

# 日々是Oracle APEX

Oracle APEXを使った作業をしていて、気の付いたところを忘れないようにメモをとります。

2021年2月16日 火曜日

## ブロックチェーン表にPDFを保存し全文検索を行う

題材の相談を受けたので、ちょっとアプリを作ってみました。作り方は[以前に記事に書いて](#)はいるので、あまり細かい実装はせず、ブロックチェーン表で可能かどうかの確認を目的とします。

結論としてはAlways Freeの21cでは、Oracle Textのauto filterに何か問題があるようで、うまく動きませんでした。ブロックチェーン表だからできない、ということではなく、ブロックチェーン表でも全文検索索引は機能するはずですが、現状ではAlways Free以外に利用可能な21cのインスタンスは無いので確認できません。

以下より作業ログです。

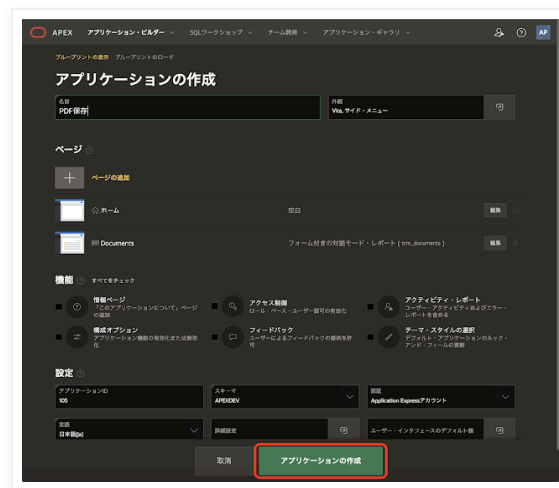
最初にPDFを保存するブロックチェーン表を定義します。構造はクイックSQLの以下のモデルを使います。

```
# prefix: trm
# semantics: default
documents
  title vc400
  document file
```

DDLを生成し、スクリプトを保存してから、**レビューと実行**を行うと、DDLの編集画面になります。DDLをブロックチェーン表となるように書き換えます。

```
-- create tables
create blockchain table trm_documents (
  id                number generated by default on null as identity
                  constraint trm_documents_id_pk primary key,
  title             varchar2(400),
  document          blob,
  document_filename varchar2(512),
  document_mimetype varchar2(512),
  document_charset  varchar2(512),
  document_lastupd  date
)
no drop until 31 days idle no delete locked HASHING USING "SHA2_512" version "v1"
;
```

DDLを実行し表を作成した後、**アプリケーションの作成**を行い、**アプリケーション作成ウィザード**を起動します。作成するアプリケーションの**名前**を指定し（ここでは**PDF保存**）、**アプリケーションの作成**をクリックします。



これで、PDF(に限らず、あらゆるファイル)のアップロードと保存、ダウンロードを行うアプリケーションが作成されました。

次に全文検索索引をブロックチェーン表に付加します。

最初に日本語レクサーを登録します。

```
begin
  ctx_ddl.create_preference('ja_lexer', 'JAPANESE_LEXER');
end;
/
```

Autonomous Databaseの場合は、ワークスペース・スキーマにCTX\_DDLパッケージの実行権限が割り当てられていません。SQL Developer WebなどからADMINでサインインして実行権限を割り与えます。

```
grant execute on ctx_ddl to スキーマ名;
```

BLOB(バイナリ・データ)として登録されたファイルより、文字情報を抽出するフィルタを設定します。

```
begin
  ctx_ddl.create_preference('auto_filter', 'AUTO_FILTER');
  ctx_ddl.create_policy('auto_policy', 'auto_filter');
end
/
```

