Greenplum　gpload 命令使用

目录

Greenplum　gpload 命令使用 1
1 查看 gpload 帮助 1
2 编写 yml 文件 16
3 查看需要导入的数据 17
4 创建需要插入的表 17
5 使用 gpload 加载数据 17
6 使用 COPY 加载数据 18
7 查看数据的行数与大小 18
7.1 查看 gpload 表的信息 18
7.2 查看 COPY 表的信息 19

1 查看 gpload 帮助

$ gpload --help
COMMAND NAME: gpload

Runs a load job as defined in a YAML formatted control file.

****
SYNOPSIS
****

gpload -f <control_file> [-l <log_file>] [-h <hostname>] [-p <port>]
[-U <username>] [-d <database>] [-W] [--gpfdist_timeout <seconds>]
[--no_auto_trans] [[-v | -V] [-q]] [-D]

gpload -?

gpload --version

****
PREREQUISITES
****

The client machine where gpload is executed must have the following:

* Python 2.6.2 or later, pygresql (the Python interface to PostgreSQL),
  and pyyaml. Note that Python and the required Python libraries are
  included with the Greenplum Database server installation, so if you have
  Greenplum Database installed on the machine where gpload is running, you
  do not need a separate Python installation.

  Note: Greenplum Loaders for Windows supports only Python 2.5
  (available from www.python.org).

* The gpfdist parallel file distribution program installed and in your
  $PATH. This program is located in $GPHOME/bin of your Greenplum Database
  server installation.

* Network access to and from all hosts in your Greenplum Database array
  (master and segments).

* Network access to and from the hosts where the data to be loaded
  resides (ETL servers).

****
DESCRIPTION
****

gpload is a data loading utility that acts as an interface to Greenplum
Databases external table parallel loading feature. Using a load
specification defined in a YAML formatted control file, gpload executes
a load by invoking the Greenplum parallel file server (gpfdist),
creating an external table definition based on the source data defined,
and executing an INSERT, UPDATE or MERGE operation to load the source
data into the target table in the database.

The operation, including any SQL commands specified in the SQL
collection of the YAML control file, are performed as a single
transaction to prevent inconsistent data when performing multiple,
simultaneous load operations on a target table.

****
OPTIONS
****

-f <control_file>

Required. A YAML file that contains the load specification details. See following section "Control File Format".

--gpfdist_timeout <seconds>

Sets the timeout for the gpfdist parallel file distribution program to send a response. Enter a value from 0 to 30 seconds (entering "0" to disables timeouts). Note that you might need to increase this value when operating on high-traffic networks.

-l <log_file>

Specifies where to write the log file. Defaults to ~/gpAdminLogs/gpload_YYYYMMDD. See Also: LOG FILE FORMAT section.

--no_auto_trans

Specify --no_auto_trans to disable processing the load operation as a single transaction if you are performing a single load operation on the target table.

By default, gpload processes each load operation as a single transaction to prevent inconsistent data when performing multiple, simultaneous operations on a target table.

-q (no screen output)

Run in quiet mode. Command output is not displayed on the screen, but is still written to the log file.

-D (debug mode)

Check for error conditions, but do not execute the load.

-v (verbose mode)

Show verbose output of the load steps as they are executed.

-V (very verbose mode)

Shows very verbose output.

-? (show help)

Show help, then exit.

--version

Show the version of this utility, then exit.

****
CONNECTION OPTIONS
****

-d <database>

The database to load into. If not specified, reads from the load control file, the environment variable $PGDATABASE or defaults to the current system user name.

-h <hostname>

Specifies the host name of the machine on which the Greenplum master database server is running. If not specified, reads from the load control file, the environment variable $PGHOST or defaults to localhost.

-p <port>

Specifies the TCP port on which the Greenplum master database server is listening for connections. If not specified, reads from the load control file, the environment variable $PGPORT or defaults to 5432.

-U <username>

The database role name to connect as. If not specified, reads from the

load control file, the environment variable $PGUSER or defaults to the current system user name.

-W (force password prompt)

Force a password prompt. If not specified, reads the password from the environment variable $PGPASSWORD or from a password file specified by $PGPASSFILE or in ~/.pgpass. If these are not set, then gploadwill prompt for a password even if -W is not supplied.

