論　文

アプリケーション処理のI/O挙動特性を利用したディスクの実行時省電力手法とその評価: オンライントランザクション処理における省電力効果

西川　記史[†]　　中野美由紀[†]　　喜連川　優[†]

Runtime Disk Energy Saving Method using Application I/O Behavior and its Evaluation: Energy Saving Efficiency for Online Transaction Processing

Norifumi NISHIKAWA[†], Miyuki NAKANO[†], and Masaru KITSUREGAWA[†]

あらまし　デジタルデータの急増に伴い，ディスクの使用容量が著しく増加している．これにより IT 機器の消費電力におけるディスク消費電力の比率が急増している．本論文は，代表的な大量データ処理アプリケーションである DBMS，中でも OLTP 処理におけるディスクの実行時省電力手法を提案する．一般に OLTP ではデータアクセスが頻繁なため，OLTP 実行中のディスクの省電力化は困難とされている．OLTP の I/O 挙動特性を詳細に解析し，実行中の OLTP においてもディスクの省電力化の可能性があることを明らかにする．次に，得られた I/O 挙動特性を基に新たなディスクの省電力手法を提案する．実機を用いた実験により，提案手法が OLTP 実行中にも関わらず従来の手法と比較して省電力化できることを示す．

キーワード　ディスク，省電力，OLTP，データ配置制御，Write 遅延

1. はじめに

人類が生み出すデジタルデータの増加に伴い，ディスクの使用容量が急増している．ディスクの使用容量は年率 10%以上のペースで増加しており，2014 年にはディスクの使用容量は 2009 年の 9 倍以上に達すると予測されている [1].

現在，データセンタにおける IT 機器の消費電力はサーバが 60%，ネットワークが 10%，ディスクが 30%を占めている [2]．データセンタではサーバや OS，アプリケーションの仮想化によるサーバの省電力化が進展しているが，前述のようにデータそのものは急増しておりデータを格納するストレージの消費電力の比率はますます高まると指摘されている [3].

現在出荷されているディスクのほとんどは，ヘッドの退避やプラッタの回転停止等，消費電力を削減する機能を搭載している．しかし，CPU の省電力機能である Dynamic Voltage and Frequency Scaling と異なり，省電力状態中のディスクは I/O 処理ができない．またディスクを省電力状態から I/O 処理可能な状態に復帰させるために数秒以上のオーバーヘッドを要する．アプリケーション実行中に，その処理性能を低下させずにディスクの省電力機能を適用することは，CPU の省電力機能の適用と比較して非常に困難である．一方，前述の通りデータセンタで使用される IT 機器中のディスクの消費電力は今後急増すると予測されており，その消費電力の削減は急務である．

大量のデータを処理する代表的なアプリケーションとしてデータベース管理システム (DBMS) がある．ディスクの DBMS 向け出荷容量はディスクの全出荷容量の 6 割以上を占め，さらにその半数以上がバンキングや証券取引，ERP や CRM などの Business Processing と呼ばれるオンライントランザクション処理 (OLTP) に使用されている [4]．OLTP 処理中のディスクの消費電力は，OLTP に使用される IT 機器の全消費電力の 60%以上との報告もあり [5]，OLTP におけるディスクの消費電力の削減は，IT 機器全体

[†] 東京大学生産技術研究所，東京都
Institute of Industrial Science, the University of Tokyo, Komaba 4–6–1, Meguro-ku, Tokyo, 153–8505 Japan

の消費電力を削減する上で重要である．

従来から，ディスクの省電力化を目的とした研究が多数報告されている．その多くは，ディスクに対する I/O をデバイスあるいはファイルレベルで監視し，I/O 発行の制御やディスクブロック又はファイルの配置制御を行い，ディスクの省電力化の機会を作り出す手法である [6]〜[14]．また，アプリケーション情報から I/O 発行のスケジューリング等を実行前に定めるディスクの省電力手法等が提案されている [15], [16]．しかし，これらの手法は OLTP 実行中の省電力化には適さない．

多量のディスクを利用する OLTP は，高スループットで動作する場合はディスクに対して毎秒数十回ものランダム I/O を発行する．従来のデバイスレベル (システムコール) のディスクアクセス情報のみを用いた省電力手法は，I/O 発行間隔が数十秒から数分と長い場合や，アプリケーションコードの解析により実行前に I/O アクセス挙動が予測できる場合を対象としている．あるいは，現在商用化されていない回転数を変更可能なディスクが対象となる．このため高スループットで動作している OLTP に対してデバイスレベルのアクセス情報等を用いた OLTP におけるディスクの省電力化は困難である．

また，DBMS の知識を用いたディスクの省電力化の研究 [15], [16] では，システム構成や設定の違いによる消費電力量の差異など静的なシステムパラメタに着目しており，アプリケーション実行中の動的な省電力制御は考慮していない．

我々は既に複数のディスクを用いた OLTP の省電力化の可能性について予備的な評価を行ってきた [17], [18]．本論文では，実際に OLTP を実行し，取得した I/O トレースを解析し OLTP の I/O 挙動特性を明らかにする．そして OLTP においてもディスクの省電力機能を利用する機会がある可能性を示す．次に，I/O 挙動の解析結果に基づき，アプリケーション実行中にもディスクの省電力化が可能となるディスク省電力手法を提案する．我々が提案する手法の特長は，アプリケーションレベルでの論理的データ単位 (表・索引等) でディスクの省電力化を考える，つまり表・索引毎の I/O 発行間隔をモニタリングし，省電力化可能なアクセスパターンを持つ論理的データ単位を抽出し，その配置を決定する．さらに，アプリケーションレベルでの論理的データ単位のアクセス傾向が write が支配的であるデータに着目し，DB への write がログ先行書き出しプロトコル (WAL) に従えばよいことを活用した write 遅延を試みる．実機および実 I/O トレースを用いて提案手法の評価を行い，提案手法が既存のブロックレベルおよびファイルレベルの省電力手法 [11], [14] と比較して省電力効果が大きく得られることを示す．

以下，2 章では関連研究について，3 章ではディスクの消費電力特性について述べる．4 章では複数台のディスクを用いた小規模 TPC-C の I/O 挙動特性について述べ，5 章で TPC-C の I/O 挙動特性を用いたディスクの省電力化について述べる．6 章で実験結果を示し，7 章で考察を行う．そして 8 章でまとめる．

2. 関 連 研 究

ストレージやディスクの省電力手法には，デバイスレベルでのアクセス頻度等を用いた省電力手法と，DBMS などのアプリケーションレベルの省電力化手法がある．

2.1 ディスク省電力化手法

2.1.1 I/O 発行間隔制御

ディスクの省電力機能を使用する機会を増やすために，ディスクへの I/O の発行間隔を制御する手法が提案されている [6]〜[9]．

これらの手法は，物理ブロック単位アクセスにおいて RAID コントローラのキャッシュあるいは OS 主記憶などを用いてディスクのアイドル時間を伸ばす．つまり read であれば先読みを行いキャッシュにデータをロードし，write であればキャッシュに write されたデータをまとめて一括 write する．この結果物理ブロック単位の I/O 発行間隔を伸ばし，ディスク省電力機能を適用する機会が生じる．

しかし，OLTP はランダム read が主となるため先読みによりデータをキャッシュにロードしておくことは困難である．また，OLTP は小規模なものでもディスクに対して毎秒数回 read を発行する．従って write の発行間隔のみを伸ばす write の一括化は効果がなく，ディスク省電力機能を適用できるだけの I/O 発行間隔を生成することは困難である．

2.1.2 データ配置制御

ブロックやファイルの I/O アクセス頻度に基づきディスクへのデータの配置を制御することにより，ディスクの省電力機能を適用できるだけの I/O 発行間隔を生成する手法が提案されている．本手法は，I/O 頻度の高いデータを同じディスクに集中させ，I/O 頻度が

低いディスクに対してディスク省電力機能を適用することによりディスクの省電力化を図る [7], [10]〜[14].

