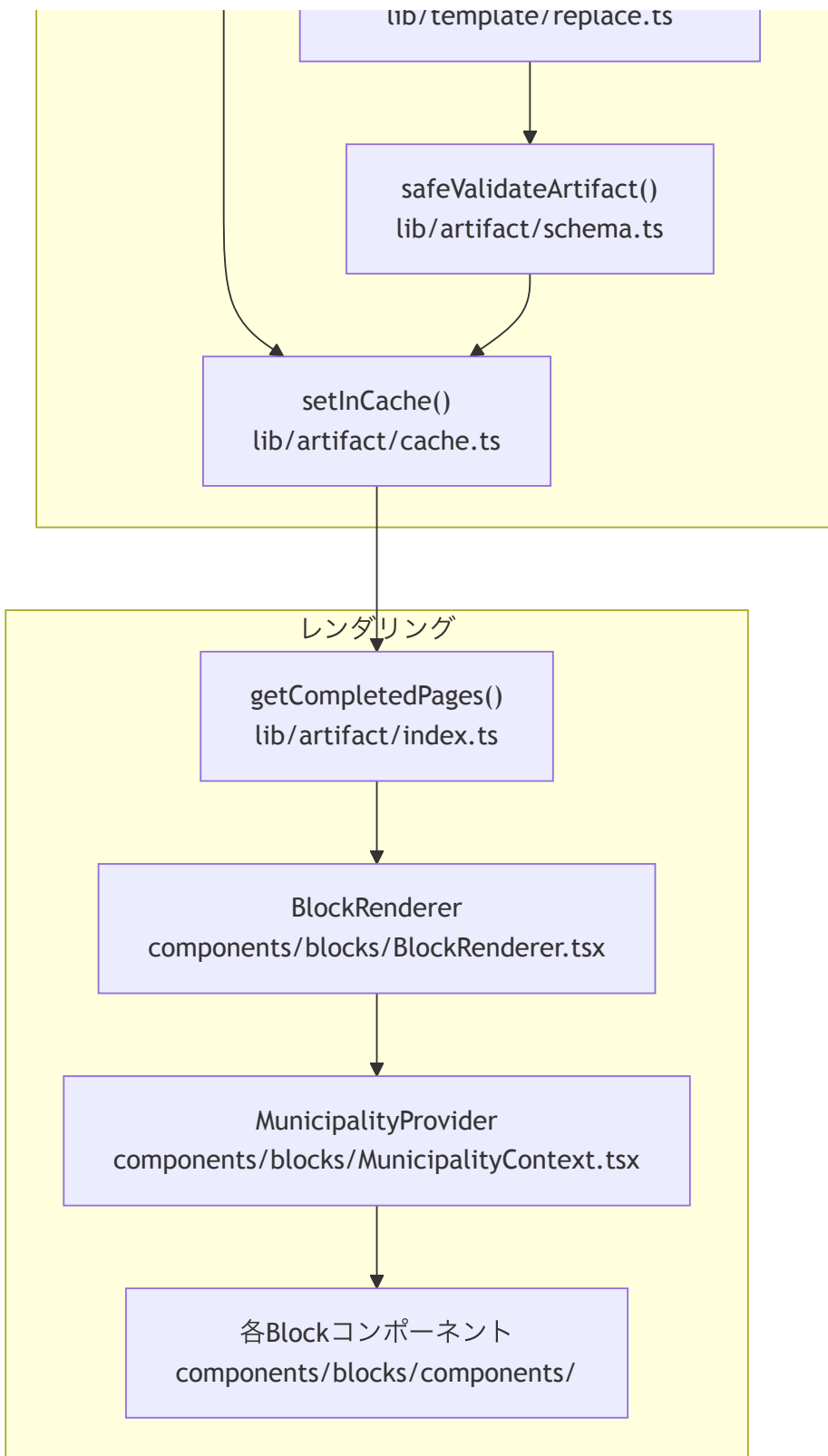
キャッシュミス

getDefaultStorageAdapter().get()
lib/storage/index.ts

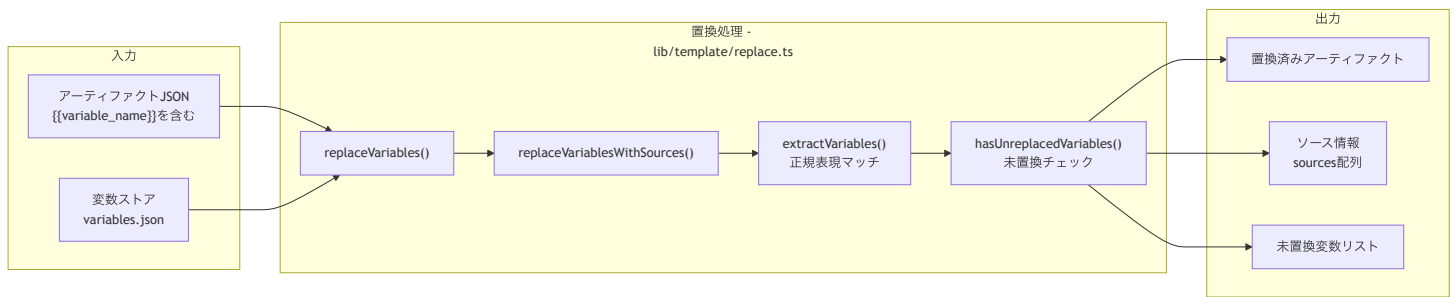
getVariableStore()
lib/template/storage.ts

キャッシュヒット

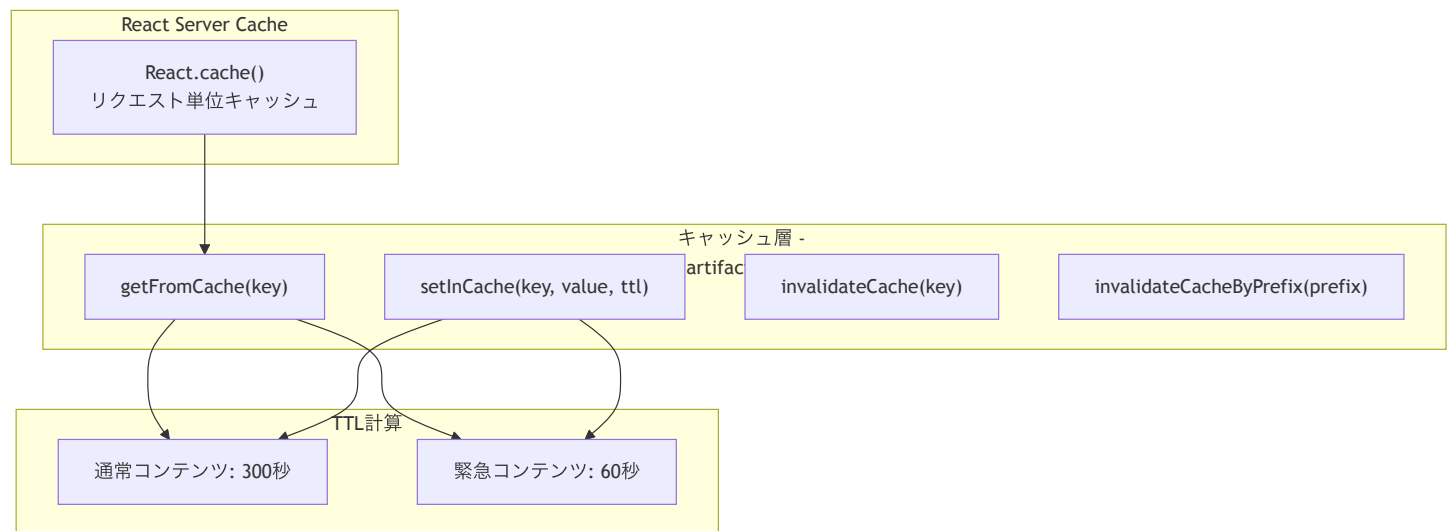
replaceVariablesWithSources()



