

1 使用解説書

この章では、OSDL DBT-1 PostgreSQL(ODBC)版について、測定を実施するための手順について記述する。各測定には、オペレーティングシステム/ミドルウェア/負荷ツールがそれぞれ必要であり、各対象に応じてそれぞれを適宜組み合わせて、環境を構築する。

1.1 環境設定

1.1.1 Linux のインストール

ディストリビューションのデフォルトインストールを想定している。インストールに関してはディストリビューションのインストレーション・マニュアル等に従うこと。

対象ディストリビューション： MIRACLE LINUX V3.0 Asianux Inside

注)DBT-1 の測定には ssh、gcc、sysstat、libstdc++6.2.2 PostgreSQL 用には、autoconf2.59 が必要。また、グラフ作成用に、gnuplot 4.0 が別途必要となる。

1.1.2 DBT-1 用オプション設定

プログラム内部では、スレッド単位にソケットを開くため、1 プロセス内で、同時に開くことが出来るファイル数の上限数が少ないと、スレッドを生成しても DB 接続に失敗する。そのため、設定ファイル(/etc/security/limits.conf)に定義を追加し、上限を引き上げておく必要がある。

/etc/security/limits.conf

```
* soft nfile 4096
* hard nfile 65536
```

“nfile”の記述が無い場合は、上記 2 行を追加し、記述がある場合は値を修正する。MIRACLE LINUX の場合、hard はデフォルトで 65536 に設定されているため、変更は不要。

1.2 ミドルウェア

1.2.1 PostgreSQL のインストール

PostgreSQL は、ここではソースからインストールする。一般的なインストール方法と異ならないが、ここでは PostgreSQL のスーパーユーザ名などを以下で指定している。手順は以下の通りである。

今回の測定開始時点で、使用したバージョンは、7.4.6、8.0.0beta5、8.0.3、8.1beta1 であったためそれぞれ適切な tarball ファイルを、入手する。複数のバージョンを同時にインストールする場合、使用するディレクトリが競合しないように工夫する必要がある。

root ユーザで、PostgreSQL の所有者となる Linux ユーザとして、pgsql ユーザと pgsq1 グループを作成する。

```
# groupadd pgsq1
# useradd pgsq1 g pgsq1
```

PostgreSQL のソースコードアーカイブを展開するディレクトリと、PostgreSQL をインストールするディレクトリを作成して、ディレクトリの所有者を pgsq1 ユーザにする。

```
# mkdir /usr/local/src/postgresql-x.x.x
# chown pgsq1 /usr/local/src/postgresql-x.x.x
# mkdir /usr/local/pgsq1
# chown pgsq1 /usr/local/pgsq1
```

pgsq1 ユーザで、PostgreSQL のソースコードアーカイブを展開し、展開したディレクトリに移動する。PostgreSQL のソースコードアーカイブは、/tmp ディレクトリにあるものとする。x.x.x は、使用する tarball のレベルである。

```
# su pgsq1
$ cd /usr/local/src
$ tar xzf /tmp/postgresql-x.x.x.tar.gz
$ cd postgresql-x.x.x
```

展開した PostgreSQL のソースコードをコンパイルし、インストールする。

```
$ ./configure
$ gmake all
$ gmake install
```

1.2.1.1 PostgreSQL用のUnixODBCのインストール

DBT-1 PostgreSQL(ODBC)を測定する場合，UnixODBC ドライバ・マネージャをインストールする必要がある。

Linux 上で ODBC によるプログラムを使用するためには，ODBC ドライバ・マネージャと PostgreSQL 用の ODBC ドライバをインストールする必要がある。ここでは，ODBC ドライバ・マネージャとして UnixODBC を使用する。ソースコードは，以下の URL より入手した。

<http://www.unixodbc.org/>

```
$ su root
# mkdir /usr/local/src/unixODBC-2.2.11
# chown pgsq1 /usr/local/src/unixODBC-2.2.11
# su - pgsq1
$ cd /usr/local/src
$ tar xzf /tmp/unixODBC-2.2.11.tar.gz
$ cd unixODBC-2.2.11
$ ./configure
$ make
$ su root
# make install
```

