Vol.2010-EVA-33 No.2
2010/11/29

# 統合 Java アプリケーションサーバーの負荷解析

高 橋 一 晃[†1] 大 井 仁[†1]

SPECjAppserver2004 を想定した多層アプリケーションサーバーを仮想化によって統合したシステムの性能解析モデルを開発中である．本報告では 4 コアの CPU を用いたシステムで計測したデータを元に, 各ドメインにおける負荷の差, および前報告における 2 コアシステムでの計測結果との比較によりコアあたりの処理時間の変化について述べる．

# Workload Analysis of a Consolidated Java Application Server

Kazuaki Takahashi [†1] and Hitoshi Oi[†1]

With SPECjAppserver2004 as the target workload, we are currently developing a performance analysis model of a muti-tier application server consolidated by virtualization. In this work-in-progress report, we will present the measurement results including the CPU utilization on each domain, and analysis of per-core workload by comparing the results we reported previously.

## 1. はじめに

仮想化技術により, 複数のサーバー群を一つの物理システム上に統合することが可能になる．単一のシステム上で複数のサーバーを実行するマルチプログラミング環境と異なり, アプリケーション毎に異なる OS や管理ポリシーを採用することが可能になる．また物理システムの台数や消費電力の削減などによりコスト削減も期待できる．一方で, 仮想化自体に CPU やメモリーに対するオーバーヘッドがあり, 各仮想マシンへの資源割り当てが複雑になるといった短所も存在する．

現在 3 層アプリケーションサーバーである SPECjAppserver2004 を仮想化によって統合したシステムを対象とし, 各仮想マシンへの負荷や仮想化のオーバーヘッドを測定し, 得られた負荷パラメータを用いて性能解析モデルを開発中である．本報告ではスループットおよびレスポンスタイム, そして各ドメインにおける負荷について, 前報告 3) との比較などにより考察する．

## 2. SPECjAppserver2004

本節においては対象システムとして用いた SPECjAppserver2004 について概説する．詳細については SPEC 公式サイト 1) の資料を参照されたし．また本報告における計測結果は 2) Section 3.7 で定義される “Research and Academic Usage” に該当し, SPEC による公式な結果と比較してはならない．

SPECjAppserver2004 は自動車製造業をモデルとし, アプリケーションの構成としては, 一般顧客に自動車を販売する Dealer, Dealer からの注文を管理する Customer, 外部業者への部品発注を取り扱う Supplier, そして製造工程を管理する Manufacturing, 顧客 (この場合 Dealer) 情報およびその在庫を管理する Corporate の各ドメインからなる．

ソフトウェア構成としては Dealer からのトランザクション (後述) に対するインターフェースである Web サーバー, トランザクション管理などを行なうアプリケーション (App) サーバー, そして注文, 顧客, 在庫状況などを管理するデータベース (DB) サーバーからできている．システムが扱うトランザクションは以下の 5 種類: Browse (カタログの閲覧), Purchase (自動車の注文), Manage (注文や在庫の管理)．この 3 種類のトランザクションは Dealer から Web サーバー経由で発行される．Manufacturing ドメインからは自動車製造ラインを模した WorkOrder と LargeOrder の 2 種類のトランザクションが発行される．前者は定期的に実行される比較的少ない台数の場合であり, 後者は Dealer の Purchase トランザクションが 100 台以上の注文の場合に発行される．WorkOrder と LargeOrder は Web サーバーを経ず App サーバーから実行が始まる．システムに対する単位時間あたりのトランザクションの発行率はシステムの規模を表わすスケーリングファクター (SF) に比例する[⋆1]．

†1 会津大学
Aizu University

⋆1 SPECjAppserver2004 の仕様では Injection Rate (IR) と呼ばれるが SPEC による公式の性能指標との誤解を避けるため本報告では SF と記す



© 2010 Information Processing Society of Japan

## 3. 性能解析モデル

現在 4) に基いた性能解析モデルを開発中である．手法としては, システムを各サーバーを表わす待ち行列によるネットワークとみなし, 計測で得られた各サーバーがそれぞれのトランザクションを処理する時間をパラメーターとして, スループットおよび CPU 使用率を求める．4) のモデルとの主な違いは以下の 3 点である．(1) 仮想化の負荷を含め, 各仮想マシンの負荷を物理 CPU の使用率に合計し反映させている (2) マルチコアマシンにおけるコア数の変化が性能に与える影響をモデル化する (3) CPU 使用率のみを考慮し, ディスク使用率は含まない．(3) は今回測定に用いたマシンでは最大スループット時においてもディスク使用率が低かったため, という暫定な仮定であり, 将来はディスク使用率を含んだモデルに拡張する計画である．

## 4. 計 測 結 果

本節では測定結果およびその解析を示す．**表 1** の Quad 側が今回計測に用いた 4 コアシステムの仕様である．Dual 側は比較に用いる前報告 3) のデータを計測するため用いた 2 コアシステムの仕様である．

**表 1** Measurement Environment
Table 1 計測環境

| Hardware | Dual | Quad |
|---|---|---|
| CPU | Xeon 1.86GHz (3040) | Xeon 1.60GHz (e5310) |
| Memory | 4GB | 6GB |
| OS | CentOS 5.3 | |
| VMM | Xen v3.1.2 | |
| App | JBoss v4.0.0 | |
| DB | PostgreSQL v8.3.5 | |
| Domain | vCPU/Memory (GB) | |
| Web | 1/0.5 | 2/0.5 |
| App | 2/2 | 4/3 |
| DB | 1/1 | 2/1.5 |
| Dom0 | 1/0.5 | 2/0.5 |