2.2 変数置換処理

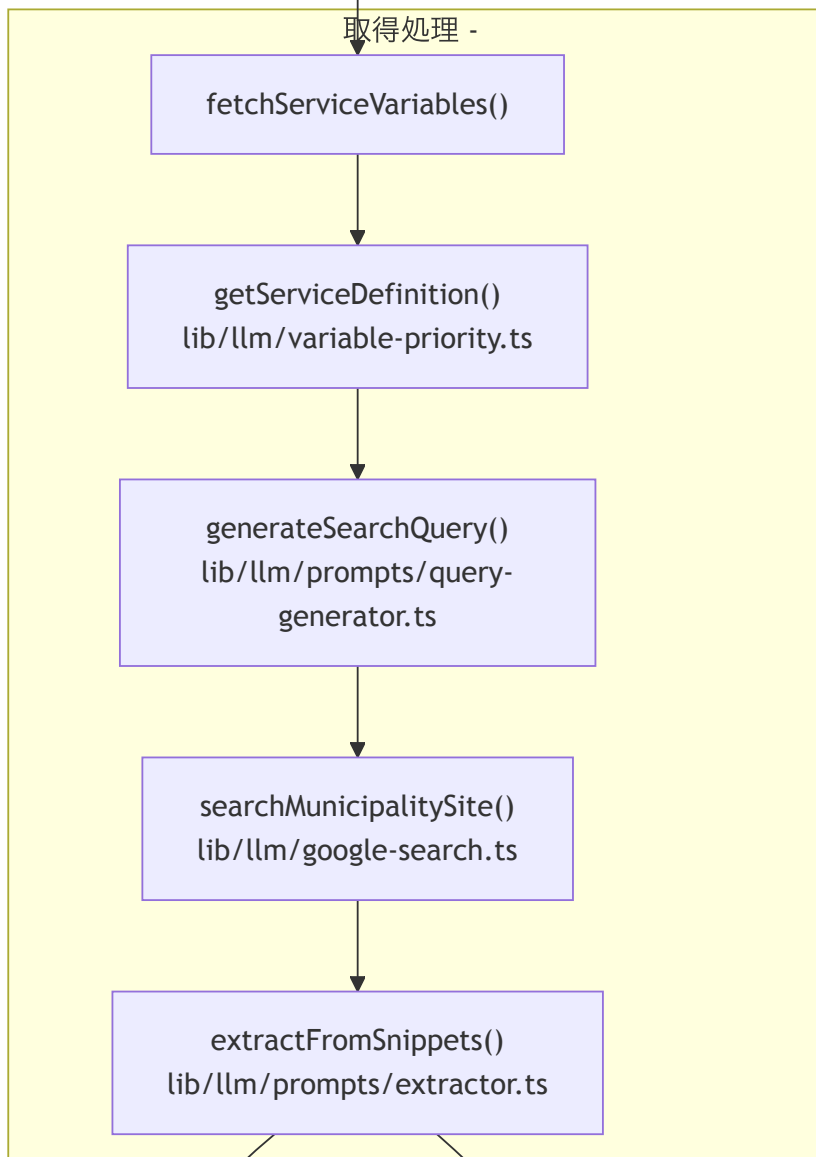
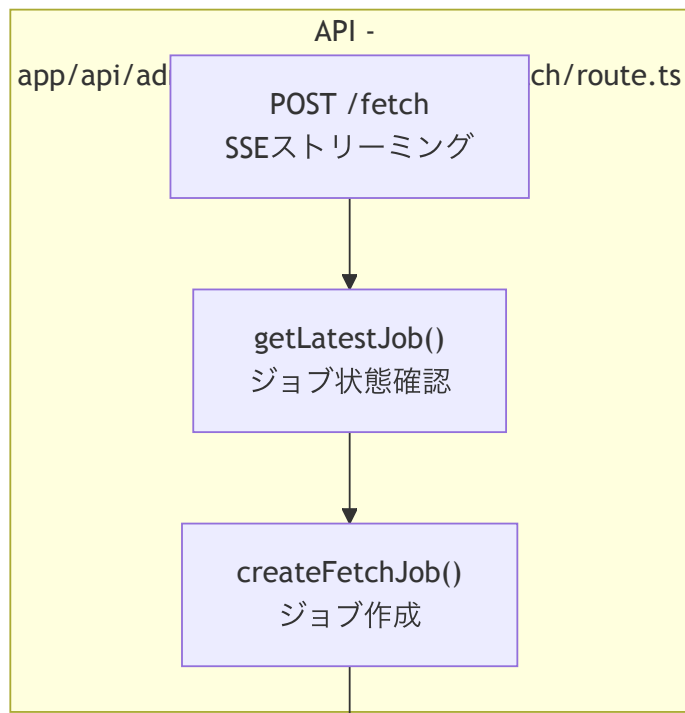


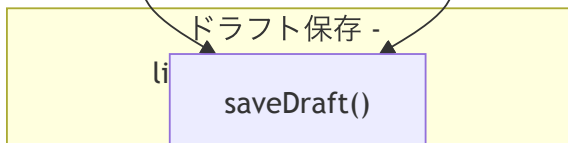
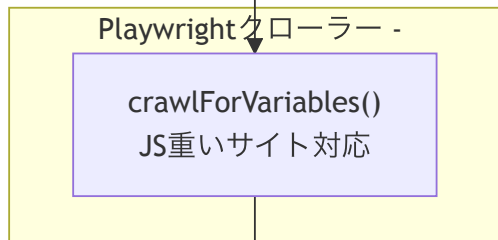
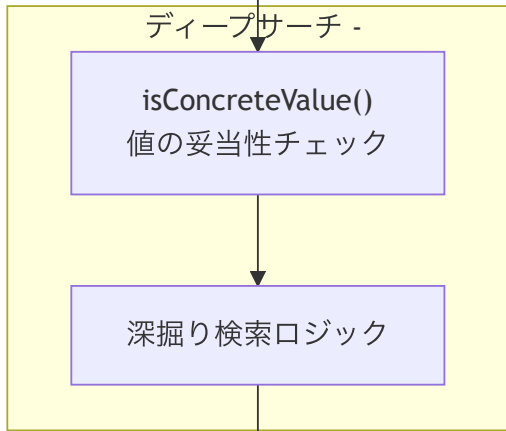
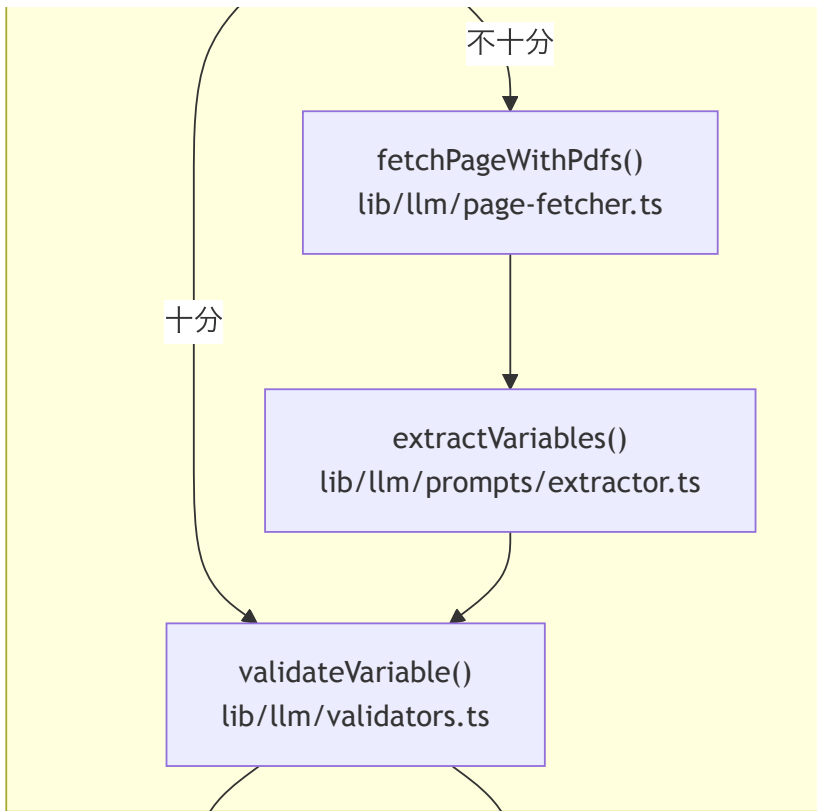
2.3 キャッシュ管理



