

プロトタイプによる分散システム性能設計のための支援ツール

3K-2

保田正則、山本勉、鈴木真理子、原田稔

(株)情報技術コンソーシアム

1 はじめに

分散システム上にアプリケーションを構築する際、処理とデータの配置の複雑さ及びインテグレートする製品組み合わせの複雑さにより性能予測が困難であり、これが基本設計を困難にする要因となっている。我々は評価環境上で基本設計案のプロトタイプを構築・評価し、結果を基本設計にフィードバックする方法を提案しているが、この方法を効率良く実現するために、主に分散データベースを用いたアプリケーションの配置、性能・資源使用率の計測、評価を支援するツールを開発したので報告する。

2 分散型業務アプリケーション基本設計方法

集中型システムと比べ、分散型アプリケーションは処理とデータの配置パターンが複数考えられ、性能予測が困難である。このような分散型業務アプリケーションの設計として、我々は、1)論理設計、2)分散化論理設計、3)分散化物理設計、4)評価環境上へのプロトタイプ構築、5)プロトタイプ性能評価、6)評価結果の設計へのフィードバックというステップを踏むことを提案している。ここで分散化論理設計とはプロセス（プログラム）とデータの配置を定義した論理設計である。本設計方法の特徴は、分散化論理設計を含む点、分散化論理設計に基づくプロトタイプを構築・評価し、意図した性能が得られなければ分散化論理設計を再考し、プロトタイプ構築・評価を繰り返し行う点にある。尚、分散配置の最小単位は、プロセス、データベース・テーブルであり、データの分散は水平分割、垂直分割、レプリケーションにより行う。

3 分散型アプリケーション性能評価支援ツール概要

分散型アプリケーション性能評価支援ツールは分散型アプリケーションの基本設計案を評価環境上に構築し、トランザクションを起動し、性能・資源使用率を計測し、設計案を評価することを支援するツール群である。

本ツールはアプリケーションの配置に伴うソースコードの書き換え、トランザクション起動の制御、計測データの整

理等の煩雑な作業を効率よく実行することを目的としている。ツールの基本概念を図1に示す。尚、トランザクション起動制御ツール、テストデータ生成ツールについては文献[1]参照のこと。

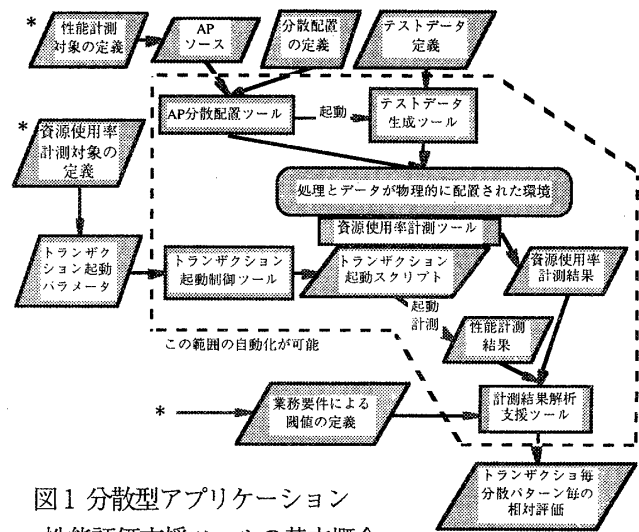


図1 分散型アプリケーション性能評価支援ツールの基本概念

3.1 アプリケーション分散配置ツール

設計者は、ある1つのマシン上に、論理設計に基づくプロトタイプを作成し、分散化論理設計に基づき評価環境上のプロセスとデータの配置を行う。このとき配置に伴いソースコードのデータアクセス部分をノード名を明記した記述に変更し、配置先に実行モジュール、データを再構築する必要がある。アプリケーション分散配置ツールはこの作業を自動的に行うことを目的としている。データアクセス部分の変更が必要なのは、ローカルアクセスからリモートアクセスとなる場合である。アプリケーション分散配置ツールは、論理設計におけるプロセスとデータの依存関係を記述したアクセス定義と分散化論理設計における配置定義を入力とする。これらの記述例を以下に示す。

