

検索を運用する

管理者向け・更新と設定

Plone × LanceDB 検索デモ

Web版の解説を、図と比較表を含めてまとめた持ち帰り資料です。実際の操作画面と最新の説明はサイトで確認できます。

Web版：管理者向け



<https://d3i4vv7cxswlt1.cloudfront.net/>

2026年10月14日版。開発中のデモのため、機能と画面は今後変わる場合があります。記載の操作は、各機能が管理者により有効化されていることを前提とします。

管理者向け

検索機能を有効にすることと、検索できるデータが揃うことは別です。コンテンツの抽出、文章のベクトル化、画像の推論は段階的に進みます。このページでは導入・設定の一覧に加え、状態をどう読み、どの処理が遅れているかを考えるための見方をまとめます。

対象はSite Administrator・Managerと導入担当者です。ここにある操作は管理作業の説明であり、このガイドを開くだけで設定やインデックスが変更されることはありません。

04 / TRANSACTIONAL OUTBOX

保存の確実さと、重い処理を切り離す

検索用データは、コンテンツの保存後にバックグラウンドで作ります。

Plone / ZODB : 同じトランザクション

コンテンツの変更
+
outbox に「更新してほしい」という記録
両方を一緒に確定。取り消すと両方消える。



worker が後で処理

outbox の記録を読む
最新の本文・権限を抽出
LanceDB に書き込む
失敗は再試行へ

同じ仕事が届いても、古い状態に戻さない

行に保存時の世代を持たせ、新しい世代の更新だけを適用する。
削除・権限の変更を優先し、本文や画像の処理を進める。

LanceDB は作り直せる検索用の写し。権限の最終判断は現在の Plone に問い合わせます。

Ploneへの保存とoutboxの記録を先に確定し、workerが後で検索用データを更新する流れ。

「保存済み」と「検索に反映済み」を分けて見る

保存するとまず更新の仕事がoutboxに積まれ、workerが文章を抽出してdocumentsへ書きます。その後に本文のベクトルや画像の特徴を作るので、キーワードで見つかるようになって、意味や画像の検索が追いついていないことがあります。待ち件数が減っているなら処理中、減らず失敗が増えているなら原因を調べる、という区別が重要です。

導入の要点

項目	内容
対応	Plone 6.0 (6.0.15 以上)・ 6.1・ 6.2、Python 3.11～3.13 (Plone 6.2 は 3.14 まで)、LanceDB 0.39.0
extra	なし (keyword 検索)・ [vector] (vector 検索と文章の自動タグ付け)・ [autotag-minilm] (文章の自動タグ付けで MiniLM を使うだけのとき)・ [vector] と同じ fastembed)・ [image] (画像検索)・ [heic] (HEIC)・ [zeo] (複数の instance)
profile	site ごとに、サイト設定 → アドオン で「c2.search.lancedb」を install する。profile を当てていない site では何もしない
最初のインデックス	install しても、今あるコンテンツは入らない。設定ページの操作で reindex_site を実行する (1 回 5,000 件まで、残りがあればもう一度)

instance を起動する環境で設定する環境変数:

変数	内容
C2_LANCE_WRITER_ROLE	active・ standby・ none (既定)。単体の instance では active。none の instance は Plone の検索で答える
C2_LANCE_URI	LanceDB の data directory (既定 /data/lancedb)
FASTEMBED_CACHE_PATH	vector の model (と、文章の自動タグ付けで選んだ MiniLM) の置き場所。既定は一時 directory なので、永続する場所を指定する
C2_IMAGE_MODEL_DIR	画像検索の model の置き場所

設定ページ

このサイトでは: 設定ページは /classic/@@lance-search-settings、状態の JSON は /classic/@@lance-search-status で開きます。

サイト設定 → アドオンの設定 → LanceDB 検索 (@@lance-search-settings) で、状態の確認・ 設定・ 操作を行います。開けるのは Manager と Site Administrator です。LanceDB を持たない instance では件数を読めず、操作も出ないので、writer か standby の instance で開きます。

状態: この instance・ インデックス・ vector 検索・ 画像検索・ 「もしかして」・ 自動タグ付けの状態と、警告。
設定: 機能・ 警告を出す条件・ 自動タグ付け (共通・ 画像・ 文章)・ 画像検索の除外・ 環境変数 (表示だけ)。操作: 保守の操作。確認が要る操作と、戻せない操作があります (下の「保守の操作」)。

「機能」の項目 (既定):

項目	key	既定
LanceDB を使う	enabled	on
Plone の検索画面を LanceDB で検索する	search_view.enabled	off
vector 検索を使う	vector.enabled	off
ICU で順位を補正する	icu.enabled	off
「もしかして」を出す	suggestions.enabled	off
検索結果に「本文からの参照」を出す	incoming_links.enabled	on
TinyMCE の画像・リンクの挿入で LanceDB を使う	tinymce_search.enabled	off
自動タグ付けを使う	autotag.enabled	off