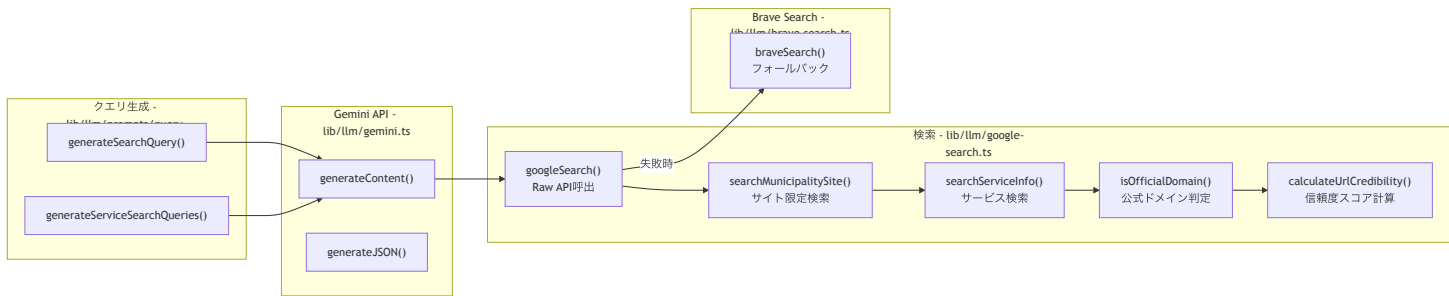
3. LLM情報取得フロー

3.1 メイン取得オーケストレーション

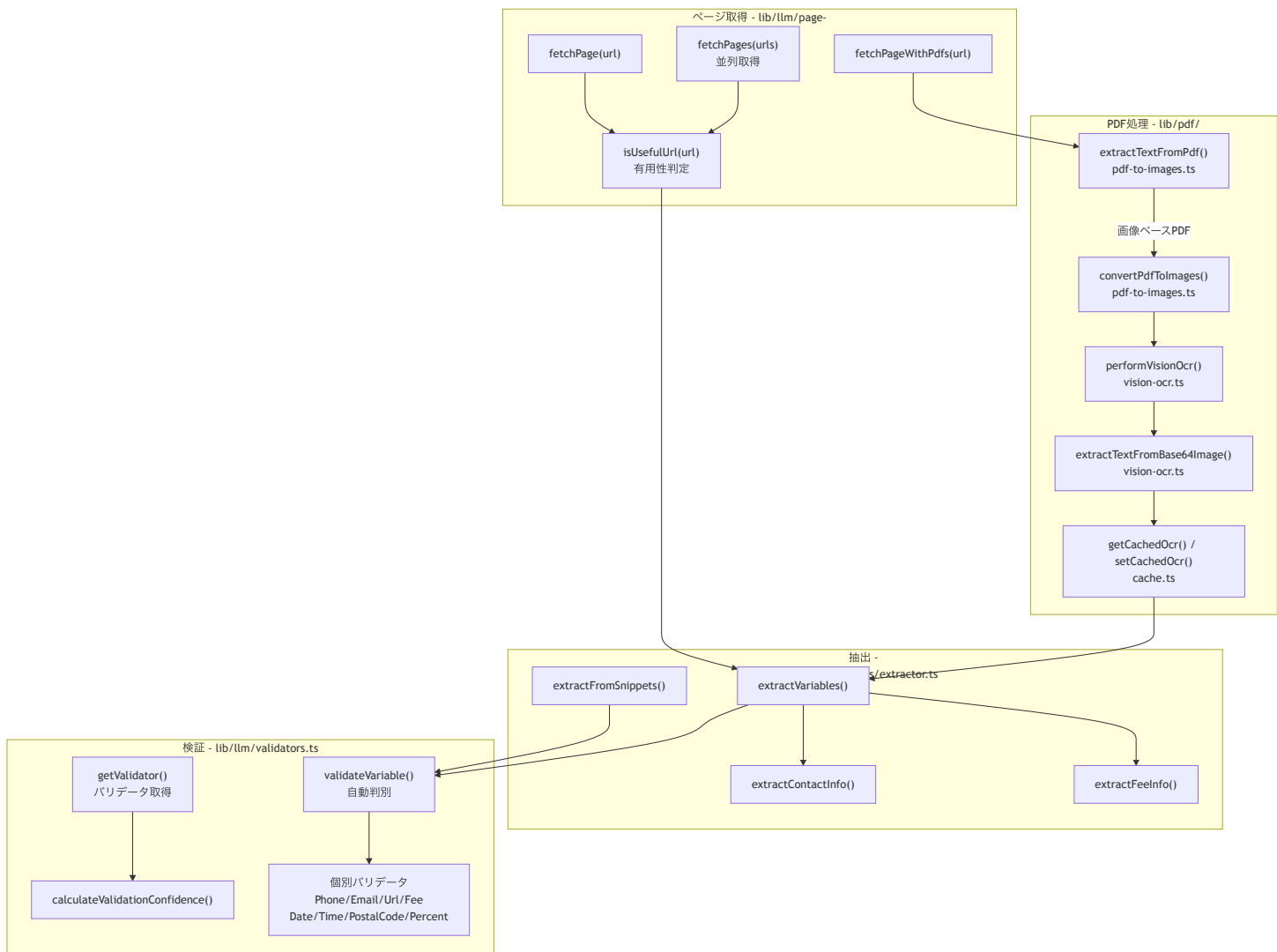




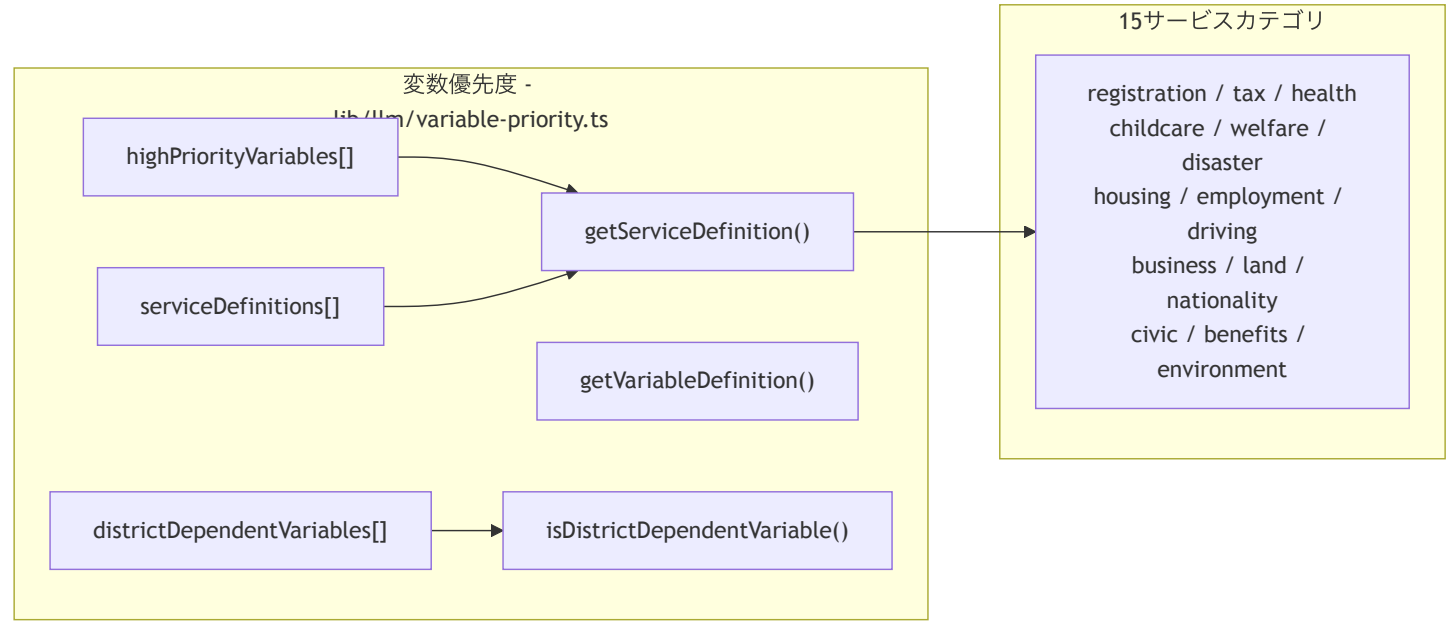
3.2 検索処理



3.3 コンテンツ取得・抽出

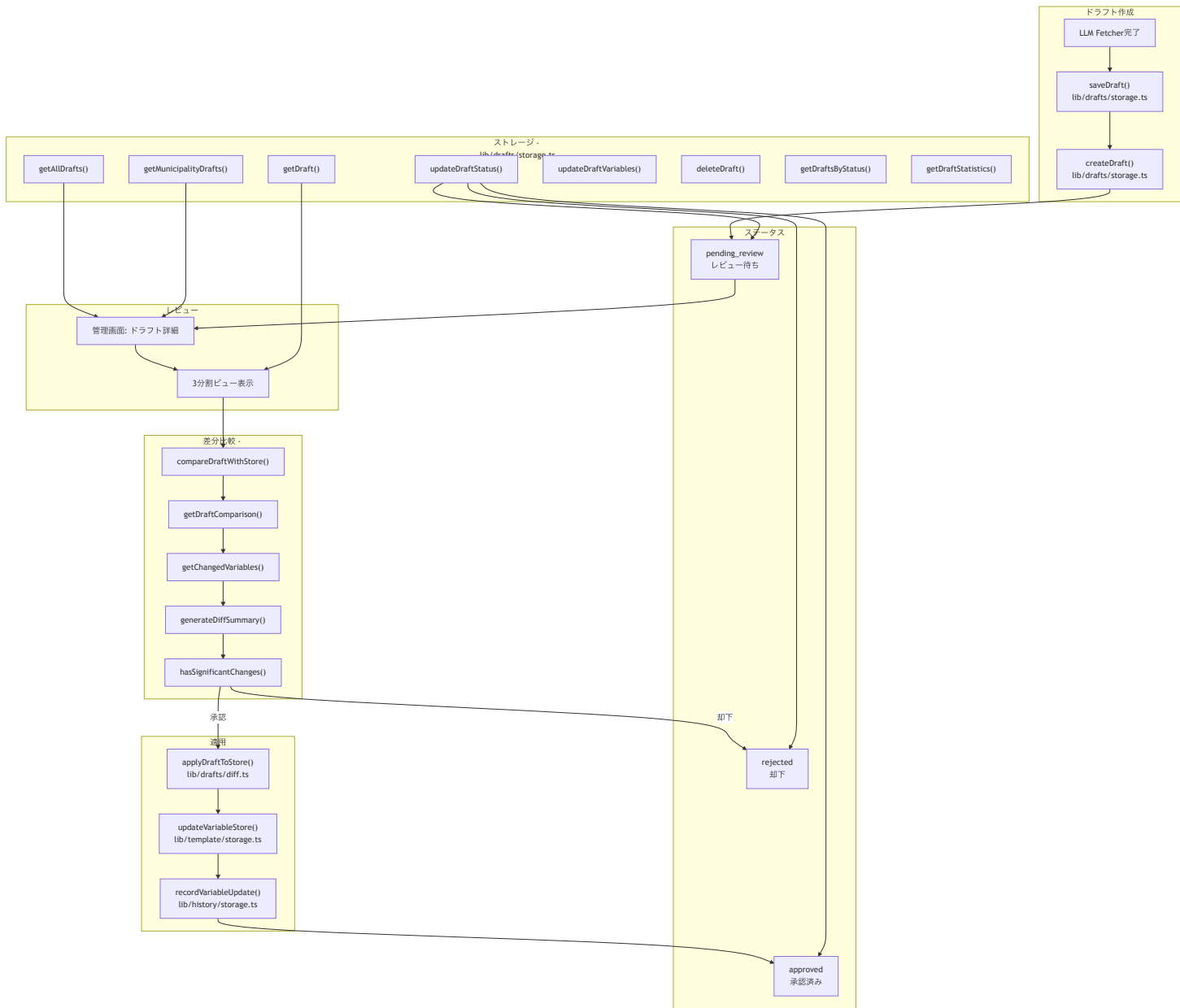


3.4 変数定義・優先度

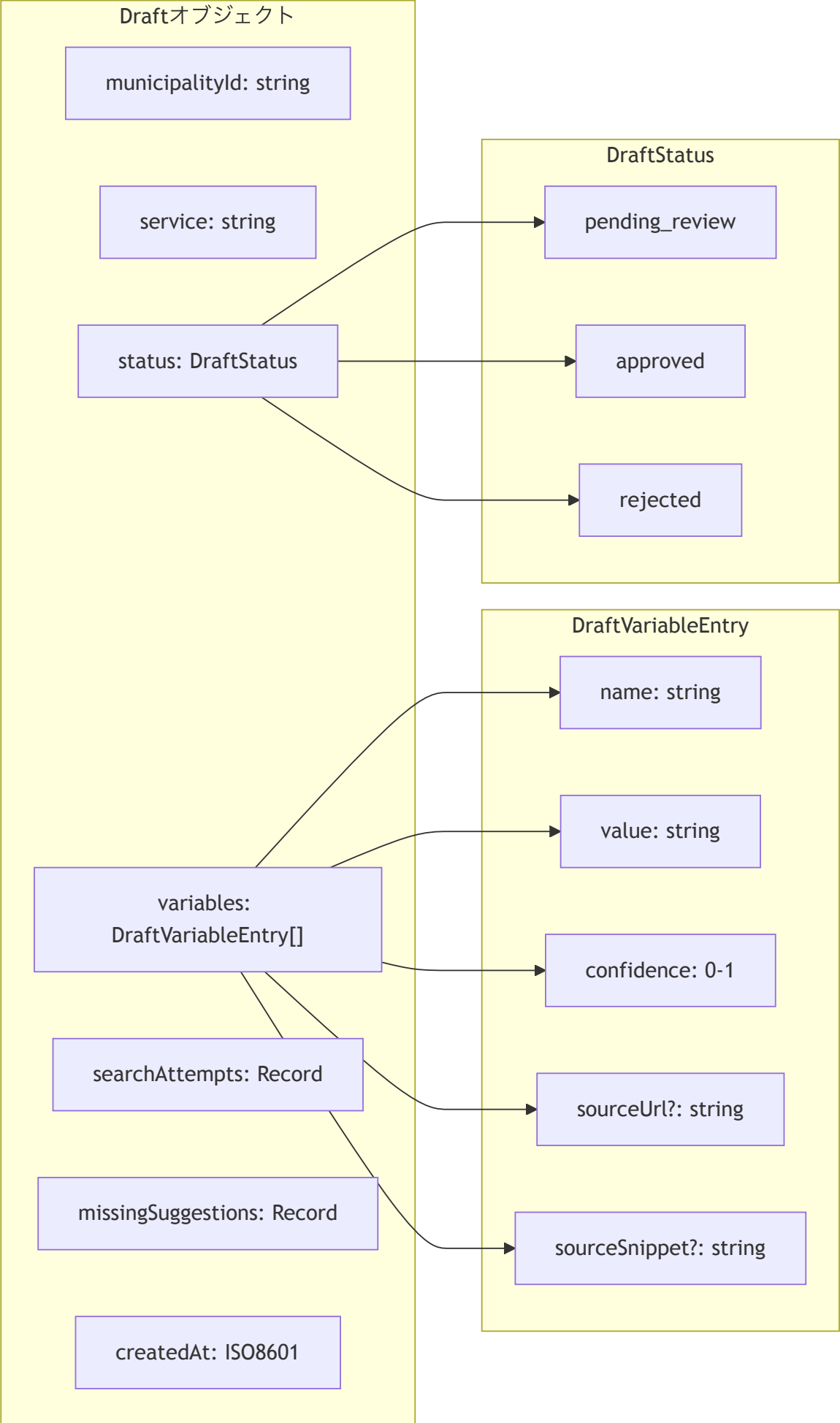


