仮想環境における I/O 性能評価及び最適化手法提案

水野和彦[†1]　今田貴之[†1]

概要：大規模なデータを処理するビッグデータ解析では，大量のデータを処理するために高い I/O 性能が求められており，SSD 等のフラッシュメモリデバイスをベースとした Storage Class Memory（SCM）の導入が提唱されている．また，この SCM 向けデバイスには，NVMe（Non-Volatile Memory Express）という規格が提供されており，仮想環境に適用されつつある．仮想サーバ（VM）に NVMe デバイスを割り当てる場合，ハードウェア／ソフトウェアの構成等の要件に適した方法を選定することになる．しかし，仮想環境では，様々な組み合わせで環境を構築するため要件に合わせた対象構成を容易に選定することができない問題がある．そこで，本研究では，仮想環境に NVMe SSD を適用した実機環境による I/O 性能評価，および，この評価結果より仮想環境に NVMe SSD を適用した時の I/O 性能最適化手法を策定した．実機環境による I/O 性能評価では，Bare/KVM 環境の構成等により I/O 性能が Bare 環境に比べて 20%から同等の性能まで大きく変化することがわかった．また，I/O 性能最適化手法の検討では，評価構成の特徴整理，I/O 性能の関連要素の分析，および，I/O 性能を最適化するためのポイントを検討し，仮想環境の導入・運用時の I/O 性能を最適化する「コンフィグガイドライン」，および，実運用時の I/O 性能を最適化する「チューニングガイドライン」を策定した．このコンフィグガイドラインを用いれば，要件に適した仮想環境の構成を明らかにすることができ，かつ，その構成を利用した時の I/O 性能を把握することができる．また，チューニングガイドラインを用いれば，リソースの利用状況により I/O 性能が限界に達しているか明らかにすることができる．

キーワード：SSD，I/O 性能，仮想環境，KVM

I/O Performance Evaluation in the Virtual Environment and Optimization Method Suggestion

KAZUHIKO MIZUNO[†1]　TAKAYUKI IMADA[†1]

Abstract: As high I/O performance is required in big data analysis, storage class memories (SCM) such as solid state drives (SSD) are introduced for big data analysis. Recently, NVMe (Non-Volatile Memory Express) is standardized for SCM device. Especially in the virtual environment, NVMe is applied to improve I/O performance. To assign NVMe device to a virtual machine (VM), administrators should consider assignment according to requirements of hardware/software configuration. However, it is difficult to configure optimal virtual environment due to the enormous number of variations and combinations of configurations such as VM, physical/logical device, direct/indirect assignment, virtual device interface, assignment options, etc. This research evaluates I/O performance in several kinds of virtual environment with NVMe SSD, and proposes I/O performance optimization method for virtual environment with NVMe SSD. The results show that I/O performance in virtual environment varies from 20% to 100% of the performance on bare metal environment, by configurations and options. The proposed I/O performance optimization method is considered as guidelines for optimization on I/O performance in virtual environment. This report proposes "Configuration Guidelines" on optimization of the I/O performance at deployment of the virtual environment, and "Tuning Guidelines" for run-time tuning of the I/O performance. With "Configuration Guidelines", suitable virtual configuration and I/O performance on it could be estimated easily. And also with "Tuning Guidelines", resource utilization may clarify whether I/O performance reaches the limit or not.

Keywords: SSD, I/O Performance, Virtual Environment, KVM

1. はじめに

近年，企業システムが取り扱うデータ量は増大し続けており，IDC Digital Universe Study[1]によれば 2020 年には 44 ゼタバイトのデータが生成・複製されると試算されている．この巨大なデータ群はビッグデータと呼ばれ，データ間の関係性等を分析することで有益な情報が得られると期待されている．

ビッグデータの分析では，リアルタイムな解析が求められており，高速な分析を可能とするインメモリ DB が活用されている．日立では，高速インメモリ DB である SAP HANA®に特化した専用サーバ機として Hitachi Unified Compute Platform for SAP HANA（UCP for SAP HANA）®[2]等を出荷している．また，日立独自の仮想化技術である Virtage™[3][4][5]では，SAP HANA®の動作可能なハードウェア仮想化技術として認定を取得しており[6]，複数の SAP HANA®システムを Virtage™の仮想環境に統合できるため，システムの運用管理コストを削減させ，サービス水準の安定した高信頼なクラウドサービスを提供している．

このインメモリ DB では，仮想環境の導入や NAND フラッシュメモリよりも高速高性能な SCM（Storage Class Memory），および，NVMe SSD（Non-Volatile Memory Express Solid State Drive）[7]の適用が進められている．

†1 (株)日立製作所　研究開発グループ　情報通信イノベーションセンタ
Hitachi, Ltd., Research & Development Group, Center for Technology Innovation – Information and Telecommunications

以降では，最初に NVMe SSD を仮想環境に適用する際の問題点について報告し，次に問題点に対するアプローチとして実機環境による I/O 性能評価，および，I/O 性能評価結果より I/O 性能を最適化させるガイドラインとして策定した I/O 性能最適化手法について報告する．