1.2.1.2 psqlodbc ドライバのインストール

DBT-1 PostgreSQL(ODBC)を測定する場合，postgreSQL 用 ODBC ドライバを unixODBC ドライバマネージャに対応するようにインストールする必要がある。psqlodbc ドライバは，デフォルトで unixODBC ドライバマネージャに対応している。最新の psqlodbc ドライバを，以下の PostgreSQL 関連プロジェクト・コミュニティ Gborg プロジェクトから入手した。

入手元 <http://gborg.postgresql.org/project/psqlodbc/projdisplay.php>

```
# mkdir /usr/local/src/psqlodbc-08.00.0102
# chown pgsq1 /usr/local/src/psqlodbc-08.00.0102
# su - pgsq1
$ cd /usr/local/src
$ tar xzf /tmp/psqlodbc-08.00.0102.tar.gz
$ cd psqlodbc-08.00.0102
$ ./configure
$ make
```

```
$ su root
# make install
```

DBT1 の c のプログラムが ODBC 経由で PostgreSQL に接続を行なうための設定を行なう。pgsql のホームディレクトリに下記内容の .odbc.ini ファイルを作成する。

```
[ODBC Data Sources]
DBT1 =Postgres DBT1
[DBT1]
Servername = localhost
Database = DBT1
Driver = /usr/local/lib/psqlodbc.so
Port = 5432
ReadOnly=No
```

1.3 負荷ツール

1.3.1 DBT-1 用の事前環境設定

1.3.1.1 ユーザの作成

PostgreSQL では、必ず PostgreSQL のスーパー・ユーザ ID=pgsql,グループ=pgsql パスワード = psql で実行すること。

1.3.1.2 sudo の準備

ログ情報には、root 権限が必要なコマンドがあるため、DBT-1 を実行するユーザが sudo を利用できるよう、/etc/sudoers ファイルを編集する。編集方法は、root ユーザで visudo を起動し、以下の 1 行を追加する。

1.3.1.3 PostgreSQL

```
pgsql ALL=(ALL) NOPASSWD: ALL
```

1.3.1.4 ユーザ環境設定

(1) PostgreSQL(ODBC)

pgsql ユーザでログインし、 ~/.bashrc ファイル 内に以下の設定を追加する。

```
export PGDIR=/usr/local/pgsql
export PATH=$PATH:$PGDIR:$PGDIR/bin
export PGUSER=pgsql
export PGDATA=/mydata/pgsql/data
export SID1=DBT1
export LD_LIBRARY_PATH=/usr/local/lib:/usr/local/pgsql/lib
```

1.3.2 DBT-1 のインストール

1.3.2.1 PostgreSQL(ODBC)版

(1) DBT-1 のダウンロード

pgsql ユーザでログインする。

PostgreSQL(ODBC)用 DBT-1 ソースをユーザのホームディレクトリにダウンロードする
(IPA サイトより)

(2) DBT-1 の解凍

次のコマンドで DBT-1 を展開する。

```
$ tar -xzvf dbt1-v2.1-PostgreSQL-ODBC-1.0.tar.gz
```

(3) DBT-1 のコンパイル

コンパイルする。

```
$ cd ~/dbt1-v2.1  
$ ./configure - with-postgresql-odbc - without-sapdb  
$ make  
$ make install
```

1.4 DBT-1 評価手順

1.4.1 前提条件

MIRACLE LINUX の導入するパッケージは標準構成をベースとし、ディストリビューションに含まれる CD の内容のみ使用する。但し DBT-1 を実行するために必要なパッケージは適宜追加する。

1.4.2 テストデータの作成

1.4.2.1 データ用ディレクトリの作成

pgsql でログインする。
データを格納するディレクトリを設定する

```
$ mkdir ~/data  
$ chmod a+w ~/data
```

1.4.2.2 データの生成

次のコマンドでテスト用の基礎データ(Text ファイル)の生成を行う

PostgreSQL

```
$ cd ~/dbt1-v2.1/datagen  
$ ./datagen -d PGSQL -i 10000 -u 1000 -p /home/pgsql/data
```

オプションの解説

-d : データベースタイプ(SAPDB または PGSQL。)

