

IA32 PC/AT システム挙動分析ツール SimAT の開発と応用

Development and Application of SimAT - PC/AT Full System Behavior Analyzer

O-29

亀山 伸 垂井 俊明 庄内 亨
Shin Kameyama Toshiaki Tarui Toru Shonai

1. まえがき

近年のインターネットビジネスの爆発的拡大によってミッションクリティカルなサーバシステムに対するニーズが増大し、それとともに PC サーバが価格性能比の良さやデスクトップ PC との使い勝手の連続性などの要因で広く普及してきた。このようなシステムでは Web やデータベースなどのアプリケーションの性能がユーザーニーズを満たす必要がある。しかし複雑化したハードウェア構成のためシステム全体の特性を明確にすることは容易ではない。またアプリの挙動はソースコードの解析によりある程度把握できるが、スケジューリング等の OS 機能の影響も考慮する必要があるため、挙動の詳細な把握は困難である。そのため、ハードとソフトを総合したシステム全体としての最適化は、現在のところ構築者の知識と経験に大きく依存し、システムが複雑になるほど構築者の負担の増大や構築期間の長期化という問題が発生する。

2. 研究の目的

システム最適化のための挙動分析手法としては、大別して実システムを用いる方法とシミュレーションによる方法がある⁽¹⁾。前者は実在のシステムしか評価できず、評価手段に柔軟性がないという欠点がある。一方後者は、ソフトウェアによる評価であるため柔軟性はあるが、シミュレーション速度が遅いという欠点がある。しかしながら事前にシステムの挙動を分析するには、柔軟なシミュレーションが実施できることは大きな長所である。そこでシミュレーション時間の欠点をカバーし、挙動分析に必要となるデータの効率の良い収集を可能にすることを目的としてフルシステムシミュレータの SimAT を開発した。以下ではその構成、機能、応用例などについて述べる。

3. SimAT の構成

フルシステムシミュレーションとは対象とするサーバシステムを仮想計算機としてホストマシン上に構築し、I/O 機器を含めたシステム全体をシミュレートすることである。フルシステムシミュレータ上で OS、アプリを実行することで OS の挙動なども含めたトレースを採取し、システムの詳細な挙動分析を行うことが可能になる。表 1 に SimAT がシミュレートする仮想計算機の仕様を示す。SimAT は図 1 に示すような複数のシミュレータで構成される。以下、それらの概略を説明する。

①CPU シミュレータ

IA32 の動作を、システム命令の実行、アドレス変換機構、割込みを含めて全てシミュレートする。メモリシステムシミュレータから、命令を 1 命令毎にフェッチし、CPU 内部状態、メモリ上のオペランドを変更する。

②PC/AT システムアーキテクチャシミュレータ

表 1 SimAT 仕様

項目	SimAT 仕様
CPU アーキテクチャ	IA32
プロセッサ構成	シングル/SMP
ディスク I/O	SCSI, DMA サポート
ディスク台数	最大 8 台
ネットワーク I/O	GigabitEther
クラスタ構成	GigabitEther クラスタ
ディスプレイ	VGA
動作 OS	Linux/Windows
最大主記憶容量	512MB
最大ディスク容量	512MB

PC の標準アーキテクチャにおける、割込み、タイマ等をサポートする。

③I/O シミュレータ

ディスク、VGA ディスプレイ、キーボードをサポートする。SimAT はディスクイメージファイルを仮想的なディスクとしてアクセスする。ディスクイメージファイル上の OS イメージをブートし、アプリを実行する。ディスプレイ、キーボードシミュレータは、シミュレーション対象の仮想マシンのコンソールを表示し、ホストマシンのキーボードからの入力を受け付ける。

④メモリシステムシミュレータ

評価対象マシンの主記憶内容を記憶する。

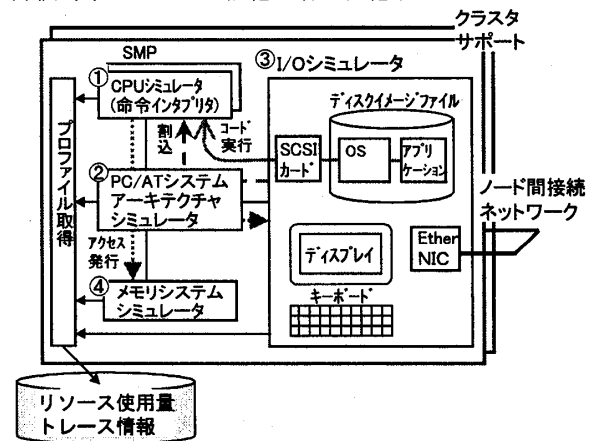


図 1 SimAT の構成

4. 主な機能

SimAT の機能はホストマシンから制御される。さらに対象とするアプリにフックを挿入することで、SimAT 上のア

プリから SimAT 自身を制御することも可能である。以下は主な機能である。

(1) プロファイル取得機能

下記のハードウェアプロファイルとソフトウェアプロファイルを任意形式でファイルへ出力する。これらのプロファイルは GUI 等によって可視化される。

<ハードウェアプロファイル>

- ・ユーザ別 CPU 使用率
- ・プロセス別 CPU 使用率
- ・メモリの read/write アクセスアドレス
- ・I/O の発生頻度

<ソフトウェアプロファイル>

- ・命令
- ・オペランド
- ・メモリアクセス
- ・タイミング情報

(2) チェックポイント機能

評価対象マシンの任意時点の内部状態(CPU のレジスタ, 主記憶の内容)をファイルに保存する。一度チェックポイントが作成されると、何度でもその時点からシミュレーションを再開させることが可能であり、シミュレーションのオーバーヘッドを短縮させることができる。