2. NVMe SSD を適用した仮想環境の概要

2.1 NVMe SSD の概要

NVMe SSD では，高負荷時に発行キューと完了キューの構成となる 1 つのキューペアを CPU コア毎に割り当て，低負荷時には複数の発行キューを 1 つの完了キューに共有させる等のようにキュー数の調整を行える点や，コマンドを処理するキューが 1 つではなく複数(6 万 5536 個)になっているという点等が特徴となる[7]．これにより多数のディスク I/O 要求を同時に処理するサーバでは，大幅な高速化が実現される．この NVMe SSD は，Redhat®や Windows®等の各 OS，および，VMware®，Linux® KVM 等の仮想環境でサポートされている．

NVMe SSD のような高性能デバイスでは，高い I/O 性能を得るために複数の I/O 処理を並列に処理するマルチキューが必要となる．また，仮想環境の仮想化デバイスドライバにおいても同様にマルチキューの効果が高いと考えられる．例えば，Linux®で提供される仮想環境 KVM では，仮想化デバイスドライバである virtio-blk がマルチキューに対応している[8]．このマルチキューで解決できないデバイスネック等では，I/O 処理の対象となるブロックデバイスを配置する SSD を分散させる対策が効果的である．

2.2 NVMe SSD を適用した仮想環境の問題点

仮想環境で NVMe SSD 等を利用する場合には，利用要件に適した物理／仮想環境の構成，および，仮想環境の提供機能等を選定することになる．例えば，利用要件として I/O 性能を保証させたい場合には，NVMe SSD 等の物理デバイスを直接 VM に割り当てる，あるいは，仮想環境で提供される仮想化デバイスドライバの利用，および，I/O 性能に関するオプションを設定することになる．

このように仮想環境では，利用要件に適した環境構築が行われるが，利用要件に適した仮想環境を検討する場合，仮想環境の構成や機能の組み合わせが多種多様であるため対象構成を容易に選定することができない問題がある．また，仮想環境では，構成や機能により I/O 性能が大きく変動するため，NVMe SSD 等の物理デバイスを利用しても I/O 性能が向上しない可能性がある．

この問題を解決するためには，選定範囲を狭めるように指標を定めればよい．ここでの指標とは，利用要件を詳細化するためのガイドラインである．例えば，「I/O 性能保証」を利用要件と定めた場合，指標には，I/O 性能低下時のサポート方法等の運用方針，および，実機評価による I/O 性能評価結果に基づいて利用環境の選定等を定めることになる．

そこで，本研究では，上記指標の一つとして仮想環境での NVMe SSD の効果検証を目的に実機環境で I/O 性能評価を実施した．また，本評価結果に基づき NVMe SSD を適用した仮想環境の I/O 性能最適化手法を策定した．

以降では，まず，NVMe SSD の実機評価に関して説明し，次に I/O 性能最適化手法について説明する．

3. Bare/KVM 環境による NVMe SSD の基本性能評価

3.1 Bare/KVM 環境における性能評価方針

NVMe SSD の I/O 性能評価では，SSD に複数の処理が同時にアクセスした際の I/O 性能を実機環境により測定し，物理環境と仮想環境の稼働状態の分析，および，I/O 性能に影響を与える要因調査を目的とする．

本性能評価には，仮想環境として Linux®で提供される KVM を利用し，ディスク I/O のベンチマーク測定ツールである fio ツールを測定に利用する．なお，以降では，物理環境を Bare 環境，仮想環境を KVM 環境と称する．

表 1　仮想環境の主な構成要素

#	構成要素	選択範囲	評価構成
1	CPU	1～max	全ての CPU コア数を利用
2	メモリ量	1～max	VM 数に合わせて均等に割り当て
3	VM 数	1～max	最小：1，最大：CPU コア数
4	仮想化デバイスドライバ	virtio-blk	本評価に利用
		virtio-scsi	本評価で未使用
		virtio-nvme	本評価で未使用
5	ディスク	HDD	本評価では未使用
		SSD	SSD を 1～2 枚利用

表 2　仮想環境の主な構成要素

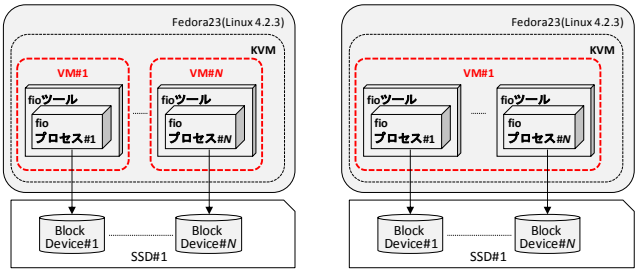
#	性能評価項目	内容
1	Bare/KVM の性能評価	SSD 利用時の I/O 性能検証 仮想化によるオーバヘッド検証 VM 構成による I/O 性能への影響調査
2	SSD の割り当て方による性能評価	SSD の割り当て方による I/O 性能への影響調査
3	オプションによる性能評価	KVM で提供されるオプションの効果検証

