Vol.2014-HPC-145 No.31
2014/7/30

Cluster-wide RAIDの実装と評価

大辻 弘貴[1,3,4]　建部 修見[2,3]

概要：ビッグデータ処理やデータインテンシブコンピューティングの広まりにより，大容量データを高速かつ確実に取り扱う技術に対する要求は日に日に高まっている．また，それらを支える記憶装置についても，容量及びアクセス性能の双方において急速に進歩している．ところが，記憶装置を利用するためのシステムに関しては，今後想定されるエクサスケールまたはそれ以上の規模において，十分な性能を発揮できるかは不透明である．本稿においては，ネットワーク経由でノードレベルの冗長性を確保し，高性能なデータアクセスを提供する Cluster-wide RAID の実装について，その詳細および評価について述べる．

1. 序論

計算機が取り扱うデータ量は増加し続けており，エクサバイトの時代は目前に迫っている．科学技術の分野においては当然のことながら，ビジネスにおいてもビッグデータ処理を目的とした大容量データの取り扱いに対する需要は高い．最近は，旧来より保たれてきたファイルとディレクトリという形のデータ管理ではなく，オブジェクトを保存するストレージとして大容量記憶装置を構築する方向性が見られる．いずれにせよ，システムはネットワークを用いて多数の記憶装置により構成される．そのような環境において信頼性を確保しつつ十分な性能を出すためには，ネットワークを介してデータの読み書きを効率的に行う仕組みが不可欠である．信頼性に関しては，記憶装置のノードレベルでの耐故障性は不可欠であり，従来の RAID[1] がカバーする範囲とは異なる．この問題に対しては，レプリケーション（ファイル複製）[2] が頻繁に用いられるが，これは複製である以上当然元のデータの 2 倍以上の記憶領域を必要とし，その分の書き込みも行わなければならないので，効率の点では不利である．

本稿においては，RAID の記録方法をノード間に拡張した，Cluster-wide RAID6 の実装について詳述する．ノード間における RAID について，RAID-4 に相当するものは筆者らにより発表されており [3]，単純に書き込み元でパリティを生成してすべてのデータを書き込む方法よりも，提案手法を用いた場合のほうが高い書き込みスループットが得られることを示している．本稿は，これを RAID-6 に拡張したものについて取り扱っている．RAID-4 は任意の 1 台の故障にしか耐えられないが，RAID-6 では任意の 2 台の故障に対応できるようになる．この Cluster-wide RAID-6 について，InfiniBand ネットワークを利用し，RDMA を活用して実装した．以降では，その詳細及び評価を示す．

2. 関連研究

ネットワークを利用するストレージシステムや，冗長記録を行うシステムが本研究に関連している．具体例は多く存在するが，そのうち良く知られた物を例に上げる．Lustre[4] や Ceph[5]，Gfarm[6] はいずれも分散ファイルシステムであり，複製機能や冗長記録を行う機能を備えている．Lustre と Gfarm はそれぞれ複製によって信頼性を確保している．しかしながら，序章にて述べたように，複製は最低でも倍以上の領域が必要になることから，今後データ量の増加が見込まれる状況においては避けたいところである．また，Ceph は冗長記録を利用しているが，リードソロモン符号を用いており，xor をベースとした RAID-6 を構成している本研究とは差異がある．本研究で用いている方式は，少ない xor 演算によって処理を行うため，高速な処理が可能となる．特に，数 GB/s を超えるスループットにおいては，この点は重要である．ネットワークアクセスに関しては，Lustre は LNET と呼ばれるネットワークライブラリを持っており，RDMA(Remote Direct Memory

[1] 筑波大学大学院システム情報工学研究科
Graduate School of Systems and Information Engineering, University of Tsukuba
[2] 筑波大学システム情報系
Faculty of Engineering, Information and Systems, University of Tsukuba
[3] 独立行政法人科学技術振興機構 CREST
JST CREST
[4] 独立行政法人日本学術振興会 特別研究員 (DC2)
JSPS DC2