いずれの手法もディスク内部で取得可能な物理ブロック毎の I/O 数，あるいは OS で取得可能なファイル毎の I/O 数を用いてデータの配置制御を行う．OLTP は頻繁にデータの追加・削除が行われるため，I/O が行われるブロックアドレスは動的に変化する．従ってブロック単位のアクセス頻度のみの情報では，I/O 頻度が高いデータを同一ディスク上に保持しておくことが困難である．

ファイル毎の I/O はブロック毎の I/O と異なり変化が激しくないと考えられるが，従来のファイル単位のデータ配置手法では，DBMS のファイル配置にはログと DB 用データのディスクを分ける必要があるなどの物理設計上の注意事項は考慮されない．このためファイルの情報のみを用いる省電力手法を OTLP に適用した場合，大幅な性能低下が生じる可能性がある．

2.2 DBMS によるストレージ省電力手法

Harizopoulos らは，従来のハードウェアのみに閉じた省電力手法はソリューションの一部であり，データ管理ソフトウェアが大規模なデータセンタの省電力化に重要な役割を果たす可能性があると述べている [15]. 具体的には，ハードウェアのみに閉じた省電力手法による高性能アルゴリズムやハードウェア構成では高性能アルゴリズムやハードウェア構成がエネルギー効率の観点からは最適ではない例を挙げ，DBMS においてもエネルギー効率を意識した DB チューニングやリソースの集約を考慮する必要があると指摘している．また Poess らは，ストレージの構成 (ディスク台数，メディア) や CPU の省電力機能，主記憶サイズを変化させた場合の TPC-H の性能と消費電力の関係を評価している [16].

文献 [16] はデータセンタの設計や構築に関する一つの解を提示しているが，データセンタが運用に入った後に発生する実行時省電力化に対する解は示していない．また，文献 [15] では DBMS による省電力制御について述べているが，その定量的な評価やアプリケーション処理性能への影響については述べられていない．

3. ディスクの消費電力特性

ディスクの省電力機能を有効に利用すべく，基本パラメタであるディスクの電力特性を実測し解析した．

3.1 計測環境

図 1 は，ディスクドライブの消費電力の計測のための機器構成図である．我々は OLTP サーバに直結されたディスクの電力を計測した．ディスクはサーバから 4 ピンの電源ケーブルにより電力供給され，赤は 5V，黄は 12V である．5V および 12V の線をそれぞれデジタル電力計 (YOKOGAWA 製 WT1600) に通して電流を計測し，さらに 5V とグランド (黒線)，及び 12V とグランド間の電圧を計測する．ディスクドライブの消費電力は，これら両者の電力の合計値である．

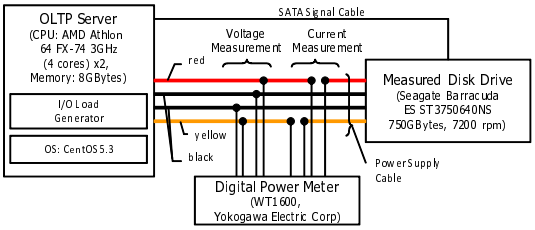


図 1　ディスク消費電力計測環境
Fig. 1　Measurement environment of a disk drive power consumption.

OLTP サーバの CPU は AMD Athlon 64 FX-74 3GHz，cache 1MB, 4 コア× 2，主記憶は 8GB である．計測対象ディスクドライブは Seagate 社の Barracuda ES ST3750640NS (750GB, 7200rpm) である． また，計測時はディスクドライブの write キャッシュを無効化している． これは，DBMS では信頼性の観点から通常ディスクドライブの write キャッシュを使用しないためである．

3.2 ディスクの電力状態と消費電力

本研究で用いたディスクドライブの電力状態は，Active, Idle, Standby, Sleep の 4 つである．Active はディスクに対して I/O が行われている状態，Idle はディスクに対する I/O は行われていないが，即座に I/O を実行できる状態である．Standby はヘッドを退避するとともにディスクの回転を停止した状態であり，Sleep はヘッドの退避，回転の停止とともに，キャッシュへの電力供給も停止している状態である．Standby と Sleep が省電力状態である．また，ディスクを Active/Idle 状態から Standby 状態に移行することを Spindown，Standby 状態から Active/Idle 状態に移行することを Spinup と呼ぶ．

これらの状態のうち本研究では Active，Idle，Standby の 3 状態を用いる．Sleep 状態の消費電力は Standby と同等であるにも関わらず Sleep 状態から他の状態への遷移にはディスクのリセットが必要であり

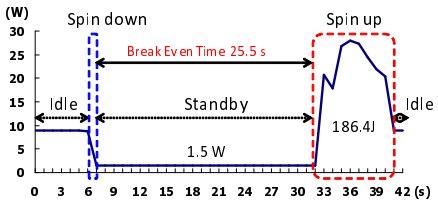


図 2　Standby 時の消費電力と Break-Even Time
Fig. 2　Power Consumption of Standby Status and Break-Even Time.

Standby 状態で代用可能なためである．

ディスクの消費電力を計測した結果，Active 時の消費電力は最大 14.3W，Idle 時の消費電力は 8.8W，Standby 時の消費電力は 1.5W であった．

3.3　Break-Even Time

ディスクの Spindown 及び Spinup により消費されるエネルギーと，ディスクを Standby 状態に移行し，その状態を維持することにより削減できるエネルギーが等しくなる，Standby 状態の持続時間を Break-Even Time と呼ぶ．ディスクを Standby 状態に移行して消費電力を削減するためには，ディスクを Standby 状態に Break-Even Time 以上保つ必要がある．

図 2 は，ディスクを Idle 状態から Standby 状態に移行した後，再度 Idle 状態とした場合の消費電力の計測結果を示した図である．ディスクの Spinup には約 186.4J のエネルギーが必要であった．この計測結果から Break-Even Time を算出したところ，約 25.5 秒であった．つまり，本研究の実験環境においてディスクの省電力機能を利用して消費電力を削減するには，I/O 発行間隔は 25.5 秒以上必要となる．

また，ディスクの Spinup に要する時間は約 8 秒であった．これはそのまま I/O の遅延時間となる．DBMS では read の遅延はそのままトランザクションスループットの低下となるため，read 時のディスク Spinup 回数は極力減らす必要がある．

4.　OLTP の I/O 挙動特性

OLTP の I/O 挙動特性を活かしたディスクドライブの省電力手法を検討するため，OLTP の代表的ベンチマークである TPC-C ベンチマーク [19] を用い，OLTP の I/O 挙動特性を詳細に解析した．

4.1　計 測 環 境

I/O 挙動特性の計測では図 1 に同じ SATA ディスクを 1 台追加し，ディスク 2 台を用いている．OLTP の I/O 挙動計測に使用したソフトウェアおよびその設定を表 1 に示す．ファイルシステムのキャッシュ及びディスクドライブのキャッシュは無効化している．

表 1　ソフトウェアおよびその設定
Table 1　Software and its Configuration.

