社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2006－EVA－17 (7)
2006／6／20

スケールアウトとスケールアップを両立する ブレードサーバ向けアーキテクチャの提案と評価

上原 敬太郎　　安井 隆　　沖津 潤
(株)日立製作所　中央研究所

Web 三階層モデル等現在の典型的なサーバシステムでは，機能に応じて複数のサーバを適宜組み合わせて用いる．そのため，サーバの種類と台数が増加し，運用管理コストの増大が問題となる．運用管理コスト削減のためには複数サーバを集約するサーバ統合が有効であり，そのためのプラットフォームとしてブレードサーバが注目されている．しかし Web 三階層のサーバ群全てを統合するためには，小規模サーバを多数並べるスケールアウトと，単体サーバ性能を増強するスケールアップの両方に対応できる必要がある．そこで本研究では，スケールアウトとスケールアップを両立するためのブレードサーバ向けアーキテクチャとして，スケーラブルブレードサーバ(SBS)を提案し，評価を行った．プロトタイプ機を用いた評価では 4 ブレード SMP 構成時に 90%という世界トップクラスのスケーラビリティを実現できる見込みを得た．

A Proposal and Evaluation of Scalable Blade Server Architecture for Scale-Out and Scale-Up Server

KEITARO UEHARA, TAKASHI YASUI and JUN OKITSU
Hitachi Ltd., Central Research Laboratory

In today's IT systems such as three-tier models, functions of servers are subdivided into several layers, and appropriate scales of servers are mixed into one system. However, the increase of kinds and amounts of servers causes the increase of management cost. To reduce the cost of management, server consolidation is one of the most effective means, and blade server systems are suitable for such consolidation. To integrate all of servers in three-tiers model, which include web servers, application servers, and database servers, blade server systems are required to support both scale-out and scale-up server for consolidation. This research proposes a new blade server architecture called scalable blade server (SBS), and evaluates the scalability of SBS prototype machine. The scalability of SBS four blades SMP configuration reaches 90%, which equals to world-wide first-ranking SMP servers.

1. 背景

オープンアーキテクチャの普及した現在の IT システムにおけるサーバシステムでは，Web 三階層モデルに見られるようにサーバ機能の分化が進み，機能に応じたサーバを適宜組み合わせて用いるようになった．これによりハードウェア・ソフトウェアの導入コストは大幅に引き下げられたが，機能毎のサーバ利用によりサーバの種類と台数が増加し，運用管理コストの増大を招いている．

運用管理コストを削減するためには複数のサーバを集約するサーバ統合が有効であり，サーバ統合のためのプラットフォームとして，サーバ群を一元管理でき，サーバ性能の柔軟な拡張が可能なブレードサーバが注目されている．しかし Web 三階層における各階層のサーバ群全てを統合するためには，Web サーバ等の比較的小規模なサーバを多数並べるスケールアウトと，データベースサーバ等に求められる単体サーバ性能を増強するスケールアップの両方に対応できる必要がある．

そこで本研究では，スケールアウトとスケールアップを両立するためのブレードサーバ向けアーキテクチャとして，スケーラブルブレードサーバ(SBS)を提案する．SBS は，複数のブレードを結合して 1 つの SMP[1]サーバとしても使用可能とするブレード間 SMP 機能を搭載し，細かい増設単位というブレードサーバとしての特長を保ちつつ，スケールアウトにもスケールアップにも対応できる柔軟性を持つ．本研究では，SBS の基本アーキテクチ

1 SMP=Symmetric Multiple Processor 対称型マルチプロセッサ

ャを設計し，プロトタイプ機による評価を行った．

2. ブレード間 SMP 機能の実現

2.1 Web 三階層システムのブレードへの統合

図 1 の上半分に一般的な Web 三階層システムの概念図を示す．Web 三階層システムは，Web サーバからなる Web 層，アプリケーションサーバからなる AP 層，データベースサーバからなる DB 層の三階層から構成される．Web 層のサーバはネット経由で送られてくる並列処理可能な多数のリクエストを受け付ける必要があるため，比較的小規模のサーバを多数並べるスケールアウトによる性能増強が効果的である．一方 DB 層のサーバは大量のデータを持つデータベースに対するクエリーを効率的に処理する必要があるため，メモリとプロセッサを追加してサーバ単体の性能を向上させるスケールアップによる性能増強が効果的である．AP 層は両者の中間の性質を持つ．