****
CONTROL FILE FORMAT
****

The gpload control file uses the YAML 1.1 document format and then implements its own schema for defining the various steps of a Greenplum Database load operation. The control file must be a valid YAML document.

The gpload program processes the control file document in order and uses indentation (spaces) to determine the document hierarchy and the relationships of the sections to one another. The use of white space is significant. White space should not be used simply for formatting purposes, and tabs should not be used at all.

The basic structure of a load control file is:

```
---
VERSION: 1.0.0.1
DATABASE: <db_name>
USER: <db_username>
HOST: <master_hostname>
PORT: <master_port>
GPLOAD:
    INPUT:
     - SOURCE:
          LOCAL_HOSTNAME:
            - <hostname_or_ip>
          PORT: <http_port>
        | PORT_RANGE: [<start_port_range>, <end_port_range>]
          FILE:
            - </path/to/input_file>
          SSL: true | false
```

```
            CERTIFICATES_PATH: </path/to/certificates>
    - COLUMNS:
                - <field_name>: <data_type>
    - TRANSFORM: '<transformation>'
    - TRANSFORM_CONFIG: '<configuration-file-path>'
    - MAX_LINE_LENGTH: <integer>
    - FORMAT: text | csv
    - DELIMITER: '<delimiter_character>'
    - ESCAPE: '<escape_character>' | 'OFF'
    - NULL_AS: '<null_string>'
    - FORCE_NOT_NULL: true | false
    - QUOTE: '<csv_quote_character>'
    - HEADER: true | false
    - ENCODING: <database_encoding>
    - ERROR_LIMIT: <integer>
    - ERROR_TABLE: <schema>.<table_name>
    - LOG_ERRORS: true | false
  EXTERNAL:
    - SCHEMA: <schema> | '%'
  OUTPUT:
    - TABLE: <schema>.<table_name>
    - MODE: insert | update | merge
    - MATCH_COLUMNS:
                - <target_column_name>
    - UPDATE_COLUMNS:
                - <target_column_name>
    - UPDATE_CONDITION: '<boolean_condition>'
    - MAPPING:
                <target_column_name>: <source_column_name> | '<expression>'
  PRELOAD:
    - TRUNCATE: true | false
    - REUSE_TABLES: true | false
  SQL:
    - BEFORE: "<sql_command>"
    - AFTER: "<sql_command>"


**************************************************
CONTROL FILE SCHEMA ELEMENT DESCRIPTIONS
**************************************************


VERSION - Optional. The version of the gpload control file
            schema. The current version is 1.0.0.1.
```

DATABASE - Optional. Specifies which database in Greenplum to
connect to. If not specified, defaults to $PGDATABASE
if set or the current system user name. You can also
specify the database on the command line using the -d option.

USER - Optional. Specifies which database role to use to connect.
If not specified, defaults to the current user or $PGUSER if set.
You can also specify the database role on the command line using
the -U option.

HOST - Optional. Specifies Greenplum master host name. If not specified,
defaults to localhost or $PGHOST if set. You can also specify the
master host name on the command line using the -h option.

PORT - Optional. Specifies Greenplum master port. If not specified, defaults
to 5432 or $PGPORT if set. You can also specify the master port on
the command line using the -p option.

GPLOAD - Required. Begins the load specification section. A GPLOAD specification
must have an INPUT and an OUTPUT section defined.

INPUT - Required. Defines the location and the format of the input data to
be loaded. gpload will start one or more instances of the gpfdist file
distribution program on the current host and create the required external
table definition(s) in Greenplum Database that point to the source data.
Note that the host from which you run gpload must be accessible over the
network by all Greenplum hosts (master and segments).

SOURCE - Required. The SOURCE block of an INPUT specification defines the
location of a source file. An INPUT section can have more than one
SOURCE block defined. Each SOURCE block defined corresponds to one
instance of the gpfdist file distribution program that will be started
on the local machine. Each SOURCE block defined must have a FILE
specification.

For more information about using the gpfdist parallel file server
and single and multiple gpfdist instances, see the "Greenplum Database
Database Administrator Guide."

LOCAL_HOSTNAME - Optional. Specifies the host name or IP address of the local
machine on which gpload is running. If this machine is configured
with multiple network interface cards (NICs), you can specify the
host name or IP of each individual NIC to allow network traffic

to use all NICs simultaneously. The default is to use the local machines primary host name or IP only.