4. ドラフト管理フロー

4.1 ドラフトライフサイクル



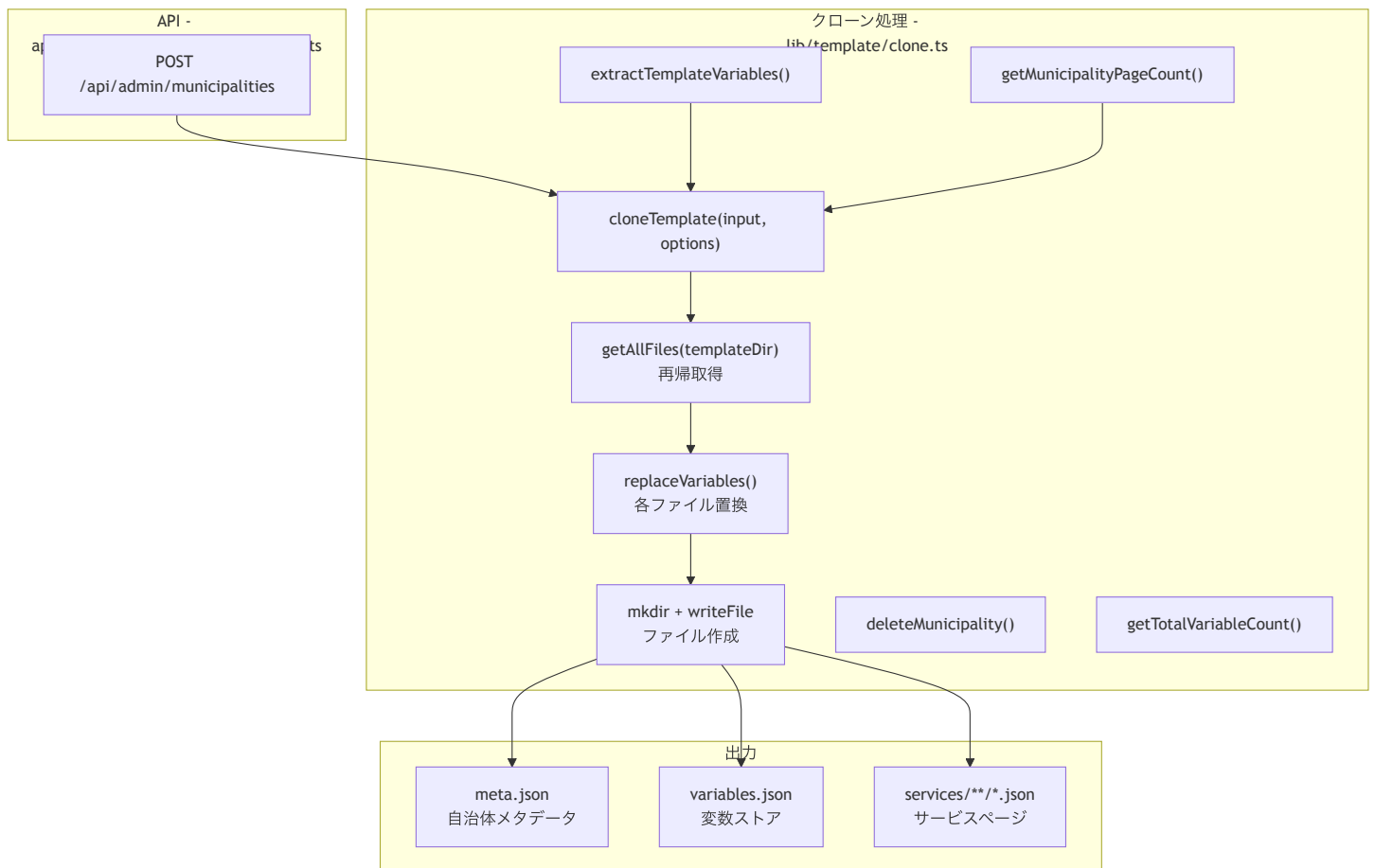
4.2 ドラフトデータ構造



approvedAt?: ISO8601

5. テンプレート・変数システム

5.1 テンプレートクローン処理



5.2 自治体データ管理

ストレージ操作 -

lib/template/storage.ts

getMunicipalities()
一覧取得

getMunicipality(id)
単体取得

getMunicipalityMeta(id)
メタ読み込み