(3) 自動トレース機能

アプリのソースにフックを挿入することで、自動的にトレースの開始、終了を設定する。この機能により、長時間のシミュレーションでもトレース採取時の人工数の大幅な削減が可能になる。

5. SimAT の応用例

応用例として、Linux 上で動作するオープンソフトのアプリケーションモデル作成に関して説明する。

5.1 アプリケーションモデル

Web, AP, DB の 3 層構造のサーバシステムの性能予測を行う方法の一つに、アプリケーションモデルを用いた待ち行列シミュレーションによる方法がある⁽²⁾。アプリケーションモデルとは、図 2 に示すようにリソース消費量と消費時間を定量的に記述したソフトウェア部品の組み合わせで構成されたテンプレートである。

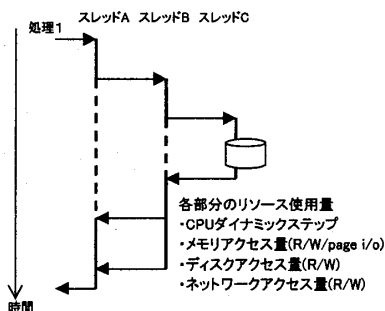


図 2 アプリケーションモデル概略

3 層の内、DB サーバに関しては一部の商用 DB に関してはモデルがすでに存在し性能予測が可能である。しかしながら AP サーバ及び Web サーバに関しては詳細なモデルは存在しないため、システム全体における総合的な性能予測を

行う場合には、これらのサーバに関する詳細モデルが必須になる。性能予測の精度はモデルの詳細さに依存するが、モデルを構成するソフトウェア部品は、これまではソースコードが参照可能であってもモデル作成者の経験を基に作成していたため精度が低かった。また精度向上も試行錯誤に頼っていたため、効率が悪いという問題があった。

5.2 アプリケーションモデルの作成

上記の問題を SimAT によって解決した。SimAT 上でアプリを実際に動作させ、ソフトウェア部品のデータを定量的に測定する。測定は下記の手順で行う。

- (1) SimAT 上に測定対象のアプリを移植する。
- (2) 性能を測定したい基本操作を単体で動作させるためのスクリプトを作成する。
- (3) (2)で策定したスクリプトを SimAT 上で動作させる。
- (4) 必要なデータモニタリング機能を有効にすることにより対象とするデータを測定する。

```
<?xml version="1.0" ?>
-<SimAT date="Thu Mar 29 14:15:28 2001">
-<log time="14900.000000 (msec)">
  <memory read="8925692 Byte" write="1764894 Byte" />
  <disk read="28 Byte" write="4120 Byte" />
  <network read="0 Byte" write="0 Byte" />
  <cpu step="2070620" />
</log>
-<log time="64466.666667 (msec)">
  <memory read="3666159 Byte" write="724938 Byte" />
  <disk read="0 Byte" write="0 Byte" />
  <network read="0 Byte" write="0 Byte" />
  <cpu step="865682" />
</log>
</SimAT>
```

図 3 トレース結果

図 3 は機能確認用 Java アプリケーションに関するモデルの一部である。アプリへのフック挿入による自動測定の結果であり、XML フォーマットで出力したものである。

6. まとめ

IA32 システムの挙動解析を行うために SimAT を開発した。SimAT の主な機能は以下のとおりである。

- ・ IA32 フルシステムシミュレーション機能
- ・ 各種データの柔軟なトレース機能
- ・ トレースの自動採取機能

また SimAT の応用例として、OS, I/O の挙動を含めたアプリケーションの挙動を定量的に記述した、システム性能予測のためのアプリケーションモデルの作成に使用した。今回の応用例はホスト OS が Linux であったが、今後はより広範囲の性能評価等に应用することを可能にするため (1) Windows 系等シミュレーション対象 OS の拡充, (2) ネットワークおよびディスク等の I/O のシミュレーション機能の拡充, を推進予定である。

参考文献

- (1) Mendel Rosnblum, 他 : Using the SimOS Machine Simulator to Study Complex Computer Systems, ACM TOMACS(平 9 - 1)
- (2) 西岡, 他 : システム性能シミュレーションのリモート実行環境の開発, 情報学会コンピュータシステムシンポジウム論文集, Vol.2001, No.16, pp.129-136, 2001年 11月