各項目の下に補足に、変えると何が起こるかという既定が出ます。数の欄は、空にして保存しても既定に戻りません(空欄は「変えない」)。「画像検索の除外」は、保存に確認が要ります。除外した画像は、画像インデックスから消えます。画像検索の on/off は設定ではなく、操作の image_enable・image_disable で行います。同じことを JSON API (@@lance-search-status・@@lance-search-controlpanel・@@lance-search-maintenance)でも行えます。送るときは CSRF token が要ります。

有効化した後、裏で何が起こるか

ベクトル検索をonにすると、既存文書の特徴計算が始まります。画像検索はモデルの準備と画像の推論が必要です。スイッチの状態だけでは完了を判断せず、インデックス・worker・待機中の仕事・失敗の状態と一緒に確認します。

機能	起こること	要るもの
vector 検索	worker が全コンテンツの embedding を作り始める。作り終わるまで、まだ embedding の無い文書は意味の近さでは出ない。画面の検索はハイブリッドになり、検索の方法の切り替えが出る	[vector] extra. model (intfloat/multilingual-e5-small) は最初に使うときに取得する
画像検索 (image_enable)	全画像を処理待ちに登録し、worker が推論する。検索画面に「画像も含めて探す」が出る	[image] extra、model の用意 (約 1.5 GB、CLI の c2-lance-image-model)、writer と standby に 6 GiB の memory。HEIC も扱うなら [heic]
自動タグ付け	保存・更新したコンテンツから判定する。今あるコンテンツは操作の autotag_review で判定する	画像の判定は画像のインデックス、文章の判定は [vector] extra (vector 検索を on にしなくてよい)
「もしかして」	worker が語彙を作る。以後 24 時間ごとに作り直す	-

機能	起こること	要るもの
ICU	字の並びだけの一致を下に回す	切り替えた後に rebuild

文章の自動タグ付けでMiniLMを選ぶと、初めて使うときに約0.22 GBのモデルを取得します。検索用のE5とタグ用のMiniLMを併用する構成では、モデルがそれぞれメモリを使う点にも注意します。

初回だけ遅いのか、毎回遅いのか

画像検索のtext modelは最初の検索時に読み込まれ、そのプロセスで再利用されます。初回の応答時間には読み込みが含まれるため、暖まった後の検索時間と分けて測ります。再起動後は同じことが起きるので、イベント前の動作確認では初回と2回目以降の両方を見ると状況を説明しやすくなります。

画像推論の子プロセスと、検索語のモデルを持つ親プロセスは別にメモリを使います。導入資料の6 GiBという画像検索構成の目安は、あらゆるデータ量での保証ではありません。Ploneのキャッシュや同時要求も含め、実際のプロセスの使用量を確認します。

似た画像を探す画面

このサイトでは: /classic/@@lance-search-similar-images で開きます。

サイト設定 → アドオンの設定 → 似た画像を探す (@@lance-search-similar-images) で、同じ画像や似た画像を探し、それを使っているコンテンツを Plone の編集・削除の画面で直せます。重複した画像の整理に使います。

探し方は、1枚の画像から、サイト全体の一覧、アップロードして比較、の3つです。比較のためにアップロードした画像は保存しません。結果の件数は12・24・48・96から選べます。

高い類似度は特徴が近いことを示します。表示値が1.000でも、丸めやモデルの表現があるため、元ファイルの完全一致を証明する値としては扱わず、原画像も確認します。機能は画像検索が有効でインデックスがある、writerまたはstandbyのインスタンスで使います。

保守の操作

設定ページの「操作」は、インデックス・ベクトル・再構築・画像・候補語・タグ付けなどに分かれます。多くはworkerが進める依頼を登録しますが、documentsのrebuild開始など要求の中でテーブルに触れる処理もあります。実行後は「状態」で進み具合と失敗を確かめます。

再抽出と再構築は、直す対象が違う

操作	主に何をし直すか	考え方
reindex_site	ZCatalogから対象を列举し、Ploneから内容を再抽出	元コンテンツを検索用データへ取り込む
rebuild	現在のdocumentsを新世代へ写し、索引設定を反映	tokenizerなど検索側の構造を作り直す
image_rebuild	画像の新世代を作り、画像の同期・推論を進める	画像には独自の検査・公開手順がある

documentsのrebuildは、Ploneの本文をすべて読み直す操作ではありません。現行テーブルから新しい世代へ写し、行数や世代の整合性を確認してmanifestの参照先を切り替えます。構築中は現在のテーブルで検索が続けられます。chunksはこのdocuments再構築の対象外なので、文章のベクトルの問題を同じ操作ですべて直せるとは限りません。