PORT - Optional. Specifies the specific port number that the gpfdist file distribution program should use. You can also supply a PORT_RANGE to select an available port from the specified range. If both PORT and PORT_RANGE are defined, then PORT takes precedence. If neither PORT or PORT_RANGE are defined, the default is to select an available port between 8000 and 9000.

If multiple host names are declared in LOCAL_HOSTNAME, this port number is used for all hosts. This configuration is desired if you want to use all NICs to load the same file or set of files in a given directory location.

PORT_RANGE - Optional. Can be used instead of PORT to supply a range of port numbers from which gpload can choose an available port for this instance of the gpfdist file distribution program.

FILE - Required. Specifies the location of a file, named pipe, or directory location on the local file system that contains data to be loaded. You can declare more than one file so long as the data is of the same format in all files specified.

If the files are compressed using gzip or bzip2 (have a .gz or .bz2 file extension), the files will be uncompressed automatically (provided that gunzip or bunzip2 is in your path).

When specifying which source files to load, you can use the wildcard character (*) or other C-style pattern matching to denote multiple files. The files specified are assumed to be relative to the current directory from which gpload is executed (or you can declare an absolute path).

SSL - Optional. Specifies usage of SSL encryption. If SSL is set to true, gpload starts the gpfdist server with the --ssl option and uses the gpfdists protocol.

CERTIFICATES_PATH - Required when SSL is true; cannot be specified when SSL is false or unspecified. The location specified in CERTIFICATES_PATH must contain the following files:

* The server certificate file, server.crt
* The server private key file, server.key
* The trusted certificate authorities, root.crt

The root directory (/) cannot be specified as

CERTIFICATES_PATH.

COLUMNS - Optional. Specifies the schema of the source data file(s) in the format of <field_name>: <data_type>. The DELIMITER character in the source file is what separates two data value fields (columns). A row is determined by a line feed character (0x0a).

If the input COLUMNS are not specified, then the schema of the output TABLE is implied, meaning that the source data must have the same column order, number of columns, and data format as the target table.

The default source-to-target mapping is based on a match of column names as defined in this section and the column names in the target TABLE. This default mapping can be overridden using the MAPPING section.

TRANSFORM - Optional. Specifies the name of the input XML transformation passed to gpload. For more information about XML transformations, see the "Greenplum Database Database Administrator Guide."

TRANSFORM_CONFIG - Optional. Specifies the location of the XML transformation configuration file that is specified in the TRANSFORM parameter, above.

MAX_LINE_LENGTH - Optional. An integer that specifies the maximum length of a line in the XML transformation data passed to gpload.

FORMAT - Optional. Specifies the format of the source data file(s) - either plain text (TEXT) or comma separated values (CSV) format. Defaults to TEXT if not specified. For more information about the format of the source data, see the "Greenplum Database Database Administrator Guide."

DELIMITER - Optional. Specifies a single ASCII character that separates columns within each row (line) of data. The default is a tab character in TEXT mode, a comma in CSV mode. You can also specify a non-printable ASCII character or a non-printable unicode character, for example: "\x1B" or "\u001B". The escape string syntax, E'<character-code>', is also supported for non-printable characters. The ASCII or unicode character must be enclosed in single quotes. For example: E'\x1B' or E'\u001B'.

ESCAPE - Specifies the single character that is used for C escape sequences (such as \n,\t,\100, and so on) and for escaping data characters that might otherwise be taken as row or column delimiters. Make sure to choose an escape character that is not used anywhere in your actual column data. The default escape character is a \ (backslash) for

text-formatted files and a " (double quote) for csv-formatted files, however it is possible to specify another character to represent an escape. It is also possible to disable escaping in text-formatted files by specifying the value 'OFF' as the escape value. This is very useful for data such as text-formatted web log data that has many embedded backslashes that are not intended to be escapes.

NULL_AS - Optional. Specifies the string that represents a null value. The default is \N (backslash-N) in TEXT mode, and an empty value with no quotations in CSV mode. You might prefer an empty string even in TEXT mode for cases where you do not want to distinguish nulls from empty strings. Any source data item that matches this string will be considered a null value.