KVM 環境では，VM を構築する場合に CPU やメモリ等の物理サーバのリソース割り当て等を実施する．表 1 に KVM 環境の主な構成要素を示す．

本性能評価で評価対象とする KVM 環境には，最小 VM 数と最大 VM 数の構成を利用し，仮想化デバイスドライバに virtio-blk を割り当て，ディスクには SSD を 1～2 枚利用することとした．ここでの最小 VM 数とは，CPU を全て割り当てた 1 つの VM であり，最大 VM 数とは，CPU コアを各 VM に割り当てることを想定した CPU コア数分の VM である．

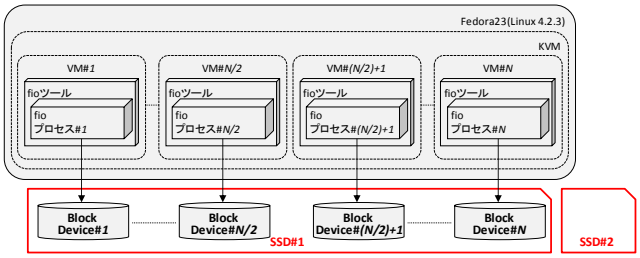
本評価構成の I/O 性能評価項目については，表 2 に I/O 性能評価項目，図 1，2 に評価構成を示す．

まずは，Bare/KVM 環境の基本的な I/O 性能評価として，実機環境で SSD を十分に利用できることを確認するために SSD の基本スペックと同等の I/O 性能を実機環境で得られるか検証する．また，仮想化によるオーバヘッドを確認するために Bare 環境と KVM 環境の I/O 性能を検証する．本性能評価の KVM 環境には，最大 VM 数の構成（1fio/vm ×N）（図 1-(a)），および，最小 VM 数の構成（N fio/vm）（図 1-(b)）を利用する．

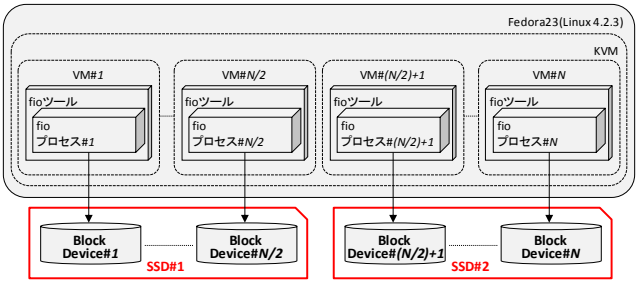


(a)別々の VM に集約（1fio/vm×N）　　(b) 1VM に集約 (N fio/vm)

図 1　Bare/KVM の性能評価構成概要



(a)1fio/vm×N で 1 枚の SSD に集約させたケース



(b)1fio/vm×N で 2 枚の SSD に分散させたケース

図 2　SSD の割り当て方変更による性能評価構成概要

次に，本性能評価では，ディスク構成による I/O 性能の影響を検証するために VM に割り当てる SSD を変更した時の性能評価を実施する．具体的には，fio ツールで負荷をかけるブロックデバイスの配置を 1 つの SSD に集約したケース（図 2-(a)），および，2 つの SSD に分散させたケース（図 2-(b)）で I/O 性能の評価を実施する．

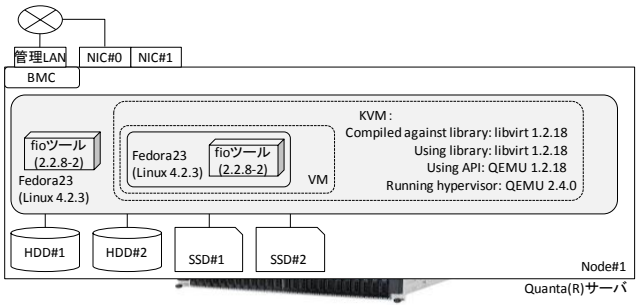
最後に，仮想環境のオプションによる I/O 性能への影響を評価する．KVM では，VM に virtio-blk を割り当てた場合，I/O スレッドを複数割り当てるオプション（iothreads）を設定することが可能であり，このオプションによる I/O 性能の向上が期待できる．

3.2　NVMe SSD 性能評価向け実機環境の概要

図 3 に NVMe SSD の I/O 性能評価を行う実機環境の概要を示す．実機環境には，Quanta®社の QuantaPlex T41SP-2U を物理サーバとして利用する．本物理サーバには，SSD が 2 台搭載されており，ページサイズを 4kB としてランダム I/O の負荷をかけた場合，Read 性能として 450kIOPS，Write 性能として 75kIOPS の I/O 性能を得ることができる[9]．

この物理サーバには，Fedora23（Linux 4.2.3）を準備し，仮想環境として Linux で提供される KVM を利用する．Linux では，ディスク I/O のベンチマーク測定ツールである fio ツール，および，Linux では CPU の稼働情報を取得する mpstat や I/O に関する稼働情報を取得する iostat 等が提供されている．これらツールにより I/O 性能評価時の稼働情報を自動的に集計できるように実機環境を構築した．

