

DB性能評価における実システムの疑似方式
(DBprobe-reverse)

5G-5

夏目義久 八田功 赤間浩樹 長濱芳寛

NTT情報通信研究所

1. はじめに

ダウンサイジングやマルチベンダ化に伴いDBシステムの構築に関する性能トラブルが多発している。

その解決を支援するため、我々はこれまで以下のような機能を持つDBの性能評価ツールDBprobeを開発してきた。^{[1]-[5]}

- ・マルチベンダに対応したDBシステムの簡易自動構築機能とDBMS物理情報のチューニング機能
- ・DBの性能に関する各種統計情報の集計とリアルタイムでビジュアルな比較表示機能、および、表計算ソフトへのインタフェース機能

2. 課題

既存（または開発直後）の実DBシステムをチューニングする時、以下のような理由から実環境と評価環境が異なる場合が存在した。

- ・実環境でチューニング等の負荷の大きな作業を行いたくない。
- ・実環境の評価ツールが貧弱なため別の評価環境上でチューニングを行いたい。

チューニングの指標には、応答時間とスループットがあるが、一般のチューニングでは単にDML単位の応答時間を短くするだけではなく、トランザクションの応答時間のある範囲内に抑えつつトランザクションのスループットを最高にすることが求められる。

そのためには、評価環境でトランザクションを再現する必要があり、これまでは実環境のAPを評価環境に移植しなければならなかった。しかし、AP全体の移植はコスト的にも負担が大きいため、実際にはDML部分だけを移植し、単に応答時間だけのチューニングで終了することが多かった。

よって、この評価指標のズレを防ぐための、評価環境での簡易なトランザクションの再現方法が求められていた。

3. DBprobe-reverseの疑似方式

評価環境においてトランザクションを再現するためにDBprobe-reverseでは、実環境のトレース情報を利用してAPのトランザクションの疑似的な再現を行う方法を実現した。

そのアルゴリズムの概要を次に示す。

Simulation method of DB system in DBprobe
Yoshihisa NATSUME, Isao HATTA
Hiroki AKAMA, Yoshihiro NAGAHAMA
NTT Information and Communication Systems Labs.

【step1】

実APのソースプログラムのDMLの前後に実行トレース情報取得点（getlog関数）を埋め込む（図1）。このgetlog関数では、APのID、プロセスのID、トレース情報取得点のID、時刻情報などの取得と共に、DMLの入出力のホスト変数値も保存し、その結果として、図2のようなトレース情報を取得する。

複数のAPやプロセスを実行した場合には、それぞれに対してトレース情報を取得する。

```
H2=net_read();
dbprobe_getlog(1,1,H2);
EXEC SQL SELECT C1 FROM T1 WHERE C2=:H2;
dbprobe_getlog(2,1,C1);
H3=net_write(C1);
dbprobe_getlog(3,2,H3,H2);
EXEC SQL UPDATE T1 SET C3=:H3 WHERE C2=:H2;
dbprobe_getlog(4,0);
...
```

図1 トレース情報取得点(下線部)の埋め込み後

```
APID:1 PID:1 TID:1 TIME:1300 HVAR:2000
APID:1 PID:1 TID:2 TIME:2350 HVAR:32
APID:1 PID:1 TID:3 TIME:2500 HVAR:105 2000
APID:1 PID:1 TID:4 TIME:4200
...
```

図2 トレース情報の例

【step2】

実環境のトレース情報を評価環境に移す。このとき文字コード変換などを行う。なお、本方式はAPの疑似方式であるので、DBのスキーマやデータ実体等については、これとは別に評価環境へ移行する必要がある。

【step3】

図3のフローチャートに従った疑似APを作成する。

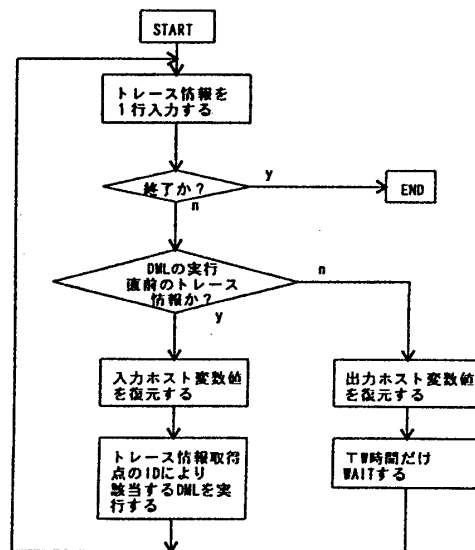


図3 疑似APのフローチャート

このとき、疑似AP中には、実AP内のDMLの埋め込みを行う。

さらに、実環境と評価環境のDML実行時間比 (α) を用いて疑似AP内のDML実行時間とWAIT時間 (T_w) の比を調整し、実環境のAPのトランザクションを疑似する。