-i : 生成するアイテム総数

-u : 生成する仮想ユーザ数

-p : 実際に生成するデータディレクトリ

-T : データを生成するテーブル名

i -- item

c -- customer

a -- author

d -- address

o -- order

-p に与えるパスは、フルパスで指定する。

指定したディレクトリ内にデータが作成される。

実際に生成されたデータファイルには/tmp ディレクトリからシンボリックリンクが張られる。

データロードには、/tmp ディレクトリからのパスが使用される。

上記パラメータで作成されるデータの総容量は、約 3.5GB となる。

1.4.3 データベースの作成

1.4.3.1 PostgreSQL(ODBC)

ユーザ・ルートでデータベース用のディレクトリを作成する。

```
# mkdir p /mydata/pgsql/data  
# chown R postgres /mydata/pgsql
```

次のコマンドで PostgreSQL にデータベースを作成する。デフォルトで作成されるデータベース名は DBT1 となる。

```
# su postgres  
$ initdb --no-locale --encoding =EUC_JP  
$ cd ~/dbt1-v2.1/scripts/postgresql  
$ ./build_db.sh c tcp_socket=on 0 0
```

‘(シングルコーテーション)の前後には、空白を必ず挿入し、実行すること。

1.4.4 パラメータの設定

1.4.4.1 DBT-1 の環境設定

エディタで DBT-1 設定ファイル `~/dbt1-v2.1/scripts/stats/dbt1.config` を編集する。

- ・ホスト名を `localhost` に修正
- ・ディレクトリを各バイナリが存在するディレクトリに修正
- ・総エミュレートユーザ数を計測を行う仮想ユーザ数に修正

MaxDB,MySQL の `dbt1.config` の各行は以下の通り。

```
[database]
#hostname instance username password
localhost:DBT1:dbt:dbt

[cache]
#hostname port dbconnections items appCache_executable_directory
localhost:9999:5:10000:/home/dbt/dbt1-v2.1/cache

[appServer]
#hostname                               appServer を実行するサーバ名
localhost
#port                                   appServer への接続ポート番号
9992
#dbconnection                           RDBMS コネクション数
20
#transaction_queue_size                 dbdriver 通信バッファ
1000
#transaction_array_size                 各処理バッファ
1000
#items                                   生成アイテム数
10000
#appServer executable directory         appServer 実行ディレクトリ
/home/dbt/dbt1-v2.1/appServer

[dbdriver]
#hostname                               dbdriver を実行するホスト名
localhost
#items                                   生成アイテム数
10000
#customers                               生成顧客数
2880000
```

#eu	総エミュレートユーザ数
200	
#eu/min	ユーザ接続レート(ユーザ/分)
100	
#mean think_time	ユーザオペレーション間隔
7.2	
#run_duration in seconds	各ユーザ実行時間
4100	
#dbdriver executable directory	dbdriver 実行ディレクトリ
/home/dbt/dbt1-v2.1/dbdriver	

PostgreSQL の場合の各行は以下のとおり

[database]	
#hostname instance username password	
localhost:DBT1:pgsql:pgsql	
[cache]	
#hostname port dbconnections items appCache_executable_directory	
localhost:9999:5:10000:/home/pgsql/dbt1-v2.1/cache	
[appServer]	
#hostname	appServer を実行するサーバ名
localhost	
#port	appServer への接続ポート番号
9992	
#dbconnection	PostgreSQL コネクション数
20	
#transaction_queue_size	dbdriver 通信バッファ
1500	
#transaction_array_size	各処理バッファ
1500	
#items	生成アイテム数
10000	
#appServer executable directory	appServer 実行ディレクトリ
/home/pgsql/dbt1-v2.1/appServer	
[dbdriver]	
#hostname	dbdriver を実行するホスト名
localhost	
#items	生成アイテム数
10000	
#customers	生成顧客数
2880000	
#eu	総エミュレートユーザ数
400	
#eu/min	ユーザ接続レート(ユーザ/分)
100	
#mean think_time	ユーザオペレーション間隔
7.2	
#run_duration in seconds	各ユーザ実行時間
2400	
#dbdriver executable directory	dbdriver 実行ディレクトリ
/home/pgsql/dbt1-v2.1/dbdriver	

今回の評価では、エミュレートユーザ数のみを変えて評価を実施する。(下線部) 複数のサーバに実行を分散する場合、サーバの数だけ、全項目を":"で区切って定義を追加する。