Access) によるデータ転送をネイティブでサポートしている．Ceph や Gfarm は現時点においては，プロトコルレベルでの RDMA 対応は行われていない．一方で本稿で示す実装は，低いレイヤの API を直接利用して RDMA を活用しており，低オーバヘッド・メモリコピーとなっている．

ネットワークを経由したデータの取扱という観点では，DRBD[7] などが挙げられる．これは，ブロックデバイスをネットワーク経由で複製するソフトウェアである．これは単にブロックデバイスの複製であり，本研究が目指しているストレージシステムとはレイヤが異なり，また実現する動作も異なるものであるが，ネットワークを経由したデータ複製の生成という点においてはいくらかの共通点がある．

本稿における提案は，ノードレベルの信頼性を確保する Cluster-wide RAID6 実装についてその最適なデータ転送手法 (書き込み) が主となる．これは，既存の分散ファイルシステムやネットワークデータ・アクセス機構とは目的や実現手段を異にするものであり，またその内容も新しいものである．

3. 前提とするシステムの構成

3.1 概要

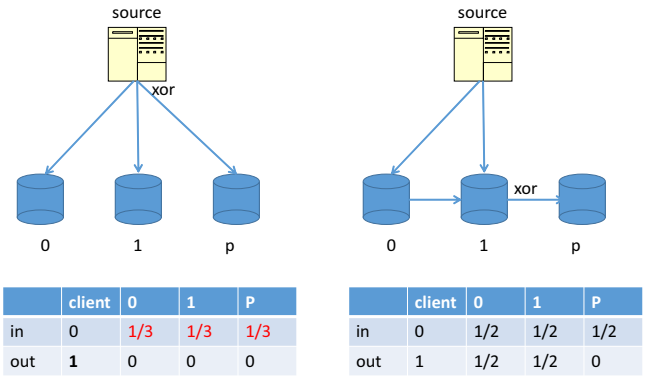
本研究はノード間にまたがる RAID-6 実装について，その最適化された書き込み手法を提案するものである．RAID-4 相当の実装については既に発表しているが，本稿で述べる RAID-6 実装は以前の実装が前提となっているため，そのうち重要な点について概要を説明する．

3.2 RDMA によるデータ転送

書き込まれるデータはすべて，RDMA により転送される．RDMA を利用すると，ネットワークを経由して，リモートコンピュータのメモリを直接読み書きすることが出来る．一般的なソケットインタフェースによってネットワークを利用する場合，ソフトウェアによるプロトコルハンドリングや再送制御など多くの中間レイヤーが存在し，オーバヘッドが大きい．一方で，RDMA を利用すると，直接リモートメモリの読み書きを行うことが出来，それらの制御はハードウェアによって処理されるため，非常に高速である．リモートファイルアクセスにおいて RDMA を活用した場合の性能向上については筆者らが既に発表しており [8]，十分な性能向上が得られることを確認している．

3.3 最適化された Cluster-wide RAID-4 実装

RAID-4 は，1 台のパリティ記録用ノードと，ストライプ化されたデータを記録する複数のノードから構成される．**図 1** の左側に示すように，単純にクライアントがストライプとパリティをそれぞれのストレージノードに送信すると，クライアントが送出するデータ量が元データよりも増えるため，その分だけ性能が低下する．一方で，図中右側



	client	0	1	P
in	0	1/3	1/3	1/3
out	**1**	0	0	0

	client	0	1	P
in	0	1/2	1/2	1/2
out	1	1/2	1/2	0

図 1 xor 計算と転送の最適化 (左：通常　右：最適化後) [3] より引用

に示す手順で転送を行うと，クライアントが送出するデータ量は元データと変わないことからスループットが低下しない．このケースでは，クライアントはストライプのみを各ストレージノードに送出し，ストレージノードは相互にデータをやりとりすることでパリティを生成する．