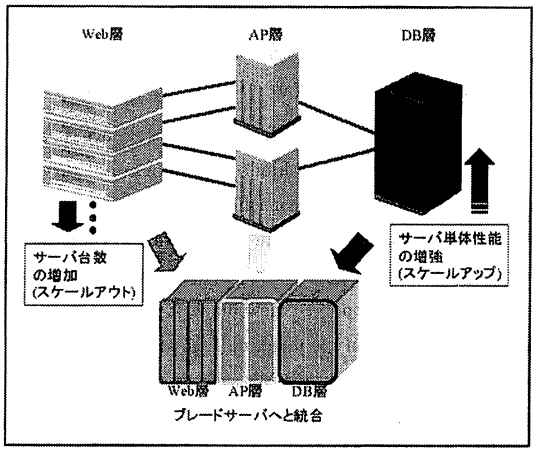

図 1. Web 三階層システムの統合

Web 三階層システムをブレードサーバ上に統合する場合には，スケールアウトとスケールアップという，異なる性質を持つサーバに適応できる必要がある．そこで SBS では，複数のブレードを結合して 1 つの SMP サーバとして使用可能とするブレード間 SMP 機能を搭載することにした．これにより，スケールアウトが必要な Web 層に対しては各ブレードをそのままサーバとして使用し，スケールアップが必要な AP 層・DB 層に対しては複数のブレードをまとめて SMP サーバとして使用することで，Web 三階層システム全てのサーバ群の統合に対応する．

2.2 ブレード間 SMP 機能の要件と課題

SBS に搭載するブレード間 SMP 機能は，次の 2 つの要件を満たす必要がある：

要件 1: 同一ブレードによりスケールアウトとスケールアップの両方に対応できること

ブレードサーバは必要に応じてブレードの数を増やすことで性能向上が図れるのが特長である．同一のブレードでスケールアウトとスケールアップの両方に対応できることで，ブレードサーバならではの柔軟な構成変更や拡張を可能にする．特にスケールアウト時の柔軟性確保のためには，ブレードの粒度（搭載 CPU 数）は小さい方が望ましい．

要件 2: アーキテクチャ依存の特別なチューニング無しで，スケールアップ時に SMP サーバとして商用 SMP 機と同等以上の性能を達成すること

Web 三階層全てを統合するためには，DB 層を統合するのに十分な性能，すなわちスケールアップ時にも耐えられるような性能を達成する必要がある．また本研究ではハードウェア・アーキテクチャに焦点を当てるため，OS やアプリケーションに対してアーキテクチャに依存した特別なチューニングは行わないことを条件とした．

以上の要件を元に，ブレード間 SMP 機能の目標を以下のように定めた：

目標 1: 1 ブレードに搭載する CPU 数は 2 とし，実用上 DB 層までカバー可能と考えられる，最大 4 ブレードで 8CPU まで増設可能とする．

目標 2: 最大構成時（4 ノード時）の目標スケーラビリティとして，商用 SMP 機を上回る 90%（1 ノードの 3.6 倍の性能）を目標とする．

以上をまとめると，ブレード間 SMP 機能設計における課題は，細かい増設単位（要件 1）と SMP 性能の維持（要件 2）の両立ということになる．

2.3 プロトタイプ機設計に向けた基本方針

SBS のブレードはスケールアウト時には単体サーバとしても使用可能とするため，各ブレードにプロセッサとメモリを搭載する．すなわち，SMP 構成時にはローカルメモリとリモートメモリが存在する ccNUMA[2]構成となる．従って，SMP 構成時のスケーラビリティを確保するためには，以下の 2