OS	Cent OS 5.3 (32bit)
File System	ext2
DMBS	MySQL Comunnity Server 5.1.40
OLTP Program	tpcc-mysql
DB Size	4GB (Warehouse=10)
DBMS Buffer Size	2GB
# of Threads	5
Think Time & Keying Time	0 s

我々は，tpcc-mysql [20] を用い，SQL 発行環境を構築した．また，当該システム構成で最も高負荷，すなわちトランザクションスループットが最も高くなる DB サイズとして，Warehouse 数が 10(DB サイズ 4GB) の場合の I/O 挙動を計測した．

Warehouse 数が 10 より小さい場合，行レベルの競合が発生しスループットが低下する．Warehouse 数が 10 より大きい場合は DB バッファ内の更新されたページの数が増えるため，チェックポイント時の更新されたページの出力と当該ページへのアクセスの競合が増加し I/O 数は減少する．Warehouse 数が 10 より小さい場合，及び Warehouse 数が 10 より大きい場合のいずれの場合も I/O 数が減少するため，省電力化は容易になると考えられる．従って，最も高スループットが得られる Warehouse 数 10 で I/O 挙動を解析した．

本計測では，TPC-C を用いてブロックデバイスに対する I/O のトレースを取得した．トランザクションスループットが安定してから 100 万 I/O を超える程度のトレース (約 3 時間分) を blktrace(I/O トレース取得ツール) [21] を用いて取得した．

4.2　TPC-C の I/O 挙動特性

DB データ毎の Read/Write 別の秒当りアクセス回数を図 3 に示す．図より，ログに対しては毎秒 100 回以上 write が発行されており，省電力化の余地はほとんどないことが分かる．しかしログ以外の表・索引に対する I/O は最大でも毎秒 3.5 回程度である．また，District や History，NewOrders，Warehouse などは I/O が少なく，発行された I/O の 90%以上が write であった．District や NewOrder，Warehouse 表は，Stock や Customer 表等と比較してサイズは小さい．このため District や NewOrder, Warehouse 表を格納

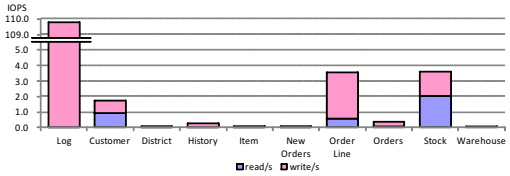


図 3　TPC-C の平均 I/O 数
Fig. 3　Average Disk I/O Number of TPC-C.

したページは，Stock 表などの大きな表と比較し，DB バッファ上にキャッシュされたデータの読み込みが多い．つまり，Stock 表などに比べ read の I/O 数は少なくなり，相対的に write の I/O 比率が高くなっていると考えられる．また，History 表はデータの追加が主体の表であるため，write の I/O 比率が高くなっていると考えられる．

図 4(a) は計測開始から 1.5 時間分の，(b) は 1.5 時間後から 3 時間後までの表・索引への I/O の発行間隔の長さの分布を示している．横軸は I/O 発行間隔の長さ (ログスケール)，縦軸は I/O 発行間隔の累積値である．図中の縦の一点鎖線は Break-Even Time を示している．図 4 (a) を見ると OrderLine を除く各表の I/O 発行間隔の長さには Break-Even Time より長いものが多数存在し，特に Warehouse，District(Warehouse とほぼ重なっている)，Item，NewOrders などの I/O 発行間隔が数百秒以上と長いものがあることが分かる．これは，高スループットで実行中の OLTP 系アプリケーションであってもディスクの省電力化の可能性があることを示している．また，図 4(a) と (b) を比較すると，表・索引毎の I/O 発行間隔の分布は時間経過に対してほとんど変化していないことが分かる．

これらの計測結果より，DB の表・索引に対する I/O の発行間隔は Break-Even Time より長いものが多数あること，I/O 発行間隔が長い表・索引に対する I/O はそのほとんどが write であること，及び表・索引単位では I/O 挙動の変化はほとんどないことが明らかとなった．これらの特徴は，ディスクやブロック単位の I/O 挙動特性解析では把握することはできない．

ここまで我々が計測に用いた環境は，表 1 に示すように，DB の Warehouse 数は 10(DB サイズは 4GB)，DB バッファサイズは 2GB である．一方，TPC-C ベンチマークのレポート等では，DB バッファサイズは DB サイズの 5%程度であり我々が計測を行った環境より小さい．そこで我々は DB バッファサイズが DB

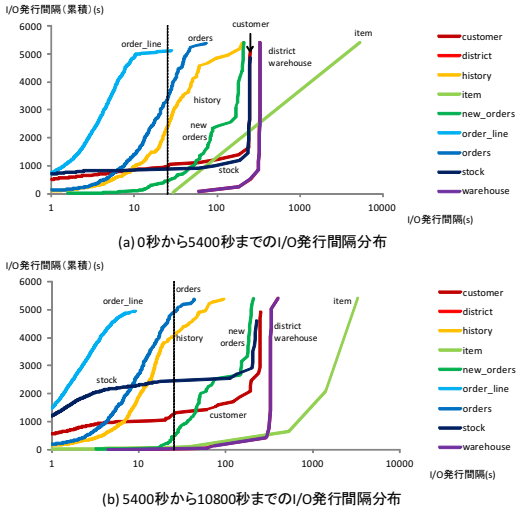


図 4　TPC-C の I/O 間隔分布
Fig. 4　I/O Interval Distribution of TPC-C.

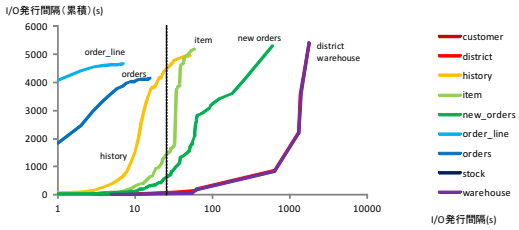


図 5　DB サイズ 40GB 時の TPC-C の I/O 間隔分布
Fig. 5　I/O Interval Distribution of TPC-C (40GB DB).