図中の表は各ノードにおけるデータの送受信量を示している．この表において，1 はクライアントが送出したデータ量を示す．左側では各ノードに 1/3 のデータしか保存されないが，最適化された右側においてはそれぞれ 1/2 となり，50%増えている．したがって，最終的なスループットも 50%向上することが見込まれる．[3] における筆者らの報告でも性能の向上を確認している．

3.4 zero-copy パイプライン処理

従前の節でも述べたように，RDMA を活用するとリモートノードのメモリに直接アクセスが可能となる．この点を活かすことにより，メモリコピーの生じない通信が可能となる．また，データを受信したメモリ上で計算を行い，さらにその領域を別のノードへ転送することも出来る．

本提案における実装は，このような利点を十分に活かすものとなっている．次章で説明する RAID-6 実装は，zero-copy パイプラインの接続により実現されている．

zero-copy パイプライン処理は，3 層のコンポーネントから構成されている．1 つ目はリングバッファで，これはデータの送受信を行うためのメモリ領域である．RDMA のためには予めメモリを InfiniBand のハードウェアに大して登録 (registration) しておく必要があり，ここではこのリングバッファ領域全体を登録している．2 つ目は Peer であり，ノード間の接続を受け持っている．Peer はコネクションを張り，メモリアドレス等のリモートアクセスに必要な情報を交換し，リングバッファの使用領域などの管理を行う．3 つ目は Pipe で，これにはゼロコピー転送を行うものと xor 処理を行うものの 2 つがある．これら 3 層のコンポーネントを接続し，ゼロコピーパイプライン処理を実現する．一度パイプラインを構築すれば，それに基づいて

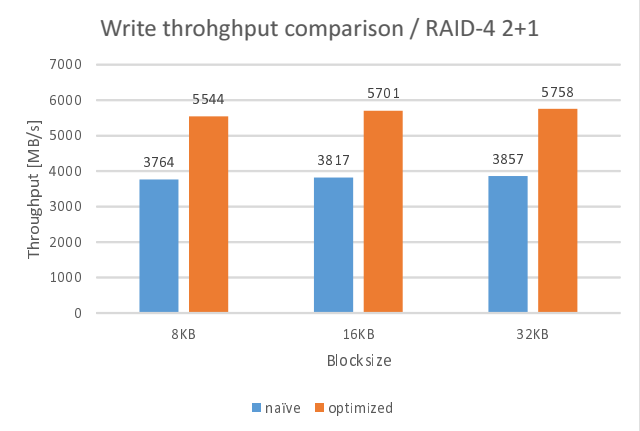


図 2　RAID-4 3 ノート (2+1) の書き込み性能評価

データが流れ，パリティの生成などが行われる．実装の詳細については [3] にて述べているため，ここでは省略する．

3.5　RAID-4 の予備評価

既に RAID-4 実装については性能評価を公開しているが，最新の実装においては性能が向上しているため，そのデータを掲載する (**図 2**)．評価対象は，ストレージノード 3 台に対して，ストライプ (2 つ) とパリティ (1 つ) を書き込みスループットである．図は，クライアントから元データのストライプとパリティをすべて送信する場合 (naive) と，提案手法により最適化して (optimized) 転送した場合の書き込みスループット比較である．横軸はブロックサイズで，クライアントが送出するデータ 1 つあたりのサイズを示している．この場合，2 つのストライプに分割されているため，データの書き込み単位としてはブロックサイズの 2 倍となる．

いずれも，ブロックサイズが大きくなると，データ転送に係る処理の回数が減ることでオーバヘッドも減り，若干の性能向上が見られる．この評価において注目すべき点は，どのケースにおいても 50%近い性能向上を達成していることである．単純にデータの書き込みを行うと，クライアントが送出するデータ量が 1.5 倍になることから，スループットは 33%ほど低下する．ところが，提案手法を用いると転送量増大による性能低下とほぼ同じ分だけ向上が見られる．これは，本提案手法がパリティ書き込みによる性能低下をほぼ完全に回避できていることを意味する．