2 ccNUMA=Cache Coherent Non-Uniform Memory Access 不均一メモリアクセス

つの要素を改善する必要がある：

- リモートアクセス時のレイテンシ
- リモートアクセス時のスループット

以下，それぞれを改善するための方式を検討する．

2.3.1 レイテンシ改善方式の検討

【ブレード間接続方式の選択】

ブレード間接続の方式としては，大きく分けてバス，スイッチ，1to1 接続の 3 通りの方式がある．また 1to1 接続の場合には各ブレード同士を直接つなぐ完全結合と Ring 結合という選択肢がある．SBS では，スループット確保のために各ブレードが独立して転送を行えるようにバスやスイッチではなく 1to1 接続を選択した．その上で，ブレード間の転送におけるレイテンシを短くするため，どのブレードに対しても 1 回のブレード渡りで到達可能な 1to1 接続による完全結合を採用した．以降，ブレード間の接続を SMP リンクと呼ぶ．

【キャッシュ一貫性制御プロトコルの選択】

キャッシュ一貫性制御プロトコルとしては，ブロードキャストスヌープ方式とディレクトリ方式が考えられる．ブロードキャストスヌープ方式では要求元はスヌープを全ノードにブロードキャストし，キャッシュを持っているノードが要求元に直接応答する方式である．一方，ディレクトリ方式では要求元はまずメモリのあるノード（ホームノード）へとリクエストを送り，ホームノードはディレクトリを引いてキャッシュを持っている可能性のあるノードに対してスヌープを送る．

ディレクトリ方式では，キャッシュを持っている可能性のあるノードにしかスヌープを送らないため，スヌープの流量が最小限で済みスループットを圧迫しない点でブロードキャストスヌープ方式よりも有利である．一方，他ノードがキャッシュを持っていた場合，要求元→ホームノード→キャッシュノードの順にトランザクションを転送する必要があるため，要求元から直接キャッシュノードへのスヌープを転送するブロードキャストスヌープ方式の方がレイテンシの面では有利である．

SBS では，(1) データ・アドレス分離方式の採用によりスヌープがデータスループットを圧迫する心配がないこと，(2) ビジネスアプリケーションを用いた計測において他ノードのキャッシュにヒットする確率から計算した結果ブロードキャストスヌープ方式の方がレイテンシで有利だったこと，などからブロードキャストスヌープ方式を採用した．

【投機メモリリード】

投機メモリリードとは，実際にメモリ上のデータが最新かどうか確定する前に，メモリに対してリードを先行発行することである．メモリのアクセスには時間がかかるため，メモリリードが確定してから読み出したのではレイテンシが延びてしまう．そこで CPU から要求が出た時点で，スヌープ結果によらずに投機メモリリードをメモリに対して発行しておく．また，メモリリードしたデータは，スヌープ結果を待たずに要求元まで返してしまう．スヌープの結果，最新のデータが他 CPU のキャッシュ上にあった場合にはメモリリードの結果は無駄になるが，その分のトラフィックを犠牲にしてレイテンシを稼ぐ．

【スヌープフィルタ機能】

スヌープフィルタ機能は，そのノードの CPU がキャッシュしているラインの情報（キャッシュコピータグ）を SRAM 上に持ち，他ノードからのスヌープ要求に対して CPU に対するスヌープ要求を上げる必要があるかどうかを判断する機能である．コピータグを見て CPU がキャッシュしていないと分かった場合にはスヌープ要求を CPU まで上げずに済むため，レイテンシ短縮の効果がある．

2.3.2 スループット改善方式の検討

【SMP リンクに必要なスループット】

SBS では最大構成時（4 ノード時）に SMP リンクがボトルネックとならないように設計する．4 ノード時の SMP 構成で，メモリは各ノードに均等に搭載され，メモリアクセスに偏りはないと仮定する．この時，各プロセッサバスがフルスループットで動いたとすると，その内 1/4 は自ノードメモリに当たると考えられ，残りの 3/4 がそれぞれ他の 3 ノードへ行くと考えられる．従って，SMP リンク 1 本あたり，プロセッサバスの 1/4 のスループットがあればSMP リンクがボトルネックとはならない計算となる．