サイズの 5%となる Warehouse 数 100 (DB サイズ 40GB) の DB を用いた場合についても I/O 発行間隔の調査を行った．この結果を図 5 に示す．図より，I/O 発行間隔が短く，省電力化が期待できない OrderLine や Order 表では Break Even Time より長い I/O 発行間隔は見られなくなった．しかし，Warehouse や District 表など I/O 発行間隔が長く省電力化が期待できる表では，Break Even Time 以上の I/O 発行間隔の長さは DB バッファサイズが DB サイズの 50%の場合と比較し，同等かそれ以上であることが分かる．省電力化が期待できる表の I/O 発行間隔が長い方がディスクの省電力化には有利である．そこで，我々はよりディスクの省電力化が厳しい環境で提案手法の有効性を評価するため，DB の Warehouse 数を 10(DB サイズ 4GB)，DB バッファサイズを 2GB とした．

我々は，今回の TPC-C 実行時 (Warehouse 数: 10,

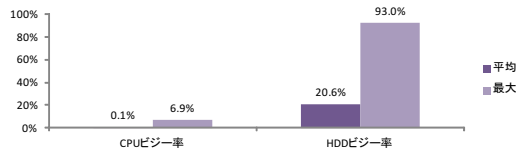


図 6 Warehouse 数 10 の場合の CPU 及び HDD のビジー率
Fig. 6 CPU and HDD Busy Rate at 10 Warehouses DB.

DB サイズ: 4GB) における計算機資源のビジー率を調査した．この結果を図 6 に示す．CPU ビジー率は平均 0.1%，最大 6.9%であるのに対し，DB データを格納したディスクのビジー率は平均 20.6%，最大 93.0%であった．我々の計測環境では CPU ビジー率と比較してディスクのビジー率の方が高く，アプリケーションスループットは入出力性能が支配的である．従って，アプリケーション実行時のディスク省電力化にとって，我々の計測環境は厳しい環境と考える．

5. OLTP の I/O 挙動特性を用いた省電力手法

本章では，前章で示した TPC-C の I/O 挙動特性を基に，アプリケーション実行中にも省電力化可能なディスクの省電力手法を提案する．前章の結果から TPC-C においても表・索引に対する I/O 発行間隔は Break-Even Time より長いものが多数あり，また表・索引単位では I/O 挙動の経時変化はほとんどない．これらの知見から，I/O 発行間隔が短い表・索引を同一のディスクに配置することにより他のディスクの I/O 発行間隔を伸ばす可能性があると考えられる．さらに，I/O 発行間隔が長い表・索引に対する I/O のほとんどが write である．この特性と DBMS の更新ログ先行書き出しプロトコルの性質を利用し，ディスクに通常適用される write をコミット時まで実際の書き込みを伸ばすことができる可能性がある．以下，アプリケーションの I/O 挙動特性を用いた実行時ディスク省電力手法について，データ配置制御および write 遅延それぞれについて説明する．

5.1 データ配置制御

ほとんどの I/O 発行間隔が Break-Even Time より短い，つまり I/O アクセス頻度の高い表および索引をなるべく少数のディスクに集め，残りのディスクを省電力化の対象とする．従来の手法では，物理ブロックアドレスやファイル等デバイスレベル，ファイルシステムレベルのアクセス頻度に着目していたが，本手法では表・索引を対象とする．

Algorithm 1 実行時省電力制御アルゴリズム

```
物理ブロックと TPC − C の論理的データ単位の I/O 挙動の監視を開始;
ディスクの省電力状態とディスクの消費電力の監視を開始;
while DBMS 実行中 do
  Hot・Cold データ計算 ();
  if Hot データが Cold データとなる，又は Cold データが Hot データとなる then
    Hot データを配置するディスク計算 ();
    移動対象データおよびデータの移動先の計算 ();
    if 新配置の電力削減量 > 現在の配置の電力削減量 then
      データ再配置;
      if Cold ディスク数 > 0 then
        Write 遅延開始;
      else
        Write 遅延停止;
      end if
    end if
    if Cold ディスク数 > 0 かつ省電力機能が適用されていない Cold ディスクがある then
      省電力機能が適用されていない Cold ディスクに省電力機能使用のためのパラメタを設定;
    end if
  end if
end while
```

ログおよび表・索引は物理ブロックと比較して大きな単位だが，4 章の分析結果 (図 4) から表・索引の論理的データ単位での I/O 発行間隔分布に基づくデータ配置でも高い省電力効果を得られる可能性があると考えられる．

実行時のデータ配置制御の概要を Algorithm1 に示す．データの初期配置に関しては，IOPS が複数ディスクでできるだけ均等になるように配置されていると仮定する．また，I/O 挙動モニタリングは，通常の DBMS のモニタと同じ頻度で行う．

Hot・Cold データ計算 論理的データ単位をモニタリングの結果を用いて Hot データと Cold データに分類する．Cold データとは同じディスクに格納されている他のデータへのアクセスは考えず，当該データのみのアクセス履歴から計算してディスクの消費電力を削減できるデータである．Cold データ以外のデータを Hot データとする．Algorithm1 を開始，あるいは前回データ再配置を実行してから現在までの時間間隔を T，データ j のみをディスクに配置したと仮定した場合に，期間 T 内でディスクに対して発行される I/O の発行間隔のうち Spindown タイムアウトより長い I/O 発行間隔を l_i とする．期間 T 内に k 回 l_i が観測され

た場合のディスクの消費電力削減量は $\sum_{i=1}^{k}((Idle$ 時電力 $- Standby$ 時電力$) \times l_i -$ ディスク起動エネルギー) である．

Hot データを配置するディスク計算　Hot データを格納するディスク数 N を求める．I_{max} をディスクが提供できる最大 IOPS，S_{max} をディスクの容量とすると，$N = max(\lceil Hot$ データの合計 $IOPS$ の最大値$/I_{max}\rceil, \lceil Hot$ データの合計サイズ$/S_{max}\rceil)$ である．$IOPS$ とは秒当たり I/O の数のことである．次に，ディスク内の合計 Hot データ量の降順にソートし，上位 N ディスクを Hot データを配置するディスクとする (以降これらのディスクを Hot ディスク，残りのディスクを Cold ディスクと呼ぶ)．Hot データの容量でソートするのはデータの移動量を削減するためである．

移動対象データおよびデータの移動先の決定　Cold ディスク中の Hot データを移動対象データとする．まず，Hot ディスクのうち，空き容量が，移動しようとする Hot データのサイズ以下のディスクを選択する．次に，それらのディスクの中から，移動対象の Hot データを配置した後の最大 IOPS が I_{max} 以下かつ Hot データ移動後の最大 IOPS が最小となる Hot ディスクを移動先として選択する．Hot ディスクに十分な空きがなく Hot データをいずれの Hot ディスクにも移動できない場合は，Hot ディスク上の Cold データを Cold ディスクに移動する．その際，移動する Hot データ以上の容量を持つ Cold データを選択する．次に，当該 Cold データを移動した場合に I/O 発行間隔が最も長くなる Cold ディスクを選択し当該 Cold データを移動する．容量および性能の要件を満たす Hot ディスクがなく，かつ Cold データの移動もできない場合，Hot ディスクの数を 1 増やして再度本ステップを実行する．