4.　Cluster-wide RAID-6 の実装

4.1　概要

本研究では，2-d xor 方式の RAID-6 を実装した．この方式については，概略のみ [3] で触れている．RAID-4 や RAID-5 では，ノード間にまたがって分散したストライプについて，それらの xor を別のノードに保存することで 1 台の故障に備えている．2-d xor 方式の RAID-6 は，RAID-5

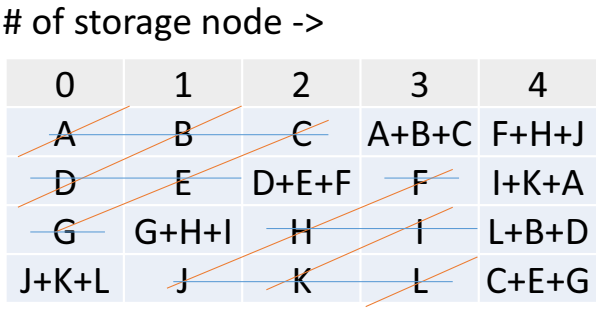


図 3　2-d xor の記録方法　[3] より

に対して斜め方向のパリティを追加した形となっている．この記録方式自体は RAID-6 の実装方法の一つとして用いられる事がある．

図 3 は，1 つのクライアントから 5 台のストレージノードに記録する場合に，データがどのように配置されるかを示したものである．A から L の 12 個のブロックは，書込元から送られてくるストライプである．0 から 3 のストレージノードは，水平方向のパリティをそれぞれ保持する．4 番のストレージノードは斜め方向のパリティを保持する．このような形で xor されたデータを持つことにより，任意の 2 ノードまでの故障に対応できるようになる．

尚，本稿においては，図 3 に示した内容と同様，5 つのストレージノードに対して記録する評価プログラムを作成している．

4.2　最適化の方針

前節に示した RAID-6 において，ストライプ 12 個に対してパリティは 8 個存在している．したがって，書き込み元がすべてのパリティを生成した上でストレージノードに転送すると，送信データ量が 66.7%増加することになる．これはすなわち計算上 40%のスループット低下につながることを意味している．

最適化の基本的な発想は RAID-4 に近い．書込元ノードはストライプデータのみを送出することで，ネットワークトラフィックの増大による性能低下を回避する．また，ストレージノードは相互にデータを交換し，パリティを生成する．これらの処理により，トラフィックの集中と xor 演算負荷の分散が可能になる．

4.3　転送パスの設計

書き込み元から転送されたストライプを元に，ストレージノードは相互にデータ交換をおこないながらパリティを計算し，最終的には所定の場所に格納する必要がある．本稿執筆時において，この転送パターンについてすべての検証および一般化を行えていないため，ここでは条件を満たす 1 つの例について示すに留める．ここに示す各行列はストレージノード間でどこからどこへ，どのようなデータが