【SMP リンクのビット配分】

SMP リンク上のビットを，リクエストやスヌー

プなどのアドレストランザクションとデータトランザクションでどのように使うかについては，データ・アドレス混在方式と，データ・アドレス分離方式の2通りが考えられる．データ・アドレス混在方式の場合，SMP リンク全てをデータ転送に使えるため，データ転送時間を短縮でき，またリンクの使用効率も高くなる．しかしアドレストランザクションがデータ転送と競合して待たされることで，レイテンシが伸びるという問題点がある．方式シミュレーションの結果，データ・アドレス混在方式ではプロセッサバスのビジー率 25%程度で飽和してしまうという結果が得られたため，SBS ではデータ・アドレス分離方式を採用した．

3. SBS プロトタイプ機の設計

前節までの設計方針に従い SBS プロトタイプ機を設計する．SBS プロトタイプ機に搭載する CPU としては，Intel 社の Itanium2 プロセッサを選択した．

図 2 に SBS の 4 ブレード 8CPU SMP 構成を示す．各ブレードはプロセッサバス（FSB[3]）とノードコントローラ(NC)，2 つのメモリコントローラ（MC）から構成される．1 ブレードにつき最大 2 つの Itanium2 プロセッサを搭載可能である．なお，IO チャネル（PCI-Express）は図では省いた．

メモリは各メモリコントローラから 4 本ずつ DIMM[4]チャネルが出ており，1 ノードで合計 8 チャネル分，16 枚の DIMM スロットを持つ．DDR2 333 に対応し，各 DIMM チャネルの最大スループットは 5.3GB/s，NC—MC 間の最大スループットは 10.6GB/s となる．

さらに NC は SMP リンクのポートを 3 本持ち，NC 間を SMP リンクによって相互に接続することで，4 ノード 8CPU の ccNUMA 構成の SMP を実現する． 2.3 節で検討した通り，SMP リンクのデータ幅は片方向 1 本当たり FSB の 1/4，すなわち 667Mbps で 4B(32bit)分とした．この結果，全 SMP リンクのトータルスループットは 32GB/s となる．

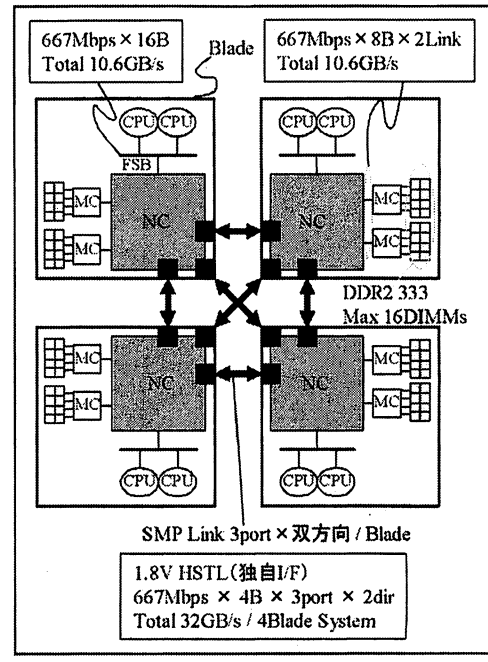

図 2. SMP リンクによる 4 ノード 8CPU SMP 構成

4. SBS プロトタイプ機の評価

本節では開発した SBS の試作機を使ってサーバモジュール間 SMP 機能の評価を行った結果を示す．また，評価に用いるベンチマークとしては，基本的な CPU の性能を測定するベンチマークである SPEC CPU2000[1]のうち，複数ジョブの実行によるスループットを測定する SPECint_rate と SPECfp_rate を使用した．