ディスクの電力削減量　Algorithm1 を開始，あるいは前回データ再配置を実行してから現在までの時間長を T，期間 T 内でディスクに対し発行される I/O の発行間隔のうちの Spindown タイムアウトより長い I/O 発行間隔を l とする．期間 T 内に k 回 l が観測された場合のディスクの電力削減量は，$\sum_{i=1}^{k}((Idle$ 時電力 $- Standby$ 時電力$) \times l_i -$ ディスク起動エネルギー) である．現在のデータ配置，および新たなデータ配置のそれぞれについて上記を計算する．

5.2 Write 遅延

WAL プロトコルは，DB バッファ上の DB データをディスクに書き出す前に，当該データの更新をログに書き出すことを保証する．これによりトランザクションのコミットとは独立した契機で DB バッファ上のデータをディスクに書き出すことが可能となる．write 遅延はこのプロトコルを利用して write をまとめてディスクに書き出すことにより Cold ディスクへの I/O 発行間隔を伸ばす手法である．

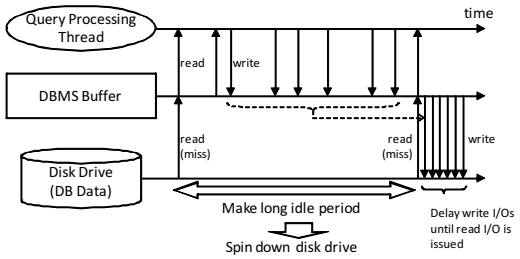


図 7　Write 遅延方式
Fig. 7　Delayed Write I/O.

DB に対して write を行う契機は，i) チェックポイントと呼ばれる更新された DBMS バッファページのディスクへの書込み，ii) チェックポイントによるディスクへの負荷を低減するための (チェックポイントに先立つ) データの write，及び iii)DB バッファに空きがない場合に空きを作るために更新されたデータを DB バッファから追い出す場合，の 3 通りである．

Write 遅延はこのうち ii) のチェックポイントに先立つデータの write を遅延する．Cold ディスク上のデータは I/O 数が少ないため ii) を利用するメリットがなく，また Cold ディスクに対する更新を一括で write してもディスクの負荷は低いままと考えられるためである．図 7 にその概要を示す．

Write 遅延は，Algorithm1 で Write 遅延対象データと判定されたデータを対象に行う．DB バッファ上のこれらのデータが更新された場合，これらのデータをチェックポイント開始時，あるいは DB バッファの更新ページ数の比率が α を超えるまで保持する．そしてチェックポイント開始あるいは更新ページ数の比率が α を超えた時点で，write 遅延対象データの更新を更新順序を変更せずにディスクに反映する．Write 遅延処理は DBMS の WAL プロトコルと整合しており，write 遅延により DBMS の信頼性が低下することはない．

図 8 に OrderLine 表および索引に write 遅延を適用した場合と適用しない場合の I/O トレースを示す．

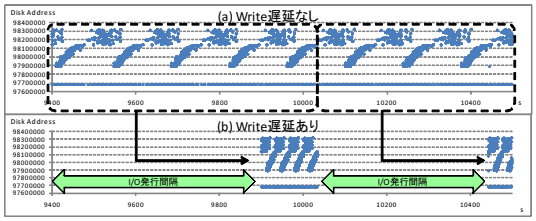


図 8　Write 遅延による I/O 発行間隔の延伸
Fig. 8　Enlarging I/O Interval by Delayed Write I/O.

図 8 より，write 遅延により数百秒の I/O 発行間隔が生成できていることが分かる．

6. 評　　価

本章では，提案手法の評価方法および結果について述べる．提案手法の効果を確認するために，ディスク 2 台および 5 台を用いた構成で評価を行った．また，既存の実行時ディスク省電力手法として，ディスクのブロック単位でデータの配置制御を行う手法 (Dynamic Data Reorganization; DDR) [14]，及びファイル単位でデータの配置制御を行う手法 (Popular Data Concenteration; PDC) [11] を用いて提案手法との比較を行う．

6.1　評価方法及びパラメタ

提案手法，PDC，DDR のそれぞれについて，ディスクの消費電力，およびトランザクションスループット，データの移動量を用いて比較する．4 章の TPC-C 実行時の I/O トレース (基本 I/O トレース) を基に各手法を適用した後の I/O トレースを生成し，それらをディスク上で再生することによりディスクの消費電力を計測した．

6.1.1　基本 I/O トレース

ディスク 2 台の場合は 4 章で取得した I/O トレースを基本 I/O トレースとした．5 台の場合は，4.1 節で述べた構成にさらに同一の SATA ディスクを 3 台追加し基本 I/O トレースを取得した．ディスク 5 台構成は，ディスク 2 台の場合と同一規模の DB を用い，その構成で最も高いスループットを達成するためにスレッド数を増やした．TPC-C の warehouse 数，Think Time，DB バッファサイズはディスク 2 台の場合と同じとしている．

6.1.2　各手法の I/O トレース生成

提案手法の I/O トレースは，Algorithm1 を基本 I/O トレースの先頭から適用することにより生成した．つまり，データ配置制御に合わせ，それ以降データの移動先のディスクの同じ物理アドレスにアクセスするように変更した．Write 遅延を適用したデータの write はチェックポイント時又は DB バッファの更新ページ数の比率が α を超えるまで遅延させている．また，データの移動に必要な新たな I/O も付加している．DDR ではブロック交換後の I/O 先の変更，及びブロック交換に必要な新たな I/O を基本 I/O トレースから生成した．PDC では，ファイル移動後の I/O 先の変更，及びファイル移動に必要な新たな I/O を基本 I/O トレースから生成した．提案手法と PDC では，データ移動のための IOPS が $I_{max}/2$ を超えないよう IOPS を制御した．DDR は Cold ディスク上のブロックにアクセスが行われる毎にブロックを交換するため，それに従ってデータ移動のための I/O を付加した．ディスクの Spindown タイムアウトは OS で設定可能な最小値である 5 秒，Write 遅延における DB バッファの更新ページ数の比率 α は 50%とした．

表 2　データの配置 (ディスク 5 台)
Table 2　Data Placement (5 disks)

Disk #	Data on Disk
Disk #1	Log
Disk #2	Stock
Disk #3	OrderLine
Disk #4	Customer
Disk #5	District, History, Item, NewOrders, Orders, Warehouse

6.1.3　データ配置の計算

(a) データ初期配置　ディスク 2 台の場合は，各手法ともログを Disk#1 に，表・索引を Disk#2 に配置した．ディスク 5 台の場合は，ログを Disk#1 に，表・索引を 4 章で取得した表・索引毎の IOPS に基づき 5 章で述べた初期配置に従い Disk#2-4 に配置した．実際の配置は表 2 に示すとおりである．

(b) データ再配置　提案手法は Algorithm1 を用いてデータの再配置を行った．DDR，PDC はそれぞれ文献 [11], [14] に示された方式に基づき再配置を行った．