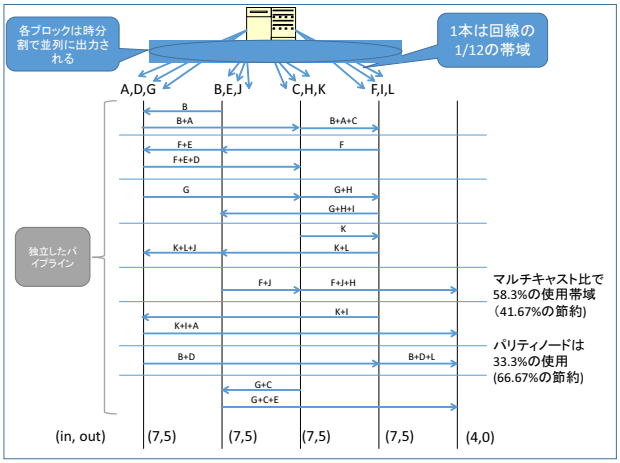


図 4　2-D xor の転送パス例 [3] より

転送されるかを示したものである．行方向は送信元のノード番号で，列方向は宛先のノード番号である．ポイントは (1) で，この転送は横及び斜めのパリティの双方で共通して利用されることである．この事により，各ノードの入出力パスの本数を 1 本ずつ削減することが可能になった．この再利用が可能になる条件は，転送を表す行列のうち，非ゼロ要素の位置が斜め方向を考慮しない N クイーン問題を満たす場合である．したがって，ストレージノードの数が偶数になると，このような再利用は不可能である．(1) から (3) は水平方向のパリティ生成に関わる転送で，(4) と (5) は斜め方向のパリティのものである ((1) は共通)．

これらの転送順序通りにパイプラインを構成すると，図 4 のようになる．

$$\begin{pmatrix} 0 & 0 & G & 0 & 0 \\ B & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & K & 0 \\ 0 & F & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (1)$$

$$\begin{pmatrix} 0 & 0 & B+A & 0 & 0 \\ F+E & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & G+H & 0 \\ 0 & K+L & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (2)$$

$$\begin{pmatrix} 0 & 0 & F+E+D & 0 & 0 \\ K+L+J & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & B+A+C & 0 \\ 0 & G+H+I & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (3)$$

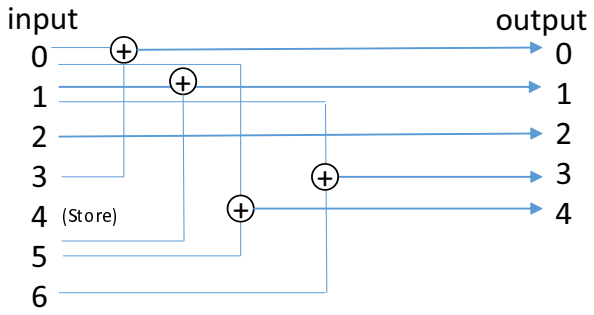


図 5　各ストレージノード内で処理されるデータのパス

表 1　評価環境

CPU	Intel(R) Xeon(R) CPU E5-2665 x2
RAM	64GB
InfiniBand	Mellanox MT27500 4x FDR

$$\begin{pmatrix} 0 & 0 & 0 & B+D & 0 \\ 0 & 0 & F+J & 0 & 0 \\ 0 & G+C & 0 & 0 & 0 \\ K+I & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (4)$$

$$\begin{pmatrix} 0 & 0 & 0 & 0 & K+I+A \\ 0 & 0 & 0 & 0 & G+C+E \\ 0 & 0 & 0 & 0 & F+J+H \\ 0 & 0 & 0 & 0 & B+D+L \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (5)$$

4.4　転送パスの実装

各ストレージノード（パリティ専用の末尾ノードを除く）は，前節までに示された転送パスを構築するため，図 5 に示される処理構造を持っている．図 4 に示されるのと同様，各ノードは 7 本の入力と 5 本の出力を持っている．この構造を，前節に示す転送パスに合致するよう接続すると，演算及び転送機能を備えたパイプラインを構成することができる．

5.　性能評価

5.1　評価条件

性能評価に用いた環境は表 1 の通りである．

前章で述べたものと同様の形で，5 つのストレージノードを使用して RAID-6 記録を行った．ただし，ローカルディスクへの書き込みは，ネットワークが圧倒的に早いことから難しく，本評価においては行っていない．計測はいずれも 3 回行い，結果はその平均値である．