I/O 性能評価では，fio プロセスがかける負荷としてページサイズに 4 k B，SSD へのアクセス方式にランダム I/O，オプションにバッファを利用しないで直接 SSD に I/O をかけるオプション（O_DIRECT）を設定した．また，本性能評価では，入力負荷に対する I/O 性能と稼働状況の変化を分析するが，ここでの入力負荷とは，fio プロセスが個々にかける負荷ではなく，SSD に対してかける負荷（各 fio プロセスの負荷の総和）を表す．



サーバ	仕様
Quanta サーバ	QuantaPlex T41SP-2U CPU:Intel(R) Xeon(R) E5-2640 v3(Haswell) @2.6GHz ×2sockets CPU core: 8 x2sockets メモリ : DDR-4 RAM – DDR4 normal Voltage(1.2V) メモリ量 : 8192[MB]@1.866GHz ×8 slots NIC : Intel(R) X540-AT2 10GbE ×2port HDD : SATA 2.5" 7.2Krpm 500GB（Intel C610） SSD : NVMe SSD 2.5" 400GB（Intel DC P3700）

図 3　実機環境概要

4.　NVMe SSD を適用した仮想環境の I/O 性能評価結果

4.1　Bare/KVM の I/O 性能評価結果

図 4 に Bare/KVM 環境における I/O 性能の評価結果を示

す．図 4 では，fio ツールの入力負荷に対して SSD で処理された IOPS の結果が示されており，左側のグラフには参照のみ（Read 比率：100%）の場合，右側のグラフには更新のみ（Write 比率：100%）の結果を示しおり，横軸に fio ツールでかける入力負荷を，縦軸に SSD で処理された IOPS の結果を示している．

Bare 環境の I/O 性能は，入力負荷が参照のみの場合に約 438kIOPS，入力負荷が更新のみの場合に約 90kIOPS であり，SSD の基本スペックと同等の性能が得られることがわかった．

KVM 環境の I/O 性能は，1fio/vm×16 のケースの場合，入力負荷が参照のみの場合に Bare 環境に比べて約 0.79 倍であり，入力負荷が更新のみの場合に Bare 環境と同等となった．また，16fio/vm のケースの場合，入力負荷が参照のみの場合に Bare 環境に比べて約 0.21 倍であり，入力負荷が更新のみの場合に Bare 環境に比べて約 0.93 倍となった．

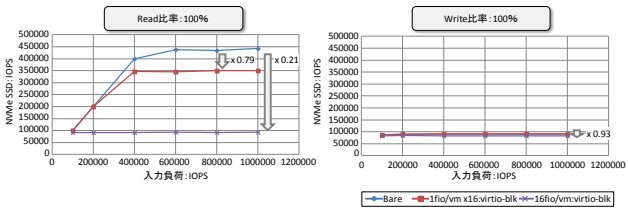


図 4　Bare/KVM 環境の I/O 性能結果

図 5 には，KVM 環境で SSD のブロックデバイスを直接 VM に割り当てた時の I/O 性能の評価結果を示す．図 5 のグラフの縦軸と横軸は，図 4 のグラフと同様である．

16fio/vm のケースで VM にブロックデバイスを直接割り当てた場合（Path Through 設定を実施した場合）には，Bare 環境での I/O 性能と同等の結果となった．従って，図 4 における 16fio/vm の性能低下は，virtio-blk の処理によるオーバヘッドと考えられる．なお，SSD のブロックデバイスを直接 VM に割り当てた場合には，ハイパバイザとなるホストから SSD が認識できなくなる．そのため，この SSD を他の VM に割り当てることができず，1fio/vm×16 のようなケースでは，Path Through 設定による I/O 性能を評価することができない．

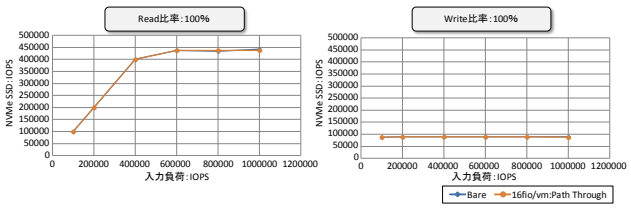


図 5　KVM 環境（Path Through 設定）の I/O 性能結果

図 6 には，Bare 環境の I/O 性能評価時に取得したデバイスビジー率，および，入力負荷を参照のみとした場合の CPU 利用率の結果を示す．なお，図 6 の左側のグラフには，横軸に fio ツールでかける入力負荷を，縦軸にデバイスビジー率の結果を示しており，右側のグラフには，横軸に fio ツールでかける入力負荷を，縦軸に CPU 利用率の結果を示している．

入力負荷が更新のみの場合には，入力負荷によらず常にデバイスビジー率が 100%であり，入力負荷が参照のみの場合には，入力負荷が 600kIOPS 以上でデバイスビジー率が 100%となり SSD のデバイスネックが発生したと考えられる．