画像には独立したbuild・検査・公開の手順があります。画像のrollbackも、単に参照名を戻すだけではなく、保持していたテーブルをbuildとして再検査します。「世代を戻せる」と「何も確認せず瞬時に元へ戻す」ことを分けて考える必要があります。

確認が要る操作は、確認のチェックを入れてから送ります。チェックの文は、その操作で失われるものです。戻せない操作は、「操作」の最後の閉じた「戻せない操作」の中にあります。次の3つはManagerだけが使えます: drop_unused_tables (使っていない古い table を消す)、image_drop_index (画像のインデックスを消す)、image_convert_heic (保存済みの HEIC を JPEG に変換し、元のファイルを残さない)。rebuild は今の設定でインデックスを作り直し、写し終えたら切り替えます。写している間も、検索は今の table で答えます。

困ったとき：どの段階で止まったかを調べる

①検索要求がどのモード・範囲へ届いたか、②LanceDBで応答したか、③索引が追いついているか、④現在の利用者が閲覧できるか、の順に切り分けると、間違った再構築を避けられます。

HTTP 200でも、文章検索は標準のZCatalogへ戻っている場合があります。API応答のlance.fallback.reasonを確認すると、「0件だからデータがない」のか「別の検索方式で答えた」のかを区別できます。画像の検索機能を利用できない場合は503となり、単純な0件とは異なります。

権限変更の直後は、最終認可で漏れを防ぎつつ、検索側への反映が遅れて結果が少なくなることがあります。公開状況や閲覧権限を確かめたうえで、必要な同期の状態を調べます。以下の表は原因が分かった後の操作の対応表です。

こと	見る所・すること
検索が Plone の検索で答えている	@@lance-search の応答の lance.fallback.reason。理由は応答の lance.fallback.reason に出る
状態の worker が動いていない	C2_LANCE_WRITER_ROLE が active か standby かを確かめる
状態に警告が出ている	警告の文に従う。警告の条件は設定の「警告を出す条件」で変えられる
権限を変えたのに検索結果に出る / 出ない	操作の resync_security
失敗で止まった job がある	操作の requeue_dead
embedding の作成に失敗した文書がある	操作の retry_embeddings
画像の推論が止まった / 失敗した画像がある	操作の image_restart_inference / image_retry_failed。失敗の一覧は @@lance-search-status?images_failures=1
画像検索を有効にできない (child_stopped・child_not_ready)	設定ページの「状態」の警告の案内に従う (image_restart_inference を押す、または少し待って押し直す)
状態に rebuild_needed が出ている	操作の rebuild で、新しい設定の table に切り替える

こと	見る所・すること
AND・OR・"..."で絞ったのに、満たさない結果が出る	「ハイブリッド」と「ベクトルのみ」では、AND・OR・"..."は語の一致で探す側だけに効く（-で除く語はどの方法でも効く）。書いたとおりに絞るには「キーワードのみ」を案内する

版を上げるとき

版を上げたら、サイト設定 → アドオン の「更新が要るアドオン」の欄で c2.search.lancedb の button を押し、upgrade step を実行します。

reinstall や upgrade step の再実行では、ZMI で外した Site Administrator の権限が戻ります (profile の rolemap が上書きするため)。そのときは外し直します。

止める・外す

「LanceDBを使う」をoffにすると、文章検索はPloneの検索へ戻り、更新の仕事は処理待ちに積まれます。再開すれば追いつく設計ですが、停止中に増えた件数に応じて時間が必要です。画像検索には同じ意味の代替検索がないため、文章と同じ復帰先があるとは限りません。

アドオンをuninstallしてもLanceDBのデータは消しません。機能を止めること、再開できるようデータを残すこと、不要データを削除することは別の操作です。運用では戻し方を先に決め、現在検索が参照している世代を確認してから保守します。

設計の理由まで知りたいとき

技術者向け：非同期処理・権限・世代の仕組み

実装を読む：管理者向けの手引き

実装を読む：再構築・縮退・writerの制約

持ち帰って試す

更新反映の観察演習

ページの本文を変更し、保存した時刻と検索結果に新しい本文が現れた時刻を記録してみましょう。保存の完了と検索への反映は別の段階です。公開状態や権限の変更を伴う場合も、閲覧できる利用者から見た結果を確認してください。

記録すること

入力した言葉 / 対象コンテンツ / 検索方式 / 結果の違い / 次に確かめたいこと

この演習の見出しはPDF本文に含まれています。サイトの検索で見出しを入力すると、PDF本文が検索対象になっていることも確かめられます。