DDR における $TARGET\_TH(Threshold)$ は 50，$HIGH\_TH$ は 100，LOW_TH は 25 とした．DDR におけるデータ再配置の契機は，文献 [14] に従い IOPS が LOW_TH 以下のディスク上のブロックに I/O が行われた時点とした．PDC のキュー数は文献 [11] に従い 12 とした．ディスク 2 台の場合はキュー 0 から 5 を Disk#1 に，6 から 11 を Disk#2 にそれぞれ対応させた．ディスク 5 台の場合は，キュー 0,1 を Disk#1,

1,2 を Disk#2，3,4 を Disk#3, 6-8 を Disk#4, 9-11 を Disk#5 にそれぞれ対応させた．PDC におけるデータ再配置の契機は文献 [11] に従い 30 分とした．

6.1.4 ディスク消費電力の計測

前節で生成した I/O トレースをディスク上で再生し，ディスクの消費電力を計測した．計測期間は基本 I/O トレースの先頭から 30 分である．トレース再生には blktrace [21] を用いた．

6.1.5 トランザクションスループットの計算

提案手法では，まず Algorithm1 により求めたデータ配置に合わせてデータをディスクに配置し，TPC-C を実行してスループットを計測した．スループットの計算値 TP_E は，TP を計測されたスループット，w を計測期間中のスピンアップ待ち時間の合計値，d を計測期間の長さとすると，$TP_E = TP \times (1 - w/d)$ により求めた．PDC については PDC のデータ配置計算アルゴリズムにより求めたデータ配置に合わせてデータをディスク上に配置し，提案手法と同じ方法でスループットを計算した．DDR については数十 I/O 毎にブロック配置が変化する可能性があるため，基本 I/O トレース取得時に取得したトランザクションスループットを前述の式に当てはめスループットを計算した．

6.2 評価結果

6.2.1 ディスク 2 台の場合

図 9. にディスク 2 台の場合のディスクの消費電力とトランザクションスループットをそれぞれ示す．

図 9 より，提案手法の消費電力は 17.3W であり省電力手法なしの場合と比較して消費電力を 23.3%削減できた．DDR の消費電力は省電力手法なしの場合と同等，PDC の消費電力は 28.7W であり省電力手法なしの場合と比較して 26.9% 増加した．DDR，PDC と比較し提案手法は消費電力を削減できている．

提案手法では，I/O の発行間隔が短いデータを一方のディスク (Disk#1) に配置したこと，および他方のディスク (Disk#2) に配置したデータへの write 遅延の適用により，Disk#2 の I/O 発行間隔を Break Even Time 以上に伸ばすことができた．DDR の省電力効果が見られないのは，2 台のディスクの合計 IOPS を $TARGET\_TH$ で除した値を切り上げた値 (Hot ディスク台数) が 2，Cold ディスクが 0 となり省電力機能が適用されなかったためである．PDC は消費電力が約 27%上昇した．これはデータの配置をディスクへの I/O 発行間隔ではなく IOPS に基づき実施しているためである．この結果，IOPS が少ないデータを格納したディスクの I/O 発行間隔が，ディスクの Spindown タイムアウト時間である 5 秒よりは長く Break Even Time よりは短い状態となり，消費電力を削減することはできなかった．

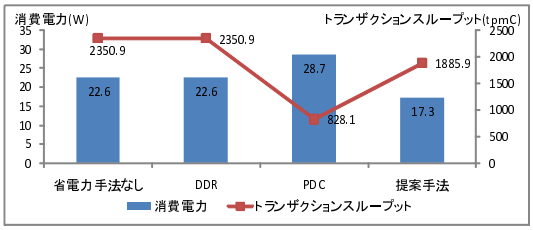


図 9 ディスク消費電力とトランザクションスループット (2HDD)

Fig. 9 Disk Power Consumption and Transaction Throughput (2HDD).

提案手法のトランザクションスループットは 1885.9tpmC であり省電力手法なしの場合と比較して 19.8%減，DDR は省電力化していないのでトランザクションスループットに変化はない．PDC は 828.1tpmC(64.8%減) であった．

提案手法で，DBMS のログと DB の表・索引が同一のディスクに配置されたことによるログ I/O の応答時間の低下のため，トランザクションスループットが低減している．PDC がトランザクションスループットが大きく低下した理由は提案手法と同様にログ I/O の応答時間の低下に加え，ディスクの起動待ちが 30 分間に約 50 回発生したためである．

初期配置からのデータ移動量は DDR は省電力手法が適用されなかったため 0，PDC は 4,522MB，提案手法は 472MB であった．PDC の移動量が多い理由は，容量の大きなデータ (Stock, OrderLine 等) が PDC が IOPS が少ないデータを配置するディスクと判断したディスク上にあり，それらを移動したためである．提案手法ではディスク内の Hot データの容量により Hot ディスクを決定しているため，データ移動量を削減できた．

トランザクション当たりの消費電力は省電力手法なしの場合は 9.6W/tpmC であるのに対し，提案手法が 9.2W/tpmC，DDR が 9.6W/tpmC，PDC が 34.6W/tpmC であった．省電力手法の場合と比較して提案手法は 4.3%減と最もすぐれており，DDR は同等，PDC は 260.4%増であった．

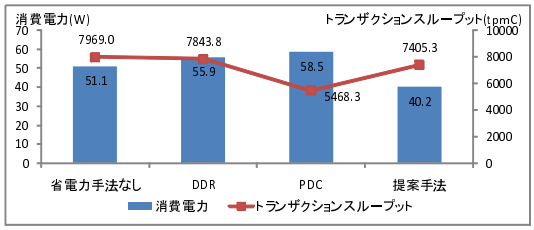


図 10　ディスク消費電力とトランザクションスループット (5HDD)
Fig. 10　Disk Power Consumption and Transaction Throughput (5HDD).

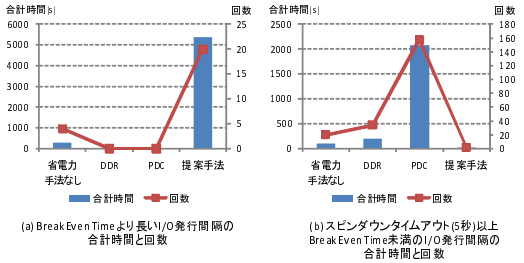


図 11　I/O 発行間隔
Fig. 11　I/O Interval.