ユーザー・データストアに利用されるファイルよりテキストを抽出するプロシージャを作成します。

```
create or replace procedure pdf_to_text(
  rid in rowid,
  tlob in out nocopy clob
)
is
  l_title trm_documents.title%type;
  l_document trm_documents.document%type;
  l_clob clob;
begin
  for c in (select title, document from trm_documents where rowid = rid)
  loop
    if c.title is not null then
      dbms_lob.writeappend(tlob, length(c.title), c.title);
    end if;
  end loop;
```

```

if dbms_lob.getlength(c.document) > 0 then
  begin
    ctx_doc.policy_filter('auto_policy', c.document, l_clob, true);
    dbms_lob.copy(tlob, l_clob, length(l_clob), length(tlob)+1, 1);
  exception
  when others then
    null;
  end;
end if;
end loop;
end pdf_to_text;

```

ユーザー・データストアtrm\_documents\_storeを登録します。APEXDEVとなっている部分は、作成したプロシージャpdf\_to\_textのオーナーに置き換えます。

```

begin
  ctx_ddl.create_preference('trm_documents_store', 'user_datastore');
  ctx_ddl.set_attribute('trm_documents_store', 'procedure', 'APEXDEV.pdf_to_text');
  ctx_ddl.set_attribute('trm_documents_store', 'output_type', 'CLOB');
end;

```

作成したユーザー・データストアを使って、全文検索索引を作成します。

```

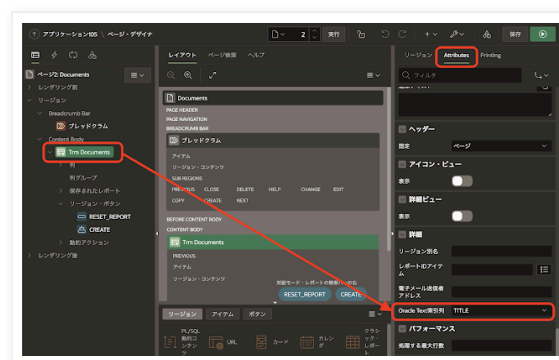
create index trm_documents_sidx on trm_documents(title)
inctype is ctxsys.context
parameters('filter ctxsys.null_filter lexer ja_lexer datastore trm_documents_store sync(on commit)');

```

以上で、全文検索索引の作成は完了です。

アプリケーションにすでに含まれている対話モード・レポートが全文検索索引を使用するように属性の設定を行います。

対話モード・レポートのリージョンを開き、右ペインでAttributesを開きます。詳細にOracle Text索引列の設定があるので、それに列TITLEが設定されていることを確認します。



以上で、一応動くはずなんですが、試すと検索にヒットしません。確認のため以下のコードを実行しました。

```

declare
  l_clob clob;
begin
  for c in (select * from trm_documents)
  loop
    ctx_doc.policy_filter('auto_policy', c.document, l_clob, true);
    dbms_output.put_line(l_clob);
  end loop;
end;

```

以下のエラーが発生します。

ORA-20000: Oracle Textエラー:

DRG-11207: ユーザー・フィルタ・コマンドが状態1で終了しました。

DRG-11221: サードパーティのフィルタが、このドキュメントの破損を示しています。

ORA-06512: "CTXSYS.DRUE", 行186

ORA-06512: "CTXSYS.CTX\_DOC", 行1682

ORA-06512: 行6

ORA-06512: 行6

ORA-06512: "SYS.DBMS\_SQL", 行1766

このインスタンスでは普通の表でも同じエラーが発生することも確認できました。東京リージョンの19cで普通の表を対象に、同じ手順でアプリケーションを作成すると、問題なく全文検索ができるので、おそらくAlways Freeの21c固有の事象ではないかと思われます。

完

Yuji N. 時刻: 10:39

共有

◀

ホーム

▶

[ウェブ バージョンを表示](#)

自己紹介

Yuji N.

日本オラクル株式会社に勤務していて、Oracle APEXのGroundbreaker Advocateを拝命しました。こちらの記事につきましては、免責事項の参照をお願いいたします。

[詳細プロフィールを表示](#)

Powered by Blogger.