

分散型業務アプリケーション性能評価 支援システムの開発

5N-9

保田 正則 岸田 一
(株)情報技術コンソーシアム

1 はじめに

今日、分散システム上に業務アプリケーションを構築する事例が増えているが、その基本設計に関する定量的評価技術は十分に確立されていない。一般的に分散型業務システムの設計案はいくつか考えられ、かつその処理性能は実際のデータ・アクセスパターンに大きく影響を受けるため性能予測が難しく最適な基本設計案を選択することは困難である[1]。

本稿では、データの分散パターンとトランザクションの性質等に基づき、分散システムの基本設計を性能に関して定量的に評価するための性能評価支援システムの試作について報告する。

2 分散型業務システム評価パラメータ

分散型業務システムの基本設計における性能は以下の要素に大きく依存するので、これらを性能評価支援システムの評価パラメータとして使用する。

(1)構成パラメータ

- ・業務システムの基本設計
 - └ 処理の定義
(処理内容、1トランザクションの大きさ、ネストの深さ)
 - └ データの定義
(データ構造、レコードサイズ、フィールドサイズ)
- ・分散配置定義
 - └ 処理分散定義
 - └ データ分散定義
 - └ テーブルの水平分散
 - └ テーブルの垂直分散
(レプリケーションの度合)
- ・システム構成定義
 - └ マシン機種
 - └ ネットワーク・トポロジー
 - └ ミドルウェア構成
(DBMS,TP-monitor等)

(2)トランザクション実行パラメータ

- ・単位時間当たり頻度
- ・参照データ量/回
- ・テーブル当たりの参照/更新比率
- ・総件数
- ・更新データ量/回
- ・アクセスの局所度
- ・マシン処理負荷の度合

3 分散型業務アプリケーション性能評価支援システム概要

分散型業務アプリケーション性能評価支援システムは、あらかじめ設計された分散型業務システムの基本設計モデルの妥当性を構成パラメータ、トランザクション実行パラメータを変動させつつ実証的に評価するためのシステムである。

分散型業務アプリケーション性能評価支援システムの基本概念を図1に示す。評価システムは大きくアプリケーション定義部(処理、データ定義)、評価環境生成ツール、評価制御ツールよりなり、評価環境生成ツールはテストテーブル作成ツールとプロセス分散配置ツールよりなる。

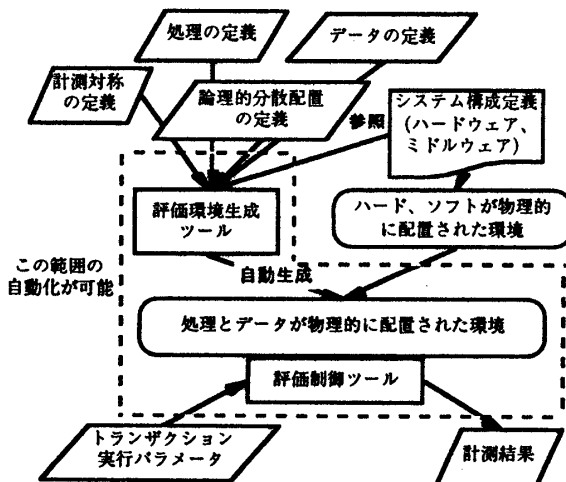


図1 評価システム基本概念

ハードウェア、ミドルウェアの物理的構成環境(テストベッド・システム)は手動で構築する。データの定義は図2のような記述で行い、これをテストテーブル作成ツールに入力することによりランダ

ムなテストテーブルを所定のノードへ作成する。

```
defTable zaikoMaster
set hostname {node1 node2}
set database zaiko
set rec_size 10000
set fields(ID){integer{not null primary key}{from 10000}}
set fields(QUANT){char{not null}{string 32}}
set foreign_key ""
set references ""
set create_index(ID)
enddef
```

図2 テストテーブル作成分散配置定義

処理の定義は、現在C言語+SQL文をコーディングすることにより行う。処理の分散配置の定義は、図3のような記述で行い、これをプロセス分散配置ツールに入力することにより、指定ノードへ指定ミドルウェア用の実行モジュールを配置する。

```
LayProcess proc_def1
process1
set target {oracle}
set host {node1:/dir1 node2:/dir2}
set tableAccess {{table1 size1 node1} {table2 size2 node2}}
process2
set target {oracle tuxedo}
set host {node1:/dir1 node3:/dir3}
set tableAccess {{table1 size1 node1} {table2 size2 node3}}
enddef
```

図3 プロセス分散配置定義

処理時間の計測は、各クライアント/サーバー・モジュールのソースプログラム中に時間計測関数を埋め込み、この関数が計測結果を所定のログファイルへ書き込むことにより行う。評価環境制御ツールは、起動時刻、クライアント・モジュール名、起動ノード名、アクセスノード名、アクセスの局所度、トランザクション発生間隔を1ジョブとして指定(図4)することにより起動スクリプトを生成し、対象ノードにて自動起動を行う。尚、参照/更新比率は起動トランザクションの種類にて制御する。

```
defTransaction
23:00 client1 {{host node1}}{dest{node2 0.8}{node3 0.2}}
{poasson 100} {times 1000}}
23:00 client2 {{host node1}}{dest{node2 0.5}{node3 0.3}}
{node4 0.2}}{poasson 10} {times 1000}}
enddef
```

図4 トランザクション実行定義

4 評価

UNIXサーバー3台(分散機能をサポートする市販のRDBMSをインストール)とクライアント用のUNIXワークステーションからなるテストベッド・システムを構築した。このシステム上に、ある在庫管理システムのプロトタイプを作成した。分散配置

の1パターンとして、全体在庫データ及び在庫参照/更新処理、集計(バッチ)処理をサーバー1台(ASV)に、ローカルデータ及び在庫参照/更新処理をサーバー2台(LSV1,LSV2)に配置した。トランザクション実行パラメータとしてローカル・データの参照+更新100件を連続して行うトランザクションを起動した結果を表1に示す。表中No.1~4は表中の割合(アクセスの局所度)にて各サーバーへアクセスを行った場合、No.3,4はさらにASVにてバッチ処理を起動しつつアクセスを行った場合である。

表1 アプリケーション性能評価結果

No.	トランザクション起動パターン	バッチ処理	処理時間比
1	LSV1:80%,LSV2:10%,ASV:10%	なし	1
2	LSV1:50%,LSV2:30%,ASV:20%	なし	1.26
3	LSV1:80%,LSV2:10%,ASV:10%	ASVにて	3.37
4	LSV1:50%,LSV2:30%,ASV:20%	ASVにて	4.21

上例よりアクセスの局所性の度合い及びバッチ処理(ロックの競合+処理負荷の度合い)がオンライン・トランザクション処理性能に影響を与えていることがわかる。上表のデータに示されるような影響の度合いは予測困難であり、本稿のような評価支援システムが有効であることがわかる。

5 おわりに

分散データベースを用いた業務システムの性能評価を支援するシステムの開発を行った。

今後は処理定義部分を、トランザクション特性が処理性能に及ぼす影響を考慮した簡潔な記述へ定式化することによる、より効率的な評価方法の検討及び評価支援ツールの開発を予定している。

6 あとがき

この研究は、情報処理振興事業協会(IPA)の先進的情報処理技術の開発促進事業で実施されている「広域分散ソフトウェア生産技術開発」プロジェクトで行われた。

参考文献

- [1] Maria R.Ebling and M.Satyanarayana, 「SynRGen:An Extensible File Reference Generator」, SIGMETRICS5/94ACM(1994)