6.2.2　ディスク 5 台の場合

ディスク消費電力とトランザクションスループット
図 10. に，ディスク 5 台の場合のディスクの消費電力とトランザクションスループットを示す．

提案手法の消費電力は 40.2W であり，省電力手法の場合と比較して 21.4% 低減できた．DDR は 55.9W(9.4%増)，PDC は 58.5W(14.5%増) であった．DDR の消費電力が増加した理由は，OLTP はランダム I/O のため IOPS が多いブロックの予測を誤ったためである．この結果 Cold ディスクへの I/O 発行間隔が Break-Even Time より短くなり，ディスクの Spinup に伴う電力が増加した．PDC の消費電力が増加した理由は，ディスク 2 台の場合と同様，IOPS に基づきデータ配置を決めたため，各ディスク毎に I/O 発行間隔が平均化され Break Even Time より長い I/O 発行間隔がほとんどできなかったためである．

また，提案手法のトランザクションスループットは 7405.3tpmC であり省電力手法なしの場合と比較して 7.1%減であった．DDR は 7843.8tpmC(1.6%減)，PDC は 5468.3tpmC(31.4%減) であった．PDC のスループットが低いのは，I/O 発行間隔が短くディスクの起動待ちが多数発生したためである．提案手法と DDR はそれほど低下していない．

初期配置からのデータ移動量は DDR は 7,184MB，PDC は 1,684MB，提案手法は 180MB であった．DDR の移動量が多い理由は IOPS の高いブロックの予測に失敗し Cold ディスク上のブロックの移動が多発したこと，およびデータ交換をおこなっているためである．ディスク 2 台の場合と同様の理由により，提案手法はデータ移動量を削減できている．

トランザクション当たりの消費電力は，省電力手法なしの場合が 6.4W/tpmC であるのに対し，提案手法が 5.4W/tpmC，DDR が 7.1W/tpmC，PCD が 10.7W/tpmC であった．消費電力制御なしの場合と比較して DDR が+11.1%，PDC は+66.8%と悪化したが，提案手法は-15.4%と大幅な削減を達成している．

以上の結果より，より高いスループットが求められる環境においても提案手法がアプリケーション実行時のディスクの消費電力削減できることが示された．

I/O 発行間隔　図 11(a) にディスク 5 台を用いて評価を行った際の Break Even Time 以上の長さの I/O 発行間隔の合計時間と回数を，(b) にスピンダウンタイムアウト (5 秒) 以上 Break Even Time 未満の I/O 発行間隔の合計時間と回数をそれぞれ示す．

図から分かるように，提案手法が，タイムアウト時間以上 BreakEven Time 未満の I/O 発行間隔を増加させることなく Break Even Time 以上の長さの I/O 発行間隔を多く作り出せていることが分かる．これは，提案手法が I/O 発行間隔を基準にデータ配置を決定していること，および write 遅延の効果である．Write 遅延を行った場合，ディスク 5 台構成ではディスクに対する write 間隔は約 350 秒であったが，write 遅延を行わない場合は約 11 秒であった．この結果より，アプリケーションの I/O 挙動特性と DBMS のディスクへの write 方法に関する知識を活用することで，I/O 発行間隔を大きく伸ばすことが可能であることを確認できた．

ディスク起動待ち時間の影響　3 章で述べたように，ディスクを Standby 状態から Active/Idle 状態に移行するには約 8 秒の時間を要する．そこで，我々はディスクの起動待ちが TPC-C の応答時間にどの程度影響を与えるかを調査した．TPC-C では，StockLevel を除く 4 つのトランザクションの応答時間の 90%が 5 秒未満，StockLevel は 20 秒未満であることが規定されている [19]．そこで，省電力手法なしの場合とデータ配置制御のみを行った提案手法 (ディスク省電力機能は使

用せず) に起動待ち回数を加算した場合のそれぞれについて，ディスク 5 台の場合の応答時間の 90%値を調査した．この結果両ケースとも NewOrder, Payment, OrderStatus, StockLevel の応答時間の 90%値は 0.2 秒，Delivery が 0.4 秒であり，TPC-C の性能要件を満たしていた．

DB サイズに関する考察　本評価で用いた DB は Warehouse 数が 10，サイズは約 4GB である．これは，ディスク 5 台を用いた場合の全ディスク容量の約 0.1%と小さい．しかしながら，TPC-C 実行時のディスクへのアクセス頻度は DB サイズではなく，トランザクションスループットにより決まる．実験結果から分かる通り，ディスク 5 台の場合のトランザクションスループットはディスク 2 台の場合と比較し十分に高く，Warehouse 数が 10(DB サイズ 4GB) の場合は 5 台のディスクに対する I/O 負荷は高いと考えられる．一方，DB サイズを増やした場合，図 5 に示した通り OrderLine 表などの省電力効果の低い表の I/O 発行間隔は短くなる (I/O 数が増加する) が，Warehouse や District 表などの省電力効果の高い表の I/O 発行間隔は長くなる．これは，DB サイズ (Warehouse 数) が大きくなると，I/O に伴う DB バッファのページのロック回数が増え I/O オーバーヘッドが増加するためである．この結果，DB サイズが大きい場合は省電力効果の高い表の I/O 発行間隔が伸び，省電力化の機会が増えると考えられる．本計測環境 (Warehouse 数: 10, DB サイズ 4GB) は，ディスクの省電力化にとって厳しい設定となっているが，十分に省電力化が可能となっている．従って，DB サイズがより大きな環境でも，提案手法はディスクの省電力化に有効であると考える．

7. 考　　察

7.1 データ再配置の契機

提案手法は，Cold データと判断されていたデータが Hot データと判断された契機，あるいは Hot データと判断されていたデータが Cold データと判断された契機でデータの配置を再計算している．

今回の実験では高スループットで実行しているため生じないが，トランザクションリクエスト数が増えると，Cold データと判定されていたデータが Hot データになり，ディスクの起動ペナルティが増加し消費電力が削減できなくなる可能性がある．これに対し，Cold データが Hot データになった契機でデータ再配置することにより，大幅な消費電力の上昇を抑えることが可能であると考える．

逆にトランザクションリクエストが減ると，I/O 発行間隔は広くなり，Hot データと判断されていたデータが Cold データになる場合がある．これに対し，Hot データが Cold データになった契機でデータ再配置を行うことにより，さらに消費電力を削減することが可能になると考える．

PDC ではデータ再配置は 30 分毎に行うため，次のデータ再配置までの間のロスが大きくなると考えられる．DDR は物理ブロック単位のデータ再配置であるが TPC-C では IOPS が増えるブロックの予測が難しく高い省電力効果を得ることは困難であると考えられる．アプリケーションレベルの I/O 挙動特性を利用することで一層の省電力化が可能となる．

7.2 OLTP 以外の大量データ処理アプリケーションへの適用

大量のデータを処理するアプリケーションとして，OLTP 以外にも意思決定支援システム (DSS) や科学技術計算向けの大規模デジタルライブラリなどがある．これらのアプリケーションでは，データに対する順次アクセスが中心であることが分かっている．このためディスクデバイスレベルから得られる情報を用いても省電力機能の適用が可能な I/O 発行間隔を見つけることは容易であると考えられる．しかし，ディスクデバイスから得られる情報を用いたデータ移動では，Cold ディスク上のブロックに I/O が行われるとそれを複数の Hot ディスクの中で最も IOPS が少ないブロックと交換する．このため，順次アクセスが行われるデータが I/O の発行順序とは異なる順序で複数のディスクに配置されることになり，I/O 性能が低下する可能性がある．一方，提案手法ではデータ単位として表・索引を考え，アプリケーションの論理的な I/O 挙動特性を用いるため，アクセスの順序を保持したままデータを再配置する．このため，性能への影響は低いと考えられる．

8. ま と め

今後も，デジタルデータは爆発的に増大すると考えられ，それらのデータを格納するために増大し続けるディスクの省電力は急務である．本論文では，アプリケーション実行時のディスク省電力手法を開発し，OLTP を用いて評価した．実行時省電力手法を開発するために，ディスクの消費電力特性を計測するととも

に，TPC-C を実行しその I/O 挙動特性を解析した．さらにこの解析結果に基づき，アプリケーション I/O 挙動特性を用いた新たなディスクの実行時省電力手法を提案した．既存手法の DDR，PDC と比較した結果，5 台のディスク構成の場合には提案手法により数%のトランザクションスループットの低下でディスクの消費電力を 20%以上削減できることを示した．

今後，我々は提案手法を RAID ストレージや TPC-H や TPC-W などの他のアプリケーション向けに拡張すると同時に提案手法を実装し，大規模ストレージ上で評価を行う予定である．

文　献

[1] D. Reinsel, “A plateau in sight for the rising costs to power and coll the world’s external storage?,” IDC White Paper #225016, 2010.