また，入力負荷が参照のみの場合にデバイスネックが発生するタイミングでは，CPU 利用率がアプリケーションで利用される CPU 利用率（%usr）よりもカーネルで利用される CPU 利用率（%sys）が高くなることがわかった．これは，fio ツールの処理に対してサーバ側の処理が間に合っていないため，I/O 処理が停滞し I/O 性能の上限に達したと考えられる．

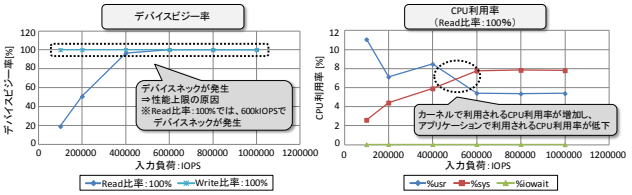


図 6　Bare 環境での性能ネック解析

4.2 SSD の割り当て方変更による性能評価結果

前節で示したように実機環境の性能上限は，デバイスネックが原因と考えられる．そこで，fio ツールで負荷をかけるブロックデイバスを 2 つの SSD に分散させることでデバイスネックを緩和させた状態で I/O 性能を測定した．図 7 には，1fio/vm×16 のケースにおいてブロックデバイスを配置する SSD を分散させた時の I/O 性能の評価結果を示す．なお，図 7 のグラフの縦軸と横軸は図 4 のグラフと同様である．

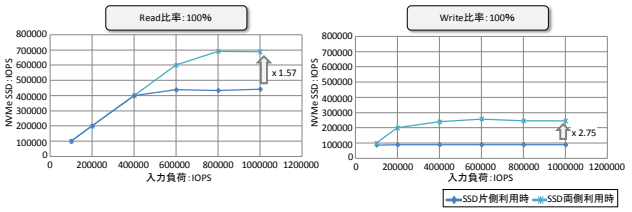


図 7　SSD の割り当て方変更による I/O 性能結果

ブロックデバイスを配置する SSD を分散させた場合には，入力負荷が参照のみの場合に約 689kIOPS であり，入力負荷が更新のみの場合に約 247kIOPS となった．SSD を分散させる前の構成（図 4 の I/O 性能評価結果）と比較し

た場合には，入力負荷が参照のみの場合に約 1.57 倍，入力負荷が更新のみの場合に約 2.75 倍と I/O 性能が向上する結果となった．

4.3 KVM 環境のオプション設定による性能評価結果

KVM の virtio-blk では，マルチキュー対応として I/O スレッドを複数割り当てるオプション（iothread オプション）が提供されている．そこで，このオプションによる効果を 16fio/vm のケースで評価した．図 8 にオプション利用時における I/O 性能の評価結果を示す．図 8 のグラフの縦軸と横軸は，図 4 のグラフと同様である．

オプション設定により 16fio/vm のケースでは，入力負荷が参照のみの場合に約 342kIOPS となり，入力負荷が更新のみの場合に約 89kIOPS となった．この I/O 性能は，1fio/vm ×16 のケースと同等でありオプションによる改善効果が高いことがわかった．

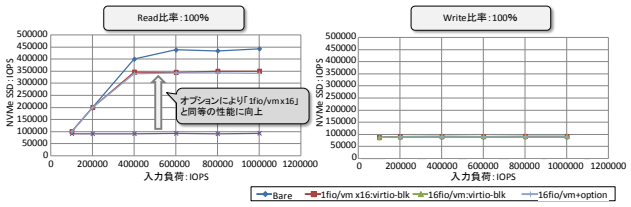


図 8　マルチキュー対応オプションによる I/O 性能結果

5. NVMe SSD を適用した仮想環境の I/O 性能最適化手法の検討

5.1 I/O 性能評価結果の考察

本節では，I/O 性能を最適化させる I/O 性能最適化手法の事前検討として性能評価の考察内容について説明する．

（1）評価構成の特徴整理

KVM 環境の I/O 性能評価では，主に 4 つの構成を利用した．表 3 に I/O 性能評価時の評価構成と特徴について示す．

構成 1 は，「1fio/vm×N」のケースであり，VM 数を増減させることで容易に入力負荷を変更できるが，メンテナンス時等では，全 VM に対して管理を行うため管理コストが大きくなる．

構成 2 は，「N fio/VM」のケースであり，メンテナンス等を VM 単体に対して行えば良いため管理コストが小さくなるが，Bare 環境に比べて I/O 性能が非常に低く，かつ，fio ツール等を増設する場合に VM の再設定が必要となる．

構成 3 は，「N fio/vm＋Path Through」のケースであり，構成 2 と同様に管理コストは小さく，かつ，SSD を直接 VM に割り当てるため，Bare 環境と同等の I/O 性能を得ることができる．しかし，他の VM で SSD を利用することができず，かつ，SSD のデバイス設定を VM 側で行うことになる．

構成 4 は，「N fio/vm＋iothread」のケースであり，構成 2，3 と同様に管理コストは小さく，また，構成 1 と同等の I/O 性能を得ることができる．しかし，VM に割り当てて全デバイスに設定が必要となり，また，virtio-blk 特有のオプションとなるため高度な SCSI 機能を利用する virtio-scsi 等の他の仮想化デバイスドライバ利用する場合に適用できない．