updateMunicipalityMeta(id,
meta)
メタ更新

getVariableStore(id)
変数読み込み

updateVariableStore(id,
vars)
変数更新

updateMunicipalityStatus(id,
status)
ステータス変更

地区依存変数 -

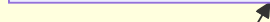
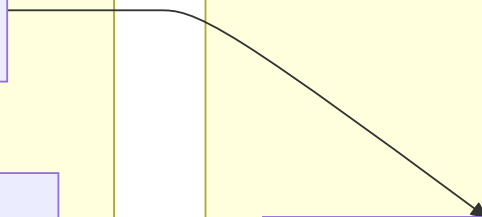
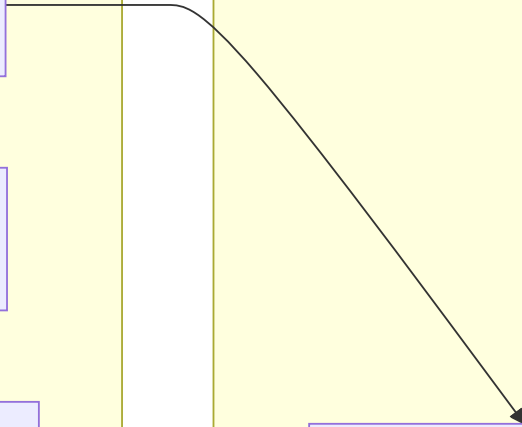
lib/template/district