[2] A.G. Yoder, “Green storage technologies, capex and opex,” Storage Networking World 2011 Spring, 2011.

[3] S. Worth, “Green stroage - the big picture,” Storage Networking World 2011 Spring, 2011.

[4] D. Reinsel, “White paper datacenter ssds: Solid footing for growth,” IDC White Paper # 210290, 2008.

[5] M. Poess and R.O. Nambiar, “Power cost, the key challenge of today’s data centers: a power consumption analysis of tpc-c results,” VLDB ’08 Proceedings, pp.1229–1240, ACM Press, 2008.

[6] A.E. Papathanasiou and M.L. Scott, “Energy efficient prefetching and caching,” Proc. of USENIX 2004 Annual Technical Conference, pp.255–268, USENIX Association Berkeley, 2004.

[7] D. Li and J. Wang, “Eeraid: Power efficient redundant and inexpensive disk arrays,” Proc. 11th Workshop on ACM SIGOPS European Workshop, pp.174–180, 2004.

[8] X. Yao and J. Wang, “Rimac: A novel redundancy based hierarchical cache architecture for power efficient,” High Performance Storage System Proc. 2006 EuroSys Conference, pp.249–262, 2006.

[9] T. Heath, E. Pinheiro, J. Hom, U. Kremer, and R. Bianchini, “Application transformations for power and performance-aware device management,” 11th International Conference on Parallel Architectures and Compilation Techniques, pp.121–130, 2002.

[10] D. Colarelli and D. Grunwald, “Massive arrays of idle disks for storage archives,” Supercomputing, ACM /IEEE 2002 Conference, pp.47–57, 2002.

[11] E. Pinheiro and R. Bianchini, “Energy conservation techniques for disk array based servers,” Proc. 18th Annual International Conference on Supercomputing, pp.68–78, ACM, 2004.

[12] O.M.Q.J. Weddle, C. and A.A. Wang, “Paraid: A gear-shifting power-aware raid,” 5th USENIX Conference on File and Storage, pp.245–267, USENIX Association, 2007.

[13] K.R.U.L. Verma, A. and R. Rangaswami, “Srcmap: Energy proportional storage using dynamic consolidation,” 8th USENIX Conference on File and Storage Technologies, pp.267–280, USENIX Association, 2010.

[14] E. Otoo, “Dynamic data reorganization for energy saving in disk storage systems”. Scientific and Statistical Database Management Conference, 2010

[15] S. Harizopoulos, M.A. Shah, J. Meza, and P. Ranganathan, “Energy efficiency: The new holy grail of data management systems research,” 4th Biennial Conf. on Innovative Data Systems, pp.112–123, 2009.

[16] M. Poess and R.O. Nambiar, “Tuning servers, storage and database for power efficient data warehouse,” 26th IEEE International Conf. on Data Engineering, pp.1006–1017, IEEE Computer Society, 2010.

[17] N. Nishikawa, M. Nakano, and M. Kitsuregawa, “Low power mnagement of oltp applications considering disk drive power saving function,” 21th International Conference of Database and Expert Systems Applications, pp.241–250, Springer, 2010.

[18] N. Nishikawa, M. Nakano, and M. Kitsuregawa, “Potentiality of power management on database systems with power saving function of disk drives”. The 22nd Australian Database Conference, 2011

[19] “Tpc-c, an online transaction processing benchmark”. Transaction Processing Performance Council, http://www.tpc.org/tpcc/

[20] V. Tkachenko, “tpcc-mysql”. https://code.launchpad.net/ percona-dev/perconatools/tpcc-mysql

[21] A.D. Brunelle, “btrecord and btreplay user guide,” 2010. http://www.cse.unsw.edu.au/ aaronc/iosched/doc/btreplay.html

（平成 xx 年 xx 月 xx 日受付）

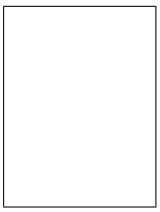
西川　記史

平元年神戸大学工学部計測工学科卒業．平 3 年同大学大学院工学研究科計測工学専攻修士課程修了．同年（株）日立製作所入社．横浜研究所にてストレージ管理ソフトウェアの研究開発に従事．現在同研究所主任研究員及び東京大学大学院情報理工学系研究科電子情報学専攻．1998 年度情報処理学会山下記念賞受賞．情報処理学会，日本データベース学会会員．

中野美由紀　（正員）

東京大学理学部情報科学科卒業．博士（情報理工学）．富士通（株）勤務．1985 年 7 月東京大学生産技術研究所助手（2004

年助教）.2008 年 7 月特任准教授．データベースシステム，ストレージシステム，データ工学の研究に従事．IEEE，電子情報通信学会，情報処理学会，ACM, 日本データベース学会各会員.



喜連川　優　（正員：フェロー）

昭 53 東大・工・電子卒．昭 58 同大大学院工学研究科情報工学専攻博士課程修了．工学博士．同年同大生産技術研究所講師．現在，同教授．平 15 より同所戦略情報融合国際研究センター長．データベース工学，並列処理，Web マイニングに関する研究に従事．情報処理学会フェロー，日本データベース学会理事．ACM SIGMOD Japan Chapter Chair，本会データ工学研究専門委員会委員長歴任．VLDB Trustee，IEEE ICDE，PAKDD，WAIM などステアリング委員，SNIA 日本支部顧問，文科省特定領域研究「情報爆発 IT 基盤」領域代表を務める.

Abstract According to rapid growth of digital data, a used disk capacity is increased rapidly. Power consumption rate of storages in IT equipment power consumption is also increased. In this paper, we discuss a runtime power saving method of OLTP DBMS which process huge data. Generally, OLTP accesses its data frequently, therefore it is difficult to save OLTP's runtime power consumption of disks. We analyze I/O behaviors of OLTP, and show a power saving potential of disks which OLTP is running. Then we propose a novel disk power saving methods based on the I/O behavior. Finally, we show an efficiency of our proposed methods can save a power consumption of disks which run OLTP by a simulation on an actual OLTP server.

Key words Disk, Power Saving, OLTP, Data Placement Control, Write Delay