表 3　I/O 性能評価構成

No	評価構成	特徴	
		メリット	デメリット
1	1fio/VM×N	・VM 数の増減で I/O 性能を制御可能 ・VM 数の増減は仮想環境で容易に対応可	・メンテナンス時等で全 VM の管理要（管理コスト大）
2	N fio/VM	・メンテナンス時等の管理コスト小 ・仮想環境への移行が容易	・I/O 性能が非常に低い ・fio 増設時等で VM の再設定要
3	N fio/VM + Path Through	・メンテナンス時等の管理コスト小 ・仮想化のオーバヘッド小	・fio 増設時等で VM の再設定要 ・他の VM で SDD を利用不可 ・VM 側で SSD 設定要
4	N fio/VM + Iothread	・メンテナンス時等の管理コスト小 ・仮想化のオーバヘッドは構成 1 と同等	・fio 増設時等で VM の再設定要 ・全ブロックデバイスに対して設定要 ・virtio-blk 以外に利用不可

（2）I/O 性能の関連要素の分析

実機環境による性能評価では，様々な影響を受けて I/O 性能が変動していたことから I/O 性能に関連する要素を 3 つ選定した．表 4 に I/O 性能の関連要素を示す．

関連要素 1 には，Bare 環境の I/O 性能評価でブロックデバイスの配置変更により I/O 性能の上限が向上したことから「利用する SSD 数」を選定した．

関連要素 2 には，I/O 性能の上限時にデバイスビジー率と CPU 利用率の挙動が変化することから「物理サーバのリソース利用状況」を選定した．

関連要素 3 には，入力負荷が更新時と参照時で I/O 性能の変化が異なり，かつ，入力負荷が参照時に I/O 性能が高くなることから「入力負荷の参照・更新比率」を選定した．

表 4　I/O 性能の関連要素

No	要素	内容	特徴
1	SSD 数	・空き SSD の利用（ブロックデバイスの配置を分散）	・デバイスビジー率の緩和による I/O 性能向上
2	リソース利用状況	・CPU 利用率の統計情報	・I/O 性能の上限予兆
3	Read/Write 比率	・入力負荷の参照と更新の割合	・入力負荷で参照が多い場合に I/O 性能向上

（3）I/O 性能を最適化するためのポイント

仮想環境で I/O 性能を向上させる場合には，アプリケーションを実行する VM 数を増やすこと，および，実機環境を構成するハードウェア/ソフトウェア等のスペック向上

が考えられる．ここでは，前者をスケールアウトと称し，後者をスケールアップと称する．

仮想環境の導入・運用時では，このスケールアウトとスケールアップの利用方法により仮想環境の構成が異なる（VM の管理方法が異なる）ため，I/O 性能を構成により最適化する 1 つのポイントになると考えられる．

スケールアウトで I/O 性能を向上させる場合には，VM を容易に増設できることがポイントとなるため，「1fio/vm ×N」のケースが適している．他の評価構成では，VM 上に複数のアプリケーションを集約した構成となるため適していない．

スケールアップで I/O 性能を向上させる場合には，実機環境を構成するハードウェアやソフトウェアの変更時に VM を容易に稼働・停止できることがポイントになるため，「N fio/vm」のケースが適している．但し，「N fio/vm」のケースでは，I/O 性能が極端に低いため，仮想環境の提供機能等を利用することで I/O 性能を向上させることになる．

一方，実運用時には，実機環境のリソースの利用状況が I/O 性能を最適化させるポイントになると考えられる．例えば，性能評価では，デバイスビジー率と CPU 利用率の挙動により I/O 性能の上限を判別することが可能である．また，ブロックデバイスを配置する SSD に余裕があれば，複数の SSD にブロックデバイスを分散させることで I/O 性能の向上が期待できる．

5.2 I/O 能最適化手法の策定と考察

I/O 性能最適化手法では，コンフィグガイドラインとチューニングガイドラインを策定した．本節では，各ガイドラインについて説明する．

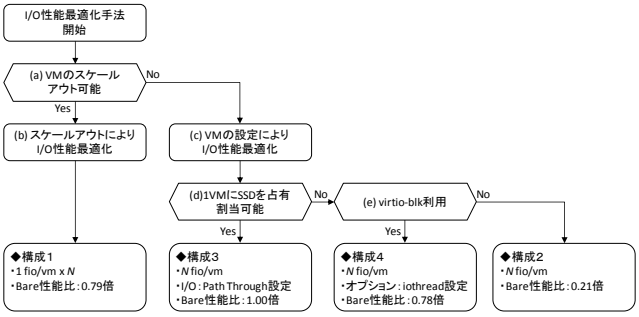


図 9　コンフィグガイドラインの概要

コンフィグガイドラインでは，仮想環境の導入・変更時における最適な仮想環境の構成を明らかにする．図 9 にコンフィグガイドラインの概要を示す．

まずは，選定する仮想環境構成を VM のスケールアウトの利用有無で 2 つに分ける（図 9-(a)）．スケールアウトを行う場合には，VM の増設が容易に行える「1fio/vm×N」を選定する．