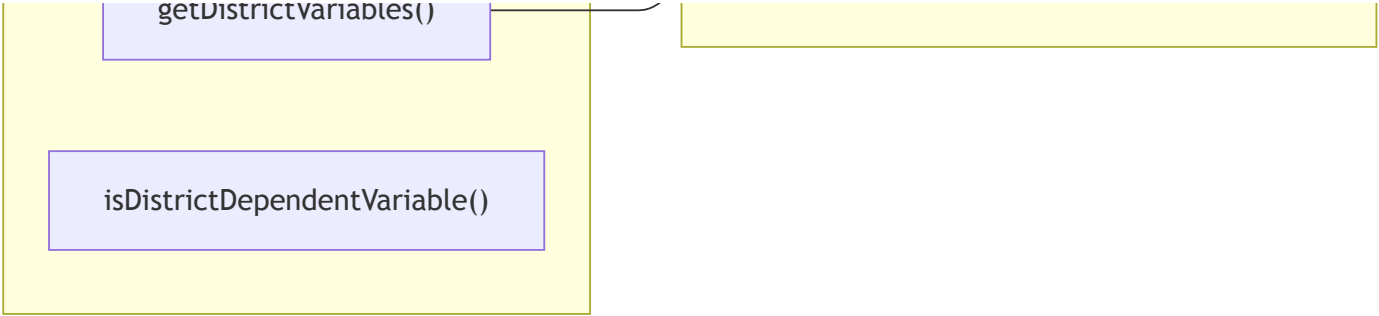
ファイル構造

data/artifacts/{id}/meta.json

data/artifacts/{id}/variables/*.json

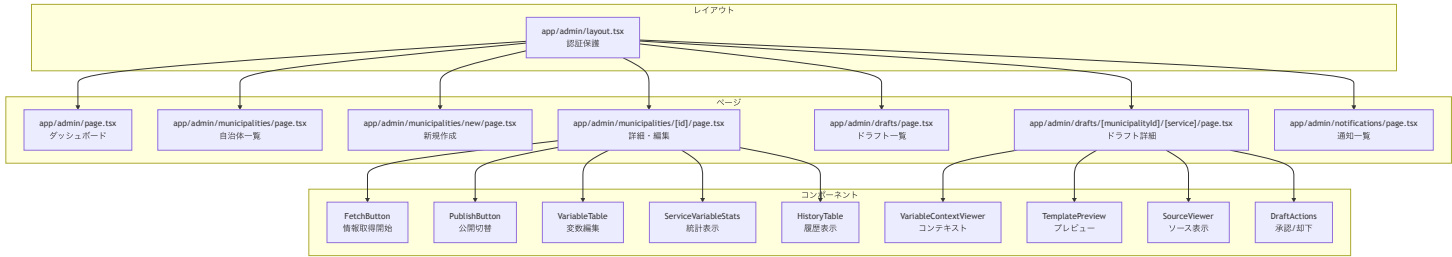
data/artifacts/{id}/services/**/*.json



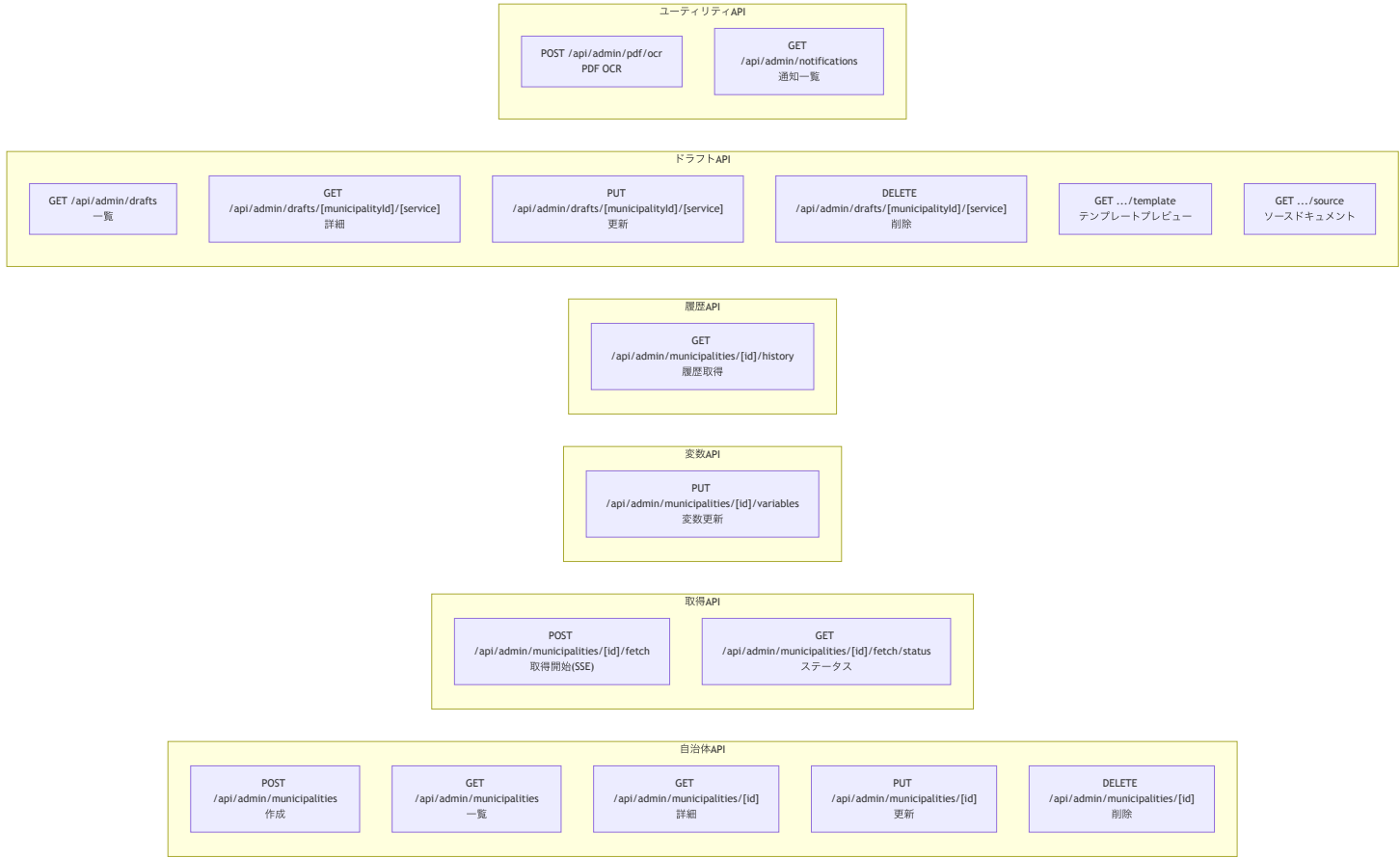


6. 管理画面フロー

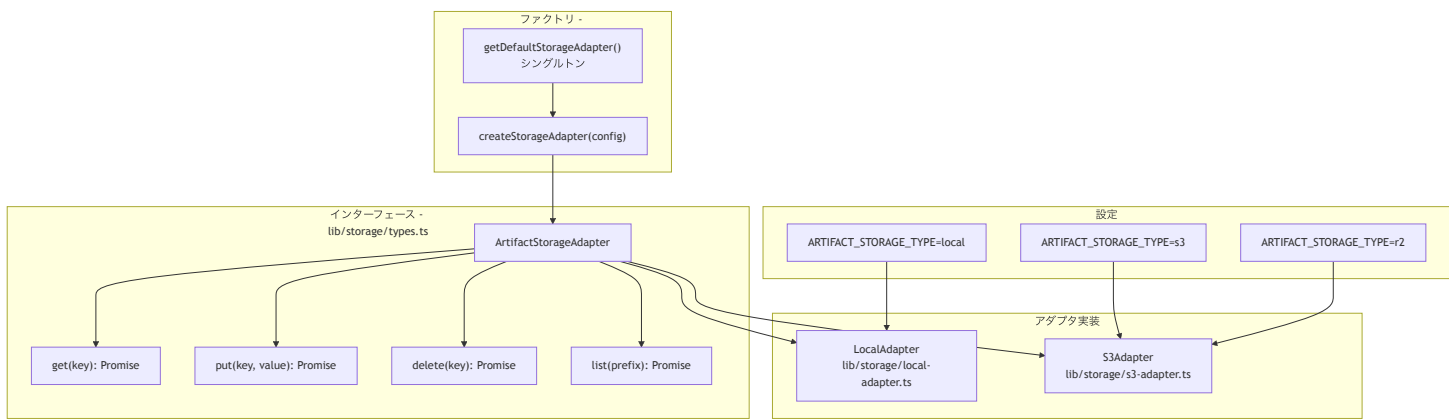
6.1 ページ構成



6.2 管理API

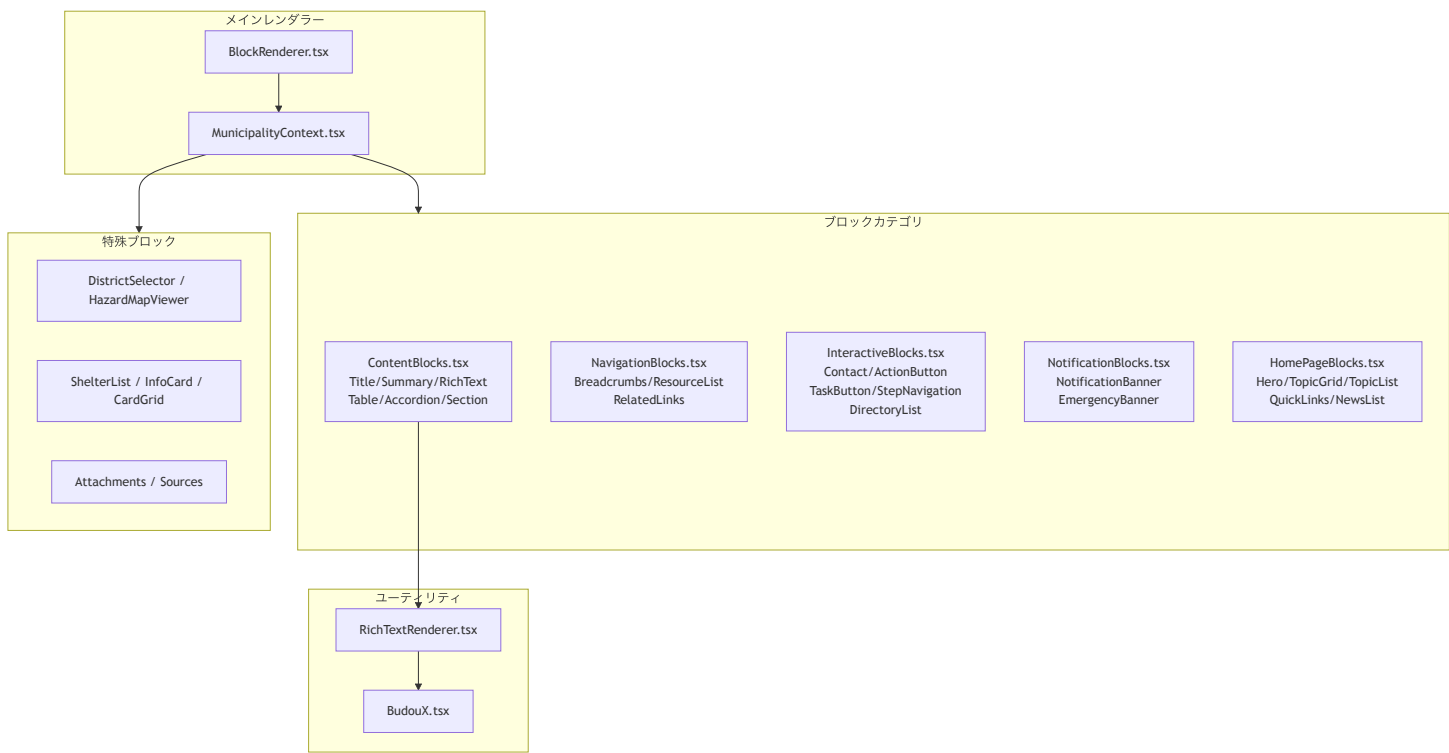


7. ストレージ抽象化層

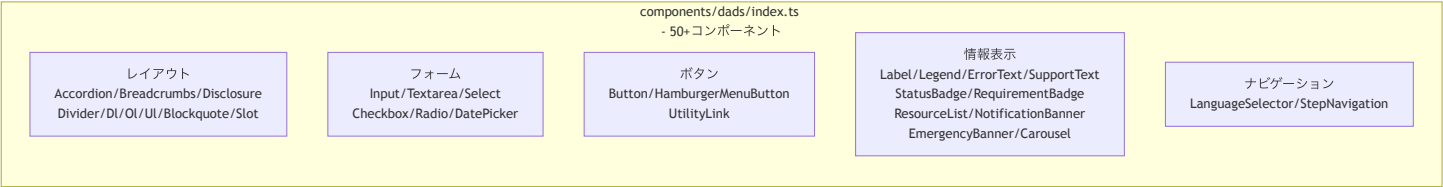


8. コンポーネント階層

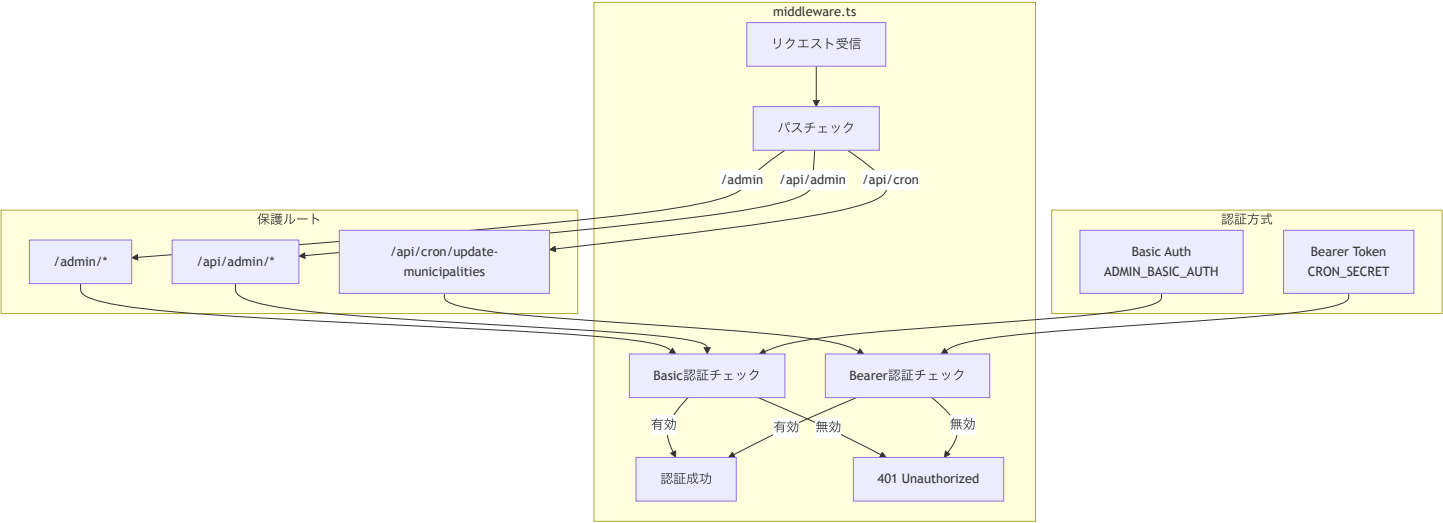
8.1 Blockレンダリング



8.2 DADSコンポーネント

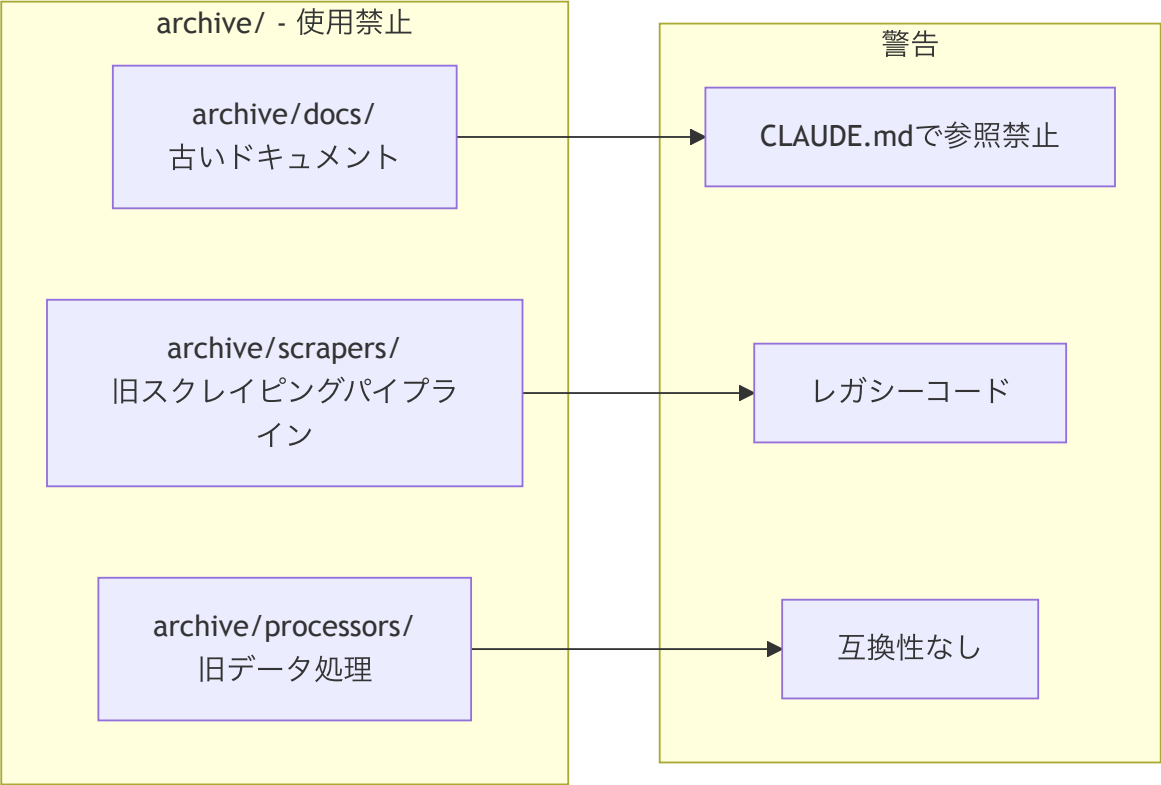


9. 認証・ミドルウェア



10. 未使用・参照用コード

10.1 アーカイブ（使用禁止）



10.2 開発・テスト用（本番未使用）

カテゴリ	パス	用途
開発ルート	<code>app/dev/dads/</code>	DADSコンポーネントショーケース
	<code>app/dev/review/</code>	コンテンツレビューUI
	<code>app/dev/structure/</code>	構造検査
スクリプト	<code>scripts/test-*.ts</code>	各種テストスクリプト
	<code>scripts/validate-artifacts.ts</code>	スキーマ検証
	<code>scripts/build-search-index.ts</code>	検索インデックス構築
参照用関数	<code>lib/llm/prompts/content-structurer.ts</code>	COMPONENT_SELECTION_RULES
	<code>lib/llm/prompts/missing-variable-suggester.ts</code>	不足変数提案