表 1 に計測時の詳細条件を示す．

表 1. SBS プロトタイプ機の評価条件

プロセッサ	Itanium2
プロセッサ数	2×ノード数
コア周波数	1.667GHz
FSB 周波数	667Mbps
L3 キャッシュ	9MB
主記憶	16GB×ノード数
OS	Red Hat Linux Advanced Server 3.0AS
コンパイラ	Intel® C++/Fortran Compiler for Linux 8.1

3 FSB=Front Side Bus
4 DIMM=Dual Inline Memory Module

4.1 スケーラビリティの評価結果

図 3 に 4 ノード時のスケーラビリティを商用 SMP サーバ機と比較した結果を示す．ここでスケーラビリティは次の式で定義する：

$$\text{スケーラビリティ} = \frac{\text{Nノード時の性能}}{\text{1ノード時の性能} \times N}$$

横軸に SPECint_rate のスケーラビリティ，縦軸に SPECfp_rate のスケーラビリティを取り，各サーバをプロットした．商用 SMP サーバ機のスケーラビリティは SPEC ホームページ[1]の登録データを基に導き出した．

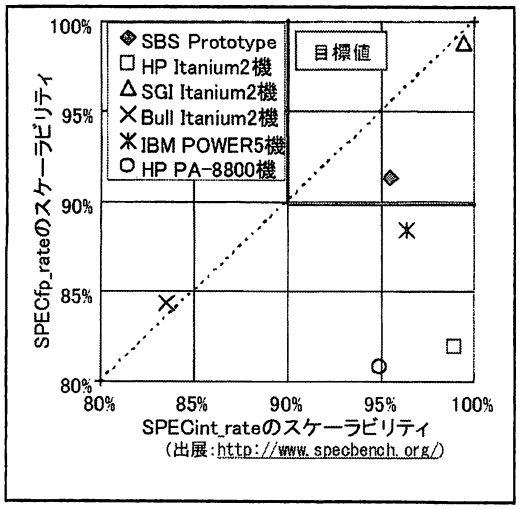

図 3. 商用 SMP サーバとのスケーラビリティ比較 (4 ノード構成時)

図 3 より SBS は，SPECint_rate/ SPECfp_rate 共に 4 ノード時に目標値であるスケーラビリティ 90%を達成した．また，商用 SMP サーバ機と比較しても，SGI 機を除いては同等かそれ以上のスケーラビリティを達成していることがわかる．

4.2 スケーラビリティに関する考察

図 4 にブレード数を 1 から 4 へと変化させた時の SBS の相対性能を示す．

この結果より，以下のことがわかる．SPECint_rate では，ノード数が 1 から 4 へと増えるにつれ，ほぼリニアに相対性能が向上する．一方の SPECfp_rate では，2,3 ノードに比べて 4 ノード時に相対性能値がより大きく上昇する下に凸なカーブを描く．以下ではこの理由について考察する．

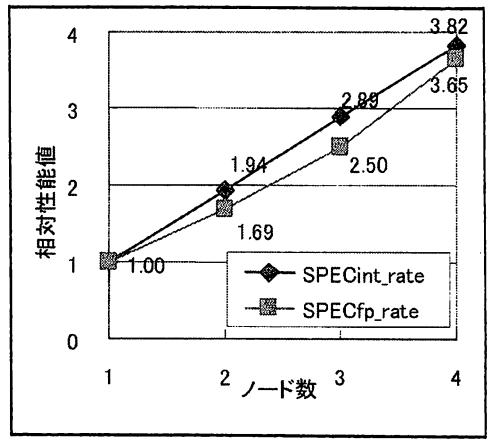

図 4. SBS の相対性能

SPECint_rate は，整数演算主体のプログラムから構成されており，それほど大きなデータセットを必要としないプログラムが多い．結果として，性能に与える影響はスループットよりもレイテンシの方が大きい．SBS のメモリ構成 (ccNUMA) では，リモートメモリはローカルメモリに比べてレイテンシが長いため，平均レイテンシはノード数が増えローカルヒット率が下がるに連れて増えていく．そのためレイテンシが支配的な SPECint_rate では，スケーラビリティも平均レイテンシの伸びに応じて減少する．