5.2　評価

図 6 は，Cluster-wide RAID-6 に対する書き込みスループットを表したものである．棒グラフは提案手法による実

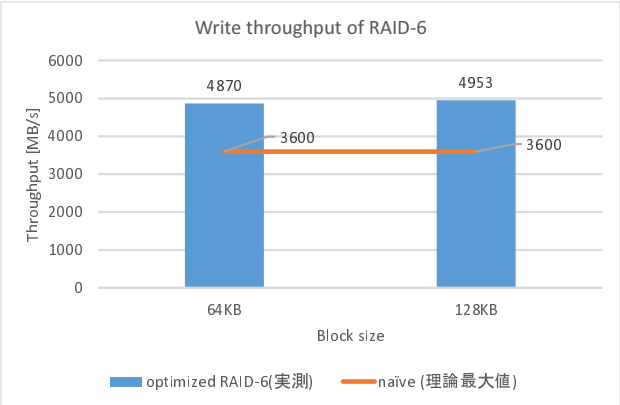


図 6 Cluster-wide RAID-6 に対する書き込みスループット

測性能で，線グラフはクライアント上でパリティの生成を行い，すべてのデータを書き込む場合における，計算上の最大性能を表している．計算上の最大性能は，4.2 において述べたように，ネットワークの 60%のスループットである．

提案手法を用いた場合の実測性能は，用いない場合の最大性能に比べ約 38%の向上が確認された．したがって，本提案手法はナイーブに RAID-6 書き込みを行った場合と比較して，その計算上の性能限界を超える事が出来ており，書き込みスループットを向上する目的を十分に達成できていることがわかる．

6. まとめ

本稿では，ネットワークを利用した Cluter-wide RAID-6 について，トラフィックと xor 演算負荷の集中を回避する方法を提案した．実装にあたっては，すべての転送を RDMA により行い，またデータ処理はゼロコピーを実現した．提案手法を適用した RAID-6 実装を評価した結果，少なくとも 38%の書き込みスループット向上が確認された．

今後の課題としては，転送パス設計を一般化し，異なるノード数にも柔軟に対応できるようにすることや，実装改善によるさらなる性能向上が挙げられる．

謝辞 本研究の一部は，JSPS 科研費 (特別研究員奨励費)26・1967，JST CREST「ポストペタスケールデータインテンシブサイエンスのためのシステムソフトウェア」，JST CREST「EBD：次世代の年ヨッタバイト処理に向けたエクストリームビッグデータの基盤技術」による．

参考文献

[1] Patterson, D. A., Gibson, G. and Katz, R. H.: A Case for Redundant Arrays of Inexpensive Disks (RAID), *SIGMOD Rec.*, Vol. 17, No. 3, pp. 109–116 (1988).
[2] Chervenak, A. L., Foster, I. T., Kesselman, C., Salisbury, C. and Tuecke, S.: The data grid: Towards an architecture for the distributed management and analysis of large scientific datasets, *JOURNAL OF NETWORK AND COMPUTER APPLICATIONSSSS*, Vol. 23, pp. 187–200 (1999).
[3] 大辻弘貴，建部修見：分散ストレージシステムに対する低オーバヘッド冗長化書き込み手法の提案と評価，情報処理学会研究報告ハイパフォーマンスコンピューティング HPC142，pp. 1–6 (2013).
[4] Braam, P. J.: Lustre, *http://www.lustre.org/*.
[5] Weil, S. A., Brandt, S. A., Miller, E. L., Long, D. D. E. and Maltzahn, C.: Ceph: A Scalable, High-performance Distributed File System, *Proceedings of the 7th Symposium on Operating Systems Design and Implementation*, OSDI '06, Berkeley, CA, USA, USENIX Association, pp. 307–320 (2006).
[6] Tatebe, O., Hiraga, K. and Sod, N.: New Generation Computing, Ohmsha, Ltd. and Springer, *Gfarm Grid File System*, Vol. 28, No. 3, pp. 257–275 (2010).
[7] Reisner, P.: DRBD v8 Replicated Storage with Shared Disk Semantics, *Proceedings of the 12th International Linux System Technology Conference* (2005).
[8] 大辻弘貴，建部修見：Infiniband を用いた遠隔ファイルアクセスの高速化，情報処理学会研究報告ハイパフォーマンスコンピューティング HPC135，pp. 1–6 (2012).