10.3 ジョブ管理（内部使用）

関数	役割
createFetchJob()	ジョブ作成
saveJob()	状態保存
getLatestJob()	最新取得
updateServiceStatus()	進捗更新
recordJobError()	エラー記録
completeJob()	完了処理

- **ストレージ:** data/artifacts/_jobs/{id}/latest.json
- **用途:** 長時間実行タスクの再開、進捗トラッキング、エラー復旧

10.4 履歴・通知（補助機能）

モジュール	関数	用途
lib/history/storage.ts	addHistoryEntry()	監査証跡
	recordVariableUpdate()	変更追跡
	recordBulkVariableUpdate()	変更追跡
	recordDraftApproval()	監査証跡
	recordDraftRejection()	監査証跡
	getHistoryList()	監査証跡
	getHistoryStats()	監査証跡
lib/notification/storage.ts	createNotification()	管理者通知
	getNotifications()	管理者通知
	deleteNotification()	管理者通知

付録: ファイル一覧と主要エクスポート

コアライブラリ

ファイルパス	主要エクスポート	役割
lib/artifact/index.ts	loadArtifact , loadArtifacts , getCompletedPages	アーティファクト読込インターフェース
lib/artifact/loader.ts	loadArtifactInternal	キャッシュ付き読込実装
lib/artifact/schema.ts	validateArtifact , safeValidateArtifact	Zodスキーマ検証
lib/artifact/cache.ts	getFromCache , setInCache , invalidateCache	インメモリキャッシュ
lib/template/clone.ts	cloneTemplate , deleteMunicipality	テンプレート複製