一方，SPECfp_rate は科学技術計算などの浮動小数点演算主体のプログラムから構成されており，巨大な配列や行列など大きなデータセットを扱うプログラムが多い．そのためレイテンシだけでなくスループットが性能に与える影響も大きい．

SBS では 4 ノード構成時のフルスループットを基にしてリンク幅を決定したため，2 ノード・3 ノード構成時には分割損が発生し，SMP リンクがボトルネックとなり得る．結果として SPECfp_rate の相対性能値は，2,3 ノード時に比べて 4 ノード構成時により大きく向上する下に凸なカーブを描く．これは 4 ノード構成時のスケールアップ性能を優先した結果であり，設計目標通りであることが確認できた．

5. 関連研究

ブレードサーバのような高密度サーバにおける性能評価の研究としては[2]などがある.

ブレードサーバ以外では, 複数サーバ間を結合して大きなSMPを構成できるサーバとして, IBM社のeServer xSeries[3]などがある.

図3より, SGIのItanium2搭載機はSPECint_rate/SPECfp_rate共にスケーラビリティ99%という飛び抜けた値を達成している. これは低レイテンシ・高スループットのNUMAlink[4]というインターコネクトを使用したハードウェア・アーキテクチャに加え, Linuxベースの独自OSであるSGI ProPack[5]により, ccNUMAの特性(メモリローカリティ)をうまく利用しているためと推測される. SBSにおいても, 実際のアプリケーションの運用に当たっては, OSやDBMS[5]のccNUMA対応機能を利用するなどの最適化オプションを利用することによって, より良いスケーラビリティを達成できる可能性がある[6].

6. まとめと今後の課題

6.1 まとめ

Web三階層システムを構成するサーバ群をのブレードサーバ上へサーバ統合するために, スケールアウトとスケールアップの両方に対応可能なブレード間SMP機能を搭載するスケーラブルブレードサーバ(SBS)アーキテクチャを提案した.

SBSのプロトタイプ機を用いた評価では, 性能に対してレイテンシが支配的なSPECint_rateとスループットが支配的なSPECfp_rateの両方のベンチマークにおいて, 目標である4ノードSMP構成時のスケーラビリティ90%を達成したことを確認した.

ブレード間SMP機能を搭載したSBSは, 低コスト・小さな増設単位というブレードサーバとしての要件と, 世界トップクラスのスケーラビリティという SMP サーバとしての要件の両立を果たした. これにより, スケールアウトにもスケールアップにも対応可能な柔軟なプラットフォームの提供を実現した.

6.2 今後の課題

今後はSBSプロトタイプ機を用いて, より実際のビジネスアプリケーションに近いベンチマーク(TPC-HやSAP SDベンチマークなど)を使った性能評価を行う予定である.

5 DBMS=Database Management System

参考文献

[1] Standard Performance Evaluation Corporation home page, http://www.specbench.org/.

[2] W. M. Felter et al., "On the performance and use of dense servers", IBM Journal Research & Development Vol.47 No.5/6, pp.671-688, September, November 2003.

[3] Mark T. Chapman, "Introducing IBM Enterprise X-Architecture Technology", IBM Whitepaper, August 2001.

[4] 日本SGI, "SGI® NUMAlink™ 業界をリードするインターコネクトテクノロジ", SGI ホワイトペーパー, 2005年.

[5] 日本SGI, SGI ProPack™ for Linux® Home Page, http://www.sgi.co.jp/solutions/linux/hpc/propack.html.

[6] Russel Clapp et al., "STiNG Revisited: Performance of Commercial Databases Benchmarks on a CC-NUMA Computer System", Proceedings of Workshop on Duplication, Deconstructing, and Debunking 2002, May 2002.