スケールアウトを行わない場合には，VM の設定により I/O 性能を最適化する（図 9-(c)）．具体的には，SSD を VM に占有させることが可能であれば，「N fio/vm＋Path Through」を選定する（図 9-(d)）．また，SSD を他の VM に利用する場合には，「N fio/vm＋iothread」を選定する（図 9-(e)）．但し，iothread オプションは，VM に virtio-blk を利用している時に設定可能であり，virtio-blk を利用しない場合には，「N fio/vm」が選定されることになる．

以上のようにコンフィグガイドラインを用いれば，利用要件に適した仮想環境の構成を明らかにすることができ，かつ，その構成を利用した時の I/O 性能を把握することができる．また，本ガイドラインでは，環境構築時に適用することで最適な仮想環境構成を選定することができるため，スタートアップ工数の削減が期待できる．実運用中であれば，本手法で選定した仮想環境構成により環境変更時の I/O 性能の変化量を把握することができる．

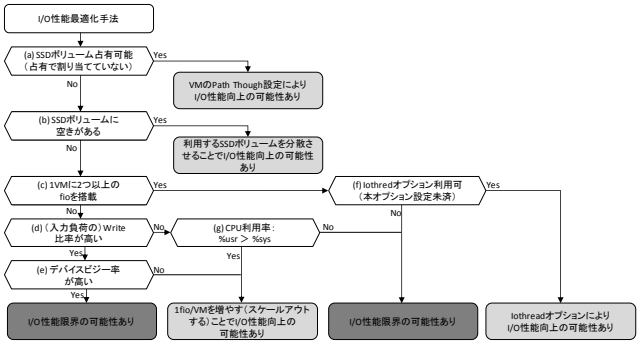


図 10　チューニングガイドラインの概要

チューニングガイドラインでは，実運用時における I/O 性能の上限を明らかにする．図 10 にチューニングガイドラインの概要を示す．

まずは，VM に直接 SSD を割り当てていない場合に SSD を占有可能か確認する（図 10-(a)）．SSD を占有できれば，Bare 環境と同等の I/O 性能が得られるため，I/O 性能の向上が期待できる．

次に，SSD の空き状況を確認する（図 10-(b)）．SSD に空きがある場合には，ブロックデバイスの配置先となる SSD を分散できるので I/O 性能の向上が期待できる．

次に，VM 上で稼働するアプリケーション（fio ツール）数を確認する（図 10-(c)）．VM 上で複数の fio ツールを稼働させている場合には，iothread オプションの利用有無を確認する（図 10-(f)）．この時に iothread オプションを利用していれば I/O 性能限界と考えられ，本オプションを利用していなければ I/O 性能の向上が期待できる．

VM 上で複数の fio ツールを稼働させていない場合には，入力負荷の参照・更新の割合を確認する（図 10-(d)）．入力負荷の更新比率が高い場合には，入力負荷の IOPS によらずデバイスビジー率が高くなっているため，アプリケーションで利用される CPU 利用率（%usr）とカーネルで利用さ

れる CPU 利用率（%sys）を確認する（図 10-(g)）．この%usr が高ければ VM をスケールアウトさせることで I/O 性能の向上が期待できるが，%sys が高ければ I/O 性能限界と考えられる．

入力負荷の参照比率が高い場合には，デバイスビジー率を確認する（図 10-(e)）．このデバイスビジー率が高い場合には，デバイスネックが発生しているため I/O 性能限界と考えられるが，デバイスビジー率が低い場合には，VM をスケールアウトさせることで I/O 性能の向上が期待できる．

以上のようにチューニングガイドラインを用いれば，リソースの利用状況により I/O 性能限界を容易に明らかにすることができる．また，本手法では，各リソースの利用状況から I/O 性能限界を判定しているため，不要なリソース（余剰リソース）も判断することができコストパフォーマンスの削減効果も期待することができる．

6. 関連研究

仮想環境に NVMe SSD を利用する際の I/O 性能に関する研究として，Kim らは，virtio-blk の data-plane 機能[10]をベースとして QEMU のドライバに CPU の Affinity 設定，ゲスト（VM）の IPI（Inter-process interrupt）処理の簡素化，I/O 要求のダイナミック調整等の機能実装，および，論理 CPU に I/O スレッドと queue を割り当てる（vCPU-dedicated I/O thread）機能を実装することで仮想環境の I/O 性能が向上することが報告されている[11]．Oh らは，I/O スタックを最適化するために考案した Pipelined polling 機能により複数の I/O 処理を並列に処理させることでロック競合の解消や割り込み処理による遅延を緩和し I/O 性能の向上や CPU 利用率の改善等が報告されている[12]．Oikawa らは，NVMe SSD 等ストレージを仮想化する VMMS（Virtual Main Memory Storage）により OS の軽量化を図り I/O 性能を向上させる手法[13][14]等が報告されている．