FORCE_NOT_NULL - Optional. In CSV mode, processes each specified column as though it were quoted and hence not a NULL value. For the default null string in CSV mode (nothing between two delimiters), this causes missing values to be evaluated as zero-length strings.

QUOTE - Required when FORMAT is CSV. Specifies the quotation character for CSV mode. The default is double-quote (").

HEADER - Optional. Specifies that the first line in the data file(s) is a header row (contains the names of the columns) and should not be included as data to be loaded. If using multiple data source files, all files must have a header row. The default is to assume that the input files do not have a header row.

ENCODING - Optional. Character set encoding of the source data. Specify a string constant (such as 'SQL_ASCII'), an integer encoding number, or 'DEFAULT' to use the default client encoding. If not specified, the default client encoding is used. For information about supported character sets, see the "Greenplum Database Reference Guide."

ERROR_LIMIT - Optional. Enables single row error isolation mode for this load operation. When enabled, input rows that have format errors will be discarded provided that the error limit count is not reached on any Greenplum segment instance during input processing. If the error limit is not reached, all good rows will be loaded and any error rows will either be discarded or logged to the table specified in ERROR_TABLE. The default is to abort the load operation on the first error encountered. Note that single row error isolation

only applies to data rows with format errors; for example, extra or missing attributes, attributes of a wrong data type, or invalid client encoding sequences. Constraint errors, such as primary key violations, will still cause the load operation to abort if encountered. For information about handling load errors, see the "Greenplum Database Database Administrator Guide."

ERROR_TABLE - Deprecated, LOG_ERRORS is encouraged instead.
Optional when ERROR_LIMIT is declared. Specifies an error table where rows with formatting errors will be logged when running in single row error isolation mode. You can then examine this error table to see error rows that were not loaded (if any).
If the error_table specified already exists, it will be used.
If it does not exist, it will be automatically generated.
For more information about handling load errors, see the "Greenplum Database Database Administrator Guide."

LOG_ERRORS - Optional when ERROR_LIMIT is declared. If true(default false), gpload would create an internal error table where rows with formatting errors will be logged when running in single row error isolation mode. You can then examine this error table by using GPDB built-in function gp_read_error_log() to see error rows that were not loaded (if any). For more information about handling load errors, see the "Greenplum Database Database Administrator Guide."
NOTE: LOG_ERRORS is not allowed to use together with ERROR_TABLE.

EXTERNAL - Optional. Defines the schema of the external table database objects created by gpload. The default is to use the Greenplum Database search_path.

SCHEMA - Required when EXTERNAL is declared. The name of the schema of the external table. If the schema does not exist, an error is returned.

If % (percent character) is specified, the schema of the table name specified by TABLE in the OUTPUT section is used. If the table name does not specify a schema, the default schema is used.

OUTPUT - Required. Defines the target table and final data column values that are to be loaded into the database.

TABLE - Required. The name of the target table to load into.

MODE - Optional. Defaults to INSERT if not specified. There are three available load modes:

INSERT - Loads data into the target table using the following
method: INSERT INTO target_table SELECT * FROM input_data;

UPDATE - Updates the UPDATE_COLUMNS of the target table
where the rows have MATCH_COLUMNS attribute values equal
to those of the input data, and the optional UPDATE_CONDITION is true.

MERGE - Inserts new rows and updates the UPDATE_COLUMNS of
existing rows where MATCH_COLUMNS attribute values are equal
to those of the input data, and the optional UPDATE_CONDITION
is true. New rows are identified when the MATCH_COLUMNS value
in the source data does not have a corresponding value in the
existing data of the target table. In those cases, the entire
row from the source file is inserted, not only the MATCH and
UPDATE columns. If there are multiple new MATCH_COLUMNS values
that are the same, only one new row for that value will be inserted.
Use UPDATE_CONDITION to filter out the rows to discard.

MATCH_COLUMNS - Required if MODE is UPDATE or MERGE. Specifies the column(s)
to use as the join condition for the update. The attribute
value in the specified target column(s) must be equal to that
of the corresponding source data column(s) in order for the
row to be updated in the target table.

UPDATE_COLUMNS - Required if MODE is UPDATE or MERGE. Specifies the
column(s) to update for the rows that meet the
MATCH_COLUMNS criteria and the optional UPDATE_CONDITION.