T0: 前回入力されたトレース行内の時刻
(DML終了直後の時刻)
T1: 入力されたトレース行内の時刻
(DML実行直前の時刻)
TA: 実環境でのDML実行時間
TB: 評価環境でのDML実行時間
 α : TB/TA (DML実行時間比)

$$T_w = \alpha (T1 - T0)$$

【step4】

トレース情報を入力として疑似APを実行する。

【step5】

疑似APのトランザクションの応答時間は α 倍されている。よって、そのスループット値の基準時間も α 倍されていることになり、疑似APのスループット値を α 倍した値が、評価環境における実環境の疑似スループット値となる。

ただし、 α はAP毎(厳密にはDML毎)に異なるので、DBシステムが複数のAPから構成される場合には、その値の平均をスループットの基準時間として用いる。

4. 実験と評価

本疑似方式が、実環境のスループットをどの程度疑似できるかについての実験を行った。

【実験の方法】

2台の全く異なるマシン (M_A , M_B) を用意し、 M_A を実環境、 M_B を評価環境と考える。なお、DBMSは同一のものを使用する。

ここで、それぞれのマシン上で同一のAPを動かし、以下の値を求める。実験の結果を図4に示す。

A: M_A 上でのスループット
B: M_B 上でのスループット
R: M_A のAPを本疑似方式によってマシン M_B 上で再現したスループット

なお、APには、会計処理をもとに作成した8つの評価用APを利用した。

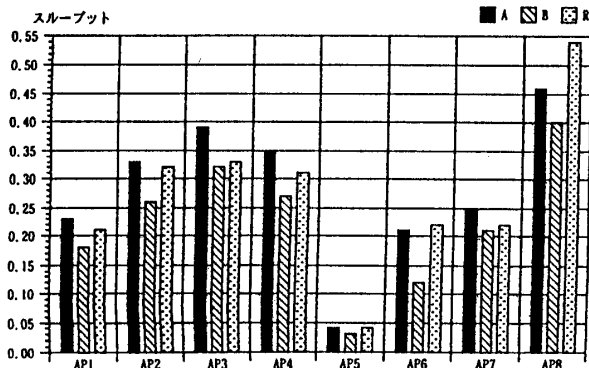


図4 AP1~AP8のスループット(A, B, R)の比較

【評価】

実験の結果から以下のことが得られた。

- 本方式 (R) は従来のDML移植 (B) による方法に比べても、遜色のないスループットの再現が可能である。
- 従来の方法で本方式よりも近似率が悪い場合が存在する理由は、たとえ、APを移植しても、マシン間でのCPU処理やI/O処理の性能比の変化により、スループットの忠実な再現が難しいためである。

5. まとめ

本稿では、我々のDB性能評価システムにおける実DBシステムの疑似方式について述べた。

本方式は以下の特徴によって、トランザクションの応答時間やスループットの疑似的な再現を可能にする。

- (1) トレース情報を使ってDBアクセスの順序再現やタイミングの疑似を行う。
- (2) DML実行の直前、直後のホスト変数値の値を使って、DML間の応答時間の比や資源のロック状況の疑似を行う。
- (3) 実環境と評価環境のDML実行時間比 (α) を用いて、WAIT時間やスループット基準時間を調整することで、実環境と評価環境の測定結果の比較を可能にする。

さらに、本方式の最大の効果は、APの疑似精度を落とすことなく、AP移植に比べ大幅な工程の自動化を可能にする点にある。これにより、評価環境でのスループットを指標としたチューニングを低コストで行うことが可能になった。

今後は、DBprobe-reverseと自動チューニングシステム (DBprobe-tune) の連携を行い、スループットの自動チューニングを目指していきたい。

参考文献

- [1] 八田, 赤間, 友野, 武田, "DB性能検証システムDBprobe-link/view", 情処48全国大会 2F-5
- [2] 長谷川, 赤間, 武田, "DBMS物理情報のチューニング支援システムDBprobe-tune", 情処48全国大会 2F-4
- [3] 石垣, "データベースシステムの性能実測評価環境について", 情処SIG-DBS98-7, 1994.5
- [4] 長濱, 赤間, 八田, "市販データベース管理システムの最適利用に向けて", NTT技術ジャーナル, 1994.10
- [5] 赤間, 石垣, "ダウンサイジングにおけるデータベース利用の課題と対策", Computer Today, 1994.3