Connexion à une Base de données distante

André Miralles

Extension DBLINK

- **Extension commune à plusieurs SGBD**
  - Oracle, PostgreSQL…

- **Liste des commandes**

dblink_connect	opens a persistent connection to a remote database
dblink_connect_u	opens a persistent connection to a remote database, insecurely
dblink_disconnect	closes a persistent connection to a remote database
dblink	executes a query in a remote database
dblink_exec	executes a command in a remote database
dblink_open	opens a cursor in a remote database
dblink_fetch	returns rows from an open cursor in a remote database
dblink_close	closes a cursor in a remote database
dblink_get_connections	returns the names of all open named dblink connections
dblink_error_message	gets last error message on the named connection
dblink_send_query	sends an async query to a remote database
dblink_is_busy	checks if connection is busy with an async query
dblink_get_notify	retrieve async notifications on a connection
dblink_get_result	gets an async query result
dblink_cancel_query	cancels any active query on the named connection
dblink_get_pkey	returns the positions and field names of a relation's primary key fields
dblink_build_sql_insert	builds an INSERT statement using a local tuple, replacing the primary key field values with alternative supplied values
dblink_build_sql_delete	builds a DELETE statement using supplied values for primary key field values
dblink_build_sql_update	builds an UPDATE statement using a local tuple, replacing the primary key field values with alternative supplied values

Extension DBLINK

- **Différentes étapes de mise en œuvre**
  - **Installation de l'extension dblink**
  - **Création d'une connexion**
    - **dblink_connect**
  - **Création d'une table distante**
    - **Dblink**
  - **Désactivation d'une connexion**
    - **dblink_disconnect**

Extension DBLINK

- **Gestion de l'extension**
  - CREATE EXTENSION dblink;
  - DROP EXTENSION dblink;

- **Connexion à une base de données distante**
  - dblink_connect(text connname, text conndef)
    - Renvoie un text
    - Paramètre conndef
      - 'host=… port=… user=… password=… dbname=…'
        - host= localhost | adresse IP | URL
        - Valeur par défaut du port : 5432
  - Mise en œuvre
    - SELECT dblink_connect(…);

Extension DBLINK

- **Déconnexion à une base de données distante**
  - dblink_disconnect(text connname)
    - Renvoie un text
  - Mise en œuvre
    - SELECT dblink_disconnect(…);
- **Exécution de requêtes**
  - dblink(text connname, text query [, bool fail_on_error])
    - Renvoie un ensemble de record
  - dblink(text connstr, text query [, bool fail_on_error])
    - Renvoie un ensemble de record
  - dblink(text query [, bool fail_on_error])
    - Renvoie un ensemble de record

Extension *DBLINK*

- **Exemple 1**
  - SELECT *
    - FROM dblink('con_name', myQuery1, true)
      - AS a(colname1 varchar, colname2 varchar);
  - myQuery1 => 'SELECT nom, prenom FROM abonne WHERE nom=''DUPONT''';
- **Exemple 2**
  - SELECT *
    - FROM dblink('con_name', myQuery2, true)
      - AS a(colname1 varchar, colname2 varchar)
        - WHERE colname1='DUPONT';
  - myQuery2 => 'SELECT nom, prenom FROM abonne‘
- **Pourquoi privilégier l’écriture de l’exemple 1 ?**

Extension *DBLINK*

- **Exemple 3**
  - CREATE TABLE dupont_list AS
    - SELECT * FROM dblink('con_name', myQuery1, true)
      - AS a(colname1 varchar, colname2 varchar);
  - Importation des données
    - Les données sont dans la table même si la connexion est désactivée
    - Pas de mise à jour

- **Exemple 4**
  - CREATE VIEW dupont_list AS
    - SELECT * FROM dblink('con_name', myQuery1, true)
      - AS a(colname1 varchar, colname2 varchar);
  - Accès dynamique aux données SEULEMENT si la connexion est ACTIVE
    - Mise à jour dynamique des données

Extension *Foreign data wrappers*

- **Extension plus flexible que DBLink**
- **Différentes étapes de mise en œuvre**
  - **Installation de l'extension postgres_fdw**
  - **Création d'un serveur distant**
    - **CREATE SERVER**
  - **Création d'un user**
    - **CREATE USER MAPPING**
  - **Création d'une table distante**
    - **CREATE FOREIGN TABLE**
    - **IMPORT FOREIGN SCHEMA**

Extension *Foreign data wrappers*

- **Gestion de l'extension**
  - CREATE EXTENSION postgres_fdw;
  - DROP EXTENSION postgres_fdw;
- **Création d'un SERVEUR DISTANT**
  - CREATE SERVER server_name [TYPE 'server_type'] [VERSION 'server_version']
    - FOREIGN DATA WRAPPER fdw_name
      - [OPTIONS (option 'value' [, ...])];
  - Option list
    - host '...', dbname '...', port '...'
- **Suppression d'un SERVEUR DISTANT**
  - DROP SERVER server_name [, ...] [CASCADE|RESTRICT];

Extension *Foreign data wrappers*

- **Création d'un USER**
  - CREATE USER MAPPING FOR {user_name|USER|CURRENT_USER|PUBLIC}
    - SERVER server_name
      - [OPTIONS (option 'value' [, ...])];
  - Option list
    - user '...', password '...'

- **Suppression d'un USER**
  - DROP USER MAPPING FOR {user_name|USER|CURRENT_USER|PUBLIC}
    - SERVER server_name;

Extension *Foreign data wrappers*

- **Création d'une TABLE DISTANTE**
  - **CREATE FOREIGN TABLE table_name**
    - **([{ column_name data_type [OPTIONS (option 'value' [, ...])]**
      [column_constraint [, ...]] | table_constraint } [, ...]])
      - **[INHERITS (parent_table [, ...])]**
        - **SERVER server_name**
          - [OPTIONS (option 'value' [, ...])];

- **Suppression d'une TABLE DISTANTE**
  - **DROP FOREIGN TABLE table_name [, ...] [CASCADE|RESTRICT];**

Extension *Foreign data wrappers*

- **Importation d'un SCHEMA DISTANT**
  - **IMPORT FOREIGN SCHEMA remote_schema**
    - **[{LIMIT TO|EXCEPT} (table_name [, ...])]**
      - **FROM SERVER server_name**
        - **INTO local_schema**
          - [OPTIONS (option 'value' [, ...])];

- **Exemple 1**
  - **CREATE FOREIGN TABLE tmp.abonne_ft (nom varchar, prenom varchar)**
    - **SERVER tp11_bibliotheque_foreign_server**
      - **OPTIONS (schema_name 'public', table_name 'abonne');**

- **Exemple 2**
  - **IMPORT FOREIGN SCHEMA public**
    - **LIMIT TO (abonne, livre)**
      - **FROM SERVER tp11_bibliotheque_foreign_server**
        - **INTO tmp;**

Quelques sites utiles

Quelques sites utiles

- **DBLINK**
  - https://www.postgresql.org/docs/12/dblink.html
  - http://www.postgresonline.com/journal/archives/44-Using-DbLink-to-access-other-PostgreSQL-Databases-and-Servers.html
- **Foreign data wrappers**
  - https://www.postgresql.org/docs/12/postgres-fdw.html
  - https://www.postgresql.org/docs/12/sql-importforeignschema.html