UPDATE_CONDITION - Optional. Specifies a Boolean condition (similar to
what you would declare in a WHERE clause) that must
be met in order for a row in the target table to be
updated (or inserted in the case of a MERGE).

MAPPING - Optional. If a mapping is specified, it overrides the default
source-to-target column mapping. The default source-to-target
mapping is based on a match of column names as defined in the
source COLUMNS section and the column names of the target TABLE.
A mapping is specified as either:
<target_column_name>: <source_column_name>
or
<target_column_name>: '<expression>'

Where expression is any expression that you would specify in the

SELECT list of a query, such as a constant value, a column reference, an operator invocation, a function call, and so on.

PRELOAD - Optional. Specifies operations to run prior to the load operation. Right now the only preload operation is TRUNCATE.

TRUNCATE - Optional. If set to true, gpload will remove all rows in the target table prior to loading it.

REUSE_TABLES - Optional. If set to true, gpload will not drop the external table objects and staging table objects it creates. These objects will be reused for future load operations that use the same load specifications. This improves performance of trickle loads (ongoing small loads to the same target table).

SQL - Optional. Defines SQL commands to run before and/or after the load operation. You can specify multiple BEFORE and/or AFTER commands. List commands in the order of desired execution.

BEFORE - Optional. An SQL command to run before the load operation starts. Enclose commands in quotes.

AFTER - Optional. An SQL command to run after the load operation completes. Enclose commands in quotes.

****
NOTES
****
If your database object names were created using a double-quoted identifier (delimited identifier), you must specify the delimited name within single quotes in the gpload control file. For example, if you create a table as follows:

```
CREATE TABLE "MyTable" ("MyColumn" text);
```

Your YAML-formatted gpload control file would refer to the above table and column names as follows:

```
- COLUMNS:
  - '"MyColumn"': text

OUTPUT:
  - TABLE: public.'"MyTable"'
```

****
LOG FILE FORMAT
****

Log files output by gpload have the following format:

<timestamp>|<level>|<message>

Where <timestamp> takes the form: YYYY-MM-DD HH:MM:SS,
<level> is one of DEBUG, LOG, INFO, ERROR,
and <message> is a normal text message.

Some INFO messages that may be of interest
in the log files are (where # corresponds
to the actual number of seconds, units of data,
or failed rows):

INFO|running time: #.## seconds
INFO|transferred #.# kB of #.# kB.
INFO|gpload succeeded
INFO|gpload succeeded with warnings
INFO|gpload failed
INFO|1 bad row
INFO|# bad rows

****
EXAMPLES
****

Run a load job as defined in my_load.yml:

    gpload -f my_load.yml

Example load control file:

VERSION: 1.0.0.1
DATABASE: ops
USER: gpadmin
HOST: mdw-1

```
PORT: 5432
GPLOAD:
    INPUT:
     - SOURCE:
              LOCAL_HOSTNAME:
                - etl1-1
                - etl1-2
                - etl1-3
                - etl1-4
              PORT: 8081
              FILE:
                - /var/load/data/*
     - COLUMNS:
                - name: text
                - amount: float4
                - category: text
                - desc: text
                - date: date
     - FORMAT: text
     - DELIMITER: '|'
     - ERROR_LIMIT: 25
     - LOG_ERRORS: True
    OUTPUT:
     - TABLE: payables.expenses
     - MODE: INSERT
    SQL:
    - BEFORE: "INSERT INTO audit VALUES('start', current_timestamp)"
    - AFTER: "INSERT INTO audit VALUES('end', current_timestamp)"


**************************************************
SEE ALSO
**************************************************

gpfdist, CREATE EXTERNAL TABLE

See the "Greenplum Database Reference Guide" for information
about CREATE EXTERNAL TABLE.

ERROR: configuration file required
```