lib/template/replace.ts	replaceVariables , replaceVariablesWithSources	変数置換
lib/template/storage.ts	getMunicipalities , getVariableStore , updateVariableStore	自治体データCRUD
lib/llm/fetcher.ts	fetchServiceVariables	LLM取得オーケストレーション
lib/llm/gemini.ts	generateContent , generateJSON	Gemini APIクライアント
lib/llm/google-search.ts	googleSearch , searchMunicipalitySite	Google検索
lib/llm/page-fetcher.ts	fetchPage , fetchPageWithPdfs	コンテンツ取得
lib/llm/validators.ts	validateVariable , validatePhone , etc.	値検証
lib/drafts/storage.ts	getAllDrafts , getDraft , saveDraft	ドラフトCRUD
lib/drafts/diff.ts	compareDraftWithStore , applyDraftToStore	差分比較・適用

ファイルパス	主要エクスポート	役割
lib/storage/index.ts	createStorageAdapter , getDefaultStorageAdapter	ストレージファクトリ

コンポーネント

ファイルパス	主要エクスポート	役割
components/blocks/BlockRenderer.tsx	BlockRenderer	ブロック振り分け
components/blocks/MunicipalityContext.tsx	MunicipalityProvider , useMunicipality	コンテキスト
components/blocks/RichTextRenderer.tsx	RichTextRenderer	リッチテキスト解析
components/dads/index.ts	50+コンポーネント	DADSライブラリ

このドキュメントは2026-02-02に更新されました。