1.4.5 起動方法

1.4.5.1 データベースの起動

(1) PostgreSQL

DBT-1 実行中の PostgreSQL の稼動情報は、PostgreSQL の統計情報収集機能 (stats collector) により取得できる。この機能を有効にする場合は、\$PGDATA/postgresql.conf の関連項目を編集し、PostgreSQL を再起動すること。

```
stats_start_collector = true
stats_command_string = true
stats_block_level = true
stats_row_level = true
stats_reset_on_server_start = true
```

PostgreSQL を起動する。起動している場合は必要ないが、PostgreSQL 内部のキャッシュなどをクリアするには、再起動することが必要である。pgsql ユーザで、以下のコマンドを実行する。

```
$ pg_ctl stop
$ pg_ctl start
```

1.4.5.2 トランザクション・パラメータの設定

トランザクション作業領域の容量が不足すると、DB とは関係の無い箇所で処理待ちが生じるため、以下の値は"#eu"以上の値を指定する。

```
#transaction_queue_size
#transaction_array_size
```

今回の試験では、エミュレートユーザ数による変化を検証するため、"# eu"以外の値は修正を加えない。使用する値は、OSDL-J にて公開されている結果をベースとしている。

1.4.5.3 測定

次のコマンドで DBT-1 を測定する。

PostgreSQL(ODBC)の場合

```
$ cd ~/dbt1-v2.1/scripts/stats
$ ./run_dbt1.sh /home/pgsql/U200
```

今回は、DB の性能評価であるため、appCache を使用しない。

run_dbt1.sh に引数として与えるディレクトリは事前に作成する必要はない。

引数のディレクトリ(/home/dbt/U200 または/home/pgsql/U200)に BT ファイルが作成され作成されており、最終行の「total errors」が 0 となっていれば、測定の成功となる。

以下に、BT ファイルの例を示す。

Interaction	%%	Avg. Response Time (s)
Admin Confirm	0.08	0.767
Admin Request	0.09	0.702
Best Sellers	4.95	1.928
Buy Confirm	1.21	0.853
Buy Request	2.59	0.861
Customer Registration	2.99	0.000
Home	16.75	0.821
New Products	5.00	1.921
Order Display	0.66	0.774
Order Inquiry	0.74	0.712
Product Detail	16.86	0.730
Search Request	19.79	0.000
Search Results	16.79	1.489
Shopping Cart	11.50	0.841
248.3 bogotransactions per second		
49.9 minute duration		
total bogotransactions 742911		
total errors 0		

1.4.6 PostgreSQL のログ情報

表 5.6-3 PostgreSQL(ODBC)ログ情報一覧

情報種別	コマンド	出力ファイル名	備考
システム統計	sar -A	run.sar.data	10 秒間隔
I/O 情報	iostat -d	io.txt	10 秒間隔
プロセス一覧	top d 120 b	top.txt	120 秒間隔
接続セッション情報	select * from pg_stat_activity	db_activity*.out	*:取得回数
内部キャッシュ・ファイルシステム I/O 統計	select * from pg_stat_database where datname='DBT 1';	db_load*.out	*:取得回数
ユーザ インデックスに関する内部キャッシュ・ファイルシステム I/O 統計	select relid, indexrelid, relname, indexrelname idx_blks_read, idx_blks_hit from pg_statio_user_indexes	index_info*.out	*:取得回数
ユーザ インデックス問い合わせ数の統計	select * from pg_stat_user_indexes;	indexes_scan*.out	*:取得回数
ロックに関する情報	select relname,pid, mode, granted from pg_locks, pg_class where relfilenode = relation	lockstats*.out	*:取得回数
ユーザ テーブルに関する内部キャッシュ・ファイルシステム I/O 統計	select relid, relname, heap_blks_read,heap_blks_hit,idx_blks_read, idx_blks_hit from pg_statio_user_tables;	table_info*.out	*:取得回数
ユーザ テーブルの問い合わせ方法の統計	select * from pg_stat_user_tables;	table_scan*.out	*:取得回数
ロックに関する情報	select * from pg_locks where transaction is not NULL;	tran_lock*.out	*:取得回数

1.4.7 DBT-1 のログ情報

表 5.6-4 DBT-1 実行結果を記録するファイル

ファイル名	説明
BT	インタラクション別平均応答時間、BT/s 等 OSDL DBT-1 測定結果が格納される。
mix.log	dbdriver から appServer へのリクエストインタラクション、応答時間(msec)が記録される。BT ファイルを計算するための、元情報。