### 4.1 スループット, トランザクション毎のレスポンスタイムの比較

**図 1** にトランザクションスループット (左) と Manage トランザクションのレスポンスタイム (90%, 右) を示す．Dual システムにおいては $SF = 8$ まで, スループットは SF に比例した値を維持しているが, それ以降は増加の割合が下がり, $SF = 14$ で飽和している．Quad システムでは $SF = 31$ まで直線的に増加した後 $SF = 32$ で飽和している．

スループットの有効性を判断する条件の一つがレスポンスタイムである．Dealer ドメインからのトランザクションに関しては, その 90%が 2 秒以内でなければならない．我々の計測した範囲では Manage トランザクションのそれがもっと性能限界に即した変化を示している (図 1 右)．Dual システムにおいては $SF = 5$ からレスポンスタイムが急速に増加し, $SF = 8$ でかろうじて限度の 2 秒を下回っている．Quad システムでは $SF = 30$ においてもスループットは SF に比例した値を保っているものの Manage トランザクションのレスポンスタイムでみると $SF = 29$ から急速に上昇している事がわかる．

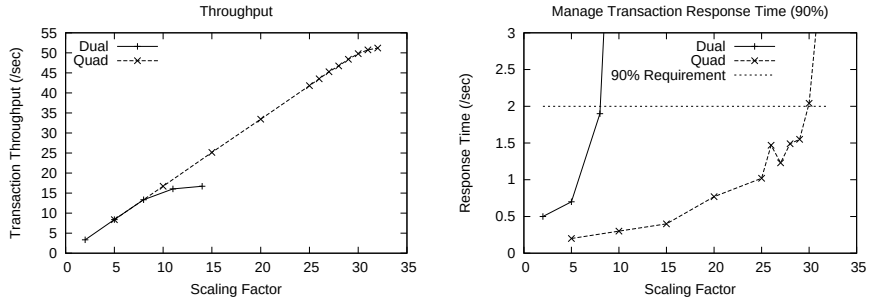


**図 1** スループットと Manage トランザクションレスポンス時間 (90%)
Fig. 1 Throughput and Manage Transaction Response Time (90%)

### 4.2 各ドメインにおける負荷の比較

**表 2** Scaling Factor and CPU Utilization Comparison
Table 2 スケーリングファクターおよび CPU 使用率の比較

| | SF | TP | CPU Utilization (%) | | | |
|---|---|---|---|---|---|---|
| | | | Web | App | DB | Dom0 |
| Dual | 8 | 13.4 | 3.0 | 58.9 | 11.9 | 7.0 |
| Quad | 29 | 48.5 | 5.2 | 51.1 | 23.3 | 10.0 |

$SF = 8$ (Dual) および $SF = 29$ (Quad) における両システムの各ドメインの CPU 使用


© 2010 Information Processing Society of Japan

**表 3** CPU Time Normalized to the Number of Transactions
Table 3 トランザクション数に正規化した CPU 使用時間

| | CPU Time/Tx (msec) | | | |
|---|---|---|---|---|
| | Web | App | DB | Dom0 |
| Dual | 22.1 | 441.1 | 88.1 | 52.7 |
| Quad | 10.7 | 105.5 | 42.0 | 20.6 |
| Ratio | 0.48 | 0.24 | 0.47 | 0.39 |

率を**表 2** に示す. 両システムにおいてもっとも負荷が高いのは App サーバーである点は同じだが負荷の相対的比率は Quad で低下している. 両システムともに飽和していないため, トランザクション間の比率は規定の値を保っている. そこで各ドメインの CPU 時間をトランザクション数で 正規化したものを**表 3** に示す. Ratio の列は Dual システムの CPU 時間を Quad システムのそれで割ったものである. 両システムで用いられている CPU がいずれも Core マイクロアーキテクチャーに基いている点 5)，6), そしてコア数およびクロック周波数の差からこの比は $1.86/(2 \times 1.6) = .58$ 程度を予想していたが, 実際の値は総じてそれよりも低い. 特に App サーバーにおいて単純換算で 4 倍以上のスピードアップになっておりスループットで 3 倍以上の差の原因となっていると考えられる.

## 5. ま と め

仮想化による統合アプリケーションサーバーの性能解析モデルのための測定結果について報告した. 現在各ドメインにおける CPU 使用率をさらにトランザクションの種類毎に分解するべく測定中であり, 今後その結果を用いて 4) と同様の手法によるモデルの構築および精度の検証を行なう予定である. また近い将来 SPECjAppserver2004 の後継ベンチマークである SPECjEnterprise2010 7) への移行も計画している.

## 参 考 文 献

1) SPECjAppServer2004, `http://www.spec.org/jAppServer2004/`
2) “SPECjAppServer2004 Run and Reporting Rules,” `http://www.spec.org/jAppServer2004/docs/RunRules.html`
3) Kazuaki Takahashi and Hitoshi Oi, “Measurement of Virtualization Overhead in a Java Application Server,” IPSJ SIG Technical Report, Vol.2010-EVA-32 No.2, August 2010.
4) Samuel Kounev, “Performance Modeling and Evaluation of Distributed Component-Based Systems Using Queueing Petri Nets,” IEEE Trans.on Software Engineering, vol.32 no.7, pp. 486-502, 2006.
5) “Intel Xeon Processor 3000 Sequence-based Platforms,” http://www.intel.com/Assets/en_US/PDF/prodbrief/318152.pdf
6) “Quad-Core Intel Xeon Processor 5300 Series,” http://www.intel.com/assets/PDF/prodbrief/xeon-5300.pdf
7) SPECjEnterprise2010, `http://www.spec.org/jEnterprise2010/`


© 2010 Information Processing Society of Japan