本研究では，実機検証による I/O 性能最適化手法を提案しており，上述した研究内容により NVMe SSD の I/O 性能の向上が期待できるため，本手法の精度向上・利用用途の拡充が期待できると考えている．

7. おわりに

本研究では，仮想環境で提供できる構成や機能の組み合わせが多種多様であるため対象構成を容易に選定することができない問題に対して，選定範囲を狭める指標を定めるために実機環境による I/O 性能評価，および，I/O 性能最適化手法を策定した．

実機環境による I/O 性能評価では，利用する Bare/KVM 環境の構成やオプションの設定により I/O 性能が大きく変化することがわかった．例えば，単純に 1 つの VM に集約させた場合には，I/O 性能は Bare 環境に比べて 20%程度となるが，ブロックデバイスを直接 VM に割り当てることで Bare 環境と同等の I/O 性能を得ることができる．

この実機評価に基づいて，仮想環境の導入・運用時に I/O 性能を最適化させるコンフィグガイドライン，および，実運用時に得られる I/O 性能を最適化させるチューニングガイドラインを I/O 性能最適化手法として策定した．

コンフィグガイドラインでは，利用要件に適した仮想環境の構成を明らかにすることができ，かつ，その構成を利用した時の I/O 性能を把握することができる．また，チューニングガイドラインを用いれば，リソースの利用状況により I/O 性能が限界に達しているか容易に明らかにすることができる．

今後の課題としては，実機評価の条件をより詳細化することで仮想環境の構成や挙動をより明確化させて本手法の精度を向上させる必要がある．また，I/O 性能最適化手法は，KVM 環境の性能評価により策定しているため，他の仮想化基盤でも同様に効果を得られるか検証する必要がある．

参考文献

[1] Digital Universe，“The DIGITAL UNIVERSE of OPPORTUNITIES: RICH DATA & the Increasing Value of the INTERNET OF THINGS,”2014 年.
[2] 日立製作所，“Hitachi Unified Compute Platform for SAP HANA(UCP for SAP HANA),” http://www.hitachi.co.jp/products/it/unified/products/model/saphana/.
[3] 上野仁他，“情報システムの運用効率を向上する「BladeSymphony」のサーバ仮想化機構「Virtage」,” http://www.hitachihyoron.com/jp/pdf/2007/07/2007_07_10.pdf，日立評論，7 月号，Vol.89 No.07 562-567(2007) ．
[4] 日立製作所，“サーバ仮想化機構「Virtage」ハードウェア透過性,” http://www.hitachi.co.jp/products/bladesymphony/virtual/dl/virtage_wp03.pdf，ホワイトペーパー.
[5] 上野仁，長谷川里美，“Virtage: Hitachi’s Virtualization Technology,”4th Workshop on Virtualization and High-Performance Cloud Computing-VHPC’09，2009 年 08 月 25 日.
[6] 日立製作所，“日立のサーバ論理分割機構 Virtage が SAP HANA®の動作可能な仮想化技術として SAP 社から認定,” http://www.hitachi.co.jp/New/cnews/month/2014/10/1021b.html，ニュースリリース，2014 年 10 月 21 日.
[7] SNIA，“PCIe SSD 101 標準，マーケット，性能の概要,” http://www.snia-j.org/tech/WH/PCIe_SSD/files/PCIe_SSD_101_J.pdf，2013 年.
[8] Ming Lei，“Virtio-blk Multi-queue Conversion and QEMU Optimization,” http://www.linux-kvm.org/images/6/63/02x06a-VirtioBlk.pdf，2014 年.
[9] Intel，“Intel Solid State Driver DC P3700 Series,” http://www.intel.cn/content/dam/www/public/us/en/documents/product-specifications/ssd-dc-s3700-spec.pdf，Product Specification，2015 年 10 月.
[10] IBM，Suse，“KVM Virtualized I/O Performance,” IEEE Computer Society，pp. 1-11， (2015).

https://www.suse.com/docrep/documents/xvbozdzzxj/kvm_virtualized_io_performance.pdf, 2013 年 6 月.
[11] Kim, T. Y.; Kang, D.; Lee, D. & Eom, Y. I., “Improving performance by bridging the semantic gap between multi-queue SSD and I/O virtualization framework.,” in 'MSST', IEEE Computer Society, pp. 1-11, (2015).
[12] M. Oh *et al.*, “Enhancing the I/O system for virtual machines using high performance SSDs, ” in *Proc. IEEE int. Performance Computing and Commun. Conf.*, Austin, TX, pp. 1-8,(2014).
[13] 追川修一, “ゲスト OS 軽量化のためのストレージ仮想化手法,” 情報処理学会, Vol.8 No.1 1-11 , 2015 年 3 月.
http://www.hitachi.co.jp/New/cnews/month/2014/10/1021b.html, ニュースリリース, 2014 年 10 月 21 日.
[14] Oikawa, “Virtual Storage as Memory for High Performance Storage Access, ” in *Proc. IEEE int. computer society*, (2014).