请仔细阅读 gpload 命令的详细使用文档

2 编写 yml 文件

$ cat test-gpload.yml

```
---
VERSION: 1.0.0.1
DATABASE: stagging
USER: gpadmin
HOST: 192.****.11
PORT: 5432
GPLOAD:
    INPUT:
     - SOURCE:
            LOCAL_HOSTNAME:
              - gpmdw
            PORT: 8081
            FILE:
              - /home/xiaoxu/test/date-dir/b.txt
     - COLUMNS:
              - filed1: text
              - filed2: varchar
     - FORMAT: text
     - DELIMITER: '|'
     - ERROR_LIMIT: 25
     - LOG_ERRORS: true
    OUTPUT:
     - TABLE: xiaoxu.test_yml
     - MODE: INSERT
    PRELOAD:
     - REUSE_TABLES: true
    SQL:
    - BEFORE: "truncate table xiaoxu.test_yml"
    - AFTER: "analyze xiaoxu.test_yml"
```

参数说明

/home/xiaoxu/test/date-dir/b.txt 下也可以使用多个匹配的模式例如

/home/xiaoxu/date-dir/*

gpmdw：是在脚本机器上的名字，也可以写成 IP 地址

BEFORE：是在插入数据之前的操作

AFTER：是插入之后的一些操作

3 查看需要导入的数据

```
# head -n 5 b.txt
A|1
A|2
A|3
A|4
A|5
**********

# du -sh b.txt
20G Dec 20 14:20 b.txt


# du -sh xaa
1.1G xaa
```

4 创建需要插入的表

以下这张表是用于 gpload 插入的

```
create table xiaoxu.test_yml(filed1 text,filed2 varchar)
with (appendonly = true, compresstype = zlib, compresslevel = 5,orientation=column)
Distributed by (filed2)
```

以下表时 copy 命令插入的

```
create table xiaoxu.test_yml_copy(filed1 text,filed2 varchar)
with (appendonly = true, compresstype = zlib, compresslevel = 5,orientation=column)
Distributed by (filed2)
```

5 使用 gpload 加载数据

```
$ time gpload -f my_load.yml
2018-12-20 14:28:50|INFO|gpload session started 2018-12-20 14:28:50
2018-12-20 14:28:50|INFO|started gpfdist -p 8081 -P 8082 -f "/home/xiaoxu/test/date-dir/b.txt"
-t 30
2018-12-20 14:28:50|INFO|did not find an external table to reuse. creating
ext_gpload_reusable_83bde63c_0420_11e9_a106_801844f3abb8
2018-12-20 14:32:40|INFO|running time: 230.02 seconds
2018-12-20 14:32:40|INFO|rows Inserted          = 4346958300
```

```
2018-12-20 14:32:40|INFO|rows Updated                  = 0
2018-12-20 14:32:40|INFO|data formatting errors = 0
2018-12-20 14:32:40|INFO|gpload succeeded

real  3m50.170s
user 0m0.190s
sys   0m0.148s
```

在以上中可以看出 gpload 先是调用 gpdist 命令开启了一个端口，然后再使用外表的形式插入到内表中，会生成唯一的 ID，本次的是 creatingext_gpload_reusable_83bde63c_0420_11e9_a106_801844f3abb8 本次插入的行数为 4346958300，错误行位 0，用时 3m50.170s

6 查看临时表的数据

6 使用 COPY 加载数据

```
$ time psql -d stagging    -h 192.****.11 -p 5432 -U gpadmin -c "COPY xiaoxu.test_yml_copy FROM '/home/xiaoxu/test/date-dir/xaa' WITH csv DELIMITER '|'";
COPY 235011866

real  4m1.774s
user 0m0.002s
sys   0m0.004s
```

由于 copy 加载数据太慢了，所以使用 235011866 行的数据，大概用时 4m1.774s

7 查看数据的行数与大小

7.1 查看 gpload 表的信息

```
stagging=# select count(*) from xiaoxu.test_yml;
     count
------------
  4346958300
(1 row)

Time: 32269.097 ms
stagging=# select pg_size_pretty(pg_relation_size('xiaoxu.test_yml'));
  pg_size_pretty
----------------
```

```
  95 MB
(1 row)

Time: 9.040 ms
```

7.2 查看 COPY 表的信息

```
stagging=# select count(*) from xiaoxu.test_yml_copy;
     count
-----------
  235011866
(1 row)
Time: 3322.220 ms


stagging=# select pg_size_pretty(pg_relation_size('xiaoxu.test_yml_copy'));
  pg_size_pretty
----------------
  3960 kB
(1 row)
Time: 32.605 ms
```

由于使用了高度压缩方式，导致 copy 加载数据过慢，带来的