P1:P2:D1
P3::D1,D2#1

図2 アクセス定義

S1:P1,P2:D1
S2:P3:D2#2

図3 配置定義

図2の1行目は、P1がP2,D1へアクセスすることを表す。

図3の1行目は処理P1,P2,データD1をマシンS1へ配置することを表す。この例ではP3からD1へのアクセスがリモートアクセスとなりソースコードが変更される。また、D2#1,D2#2はテーブルD2の水平分割テーブルを表し、この場合ソースコード上は同じテーブル名D2で変更は行わないが、作成されるテーブルのレコードの範囲は各ノードで異なる。テストデータは実際のアプリケーションが利用するデータの値の範囲と分布から自動生成したものである。このテストデータを配置定義に従って所定のマシン上に作成する。また、処理時間の計測のためにソースコード中に計測の開始/終了関数を埋め込み、この関数が計測情報をローカルマシンのログに書き出す。処理時間計測関数はC言語のライブラリとして実装した。

3.2 資源使用率計測ツール

性能が要求を満たさない原因は計算機資源の使用状況が飽和状態であることが多い。このため分散配置された実際のシステムの稼働中の資源使用率を計測する必要がある。資源使用率計測ツールは、OSとミドルウェアの資源使用情報を取得するツールで、CPU、ディスクI/O、ネットワークI/O、データベースファイルI/O、TPモニタの要求数等の情報(表1)を取得する。これらは計算機資源情報の殆どを網羅しており、性能評価のための一連のトランザクションを実施させたときの資源使用状況を取得できる。また、アプリケーションとの関連付けを容易にするために、プロセス、テーブルスペース単位の情報も取得できる。

対象資源	測定項目	測定単位
CPU	使用時間、使用率	ノード、プロセス
ディスク	稼働時間、命令回数、転送量	インタフェース
ネットワーク	送受信パケット数、コリジョン数等	インタフェース
メモリ	使用量、残り使用可能量	ノード、プロセス
DBMS	データベースファイルI/O時間等	テーブルスペース
TPモニタ	要求完了数、キュー内の要求数等	ノード

表1 取得できる情報一覧

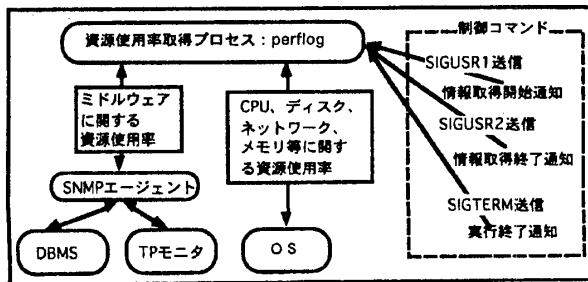


図4 資源使用率取得ツール構成概要

資源使用率計測ツールの構成概要を図4に示す。情報のうちOSに関するものは、稼働中のOSのカーネルメモリ情報から取得する。一方、ミドルウェアに関する情報は、SNMPエージェントを介して情報を取得する。図4のSNMPエージェントと資源使用率取得プロセスを測定対象マシン上で起動すると資源使用率取得プロセスはメモリに常駐して要求が来るのを待ち、トランザクション起動と同時に情報取得を開始し、トランザクション終了と同時に取得した情報をログに書き出す。

3.3 計測結果解析支援ツール

計測結果解析支援ツールは、処理時間、資源使用率のログを分散パターン毎、トランザクション毎、資源毎に集計、グラフ化し処理時間の分布、資源使用状況を一覧し設計の良否を解析することを支援する。これにより性能劣化の要因を資源ボトルネックと関連づけて解析することができ、分散化論理設計へフィードバックすることが可能である。

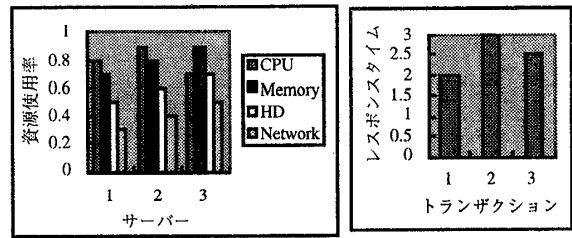


図5 計測結果解析支援ツールグラフ出力例

4 おわりに

分散型アプリケーションのプロセス及びデータを評価環境上に自動配置し、性能及び資源使用率を計測し、設計案の評価を支援するツールを開発した。これにより設計者は容易に処理とデータの配置/再配置、性能・資源使用率の計測を行うことができる。今後はこのツールとプロトタイプ評価を設計にフィードバックする方法をさらにブラッシュアップしていく予定である。

尚、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] 保田他、「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」、情報処理学会第50回全国大会、1995.3
- [2] 岸田他、広域分散ソフトウェア生産技術「実証的研究システムの概念設計」、第13回IPA技術発表会論文集、1994.10
- [3] 原田他、広域分散ソフトウェア生産技術「実証的研究システムの開発と評価」、第14回IPA技術発表会論文集、1995.10