

リアルタイム圧縮技術

— データをより効率的に保管しコストをセーブ —

日々増え続けるデータをいかに効率的に保管するかは、今の IT 業界における大きな課題となっています。この課題の解決策の 1 つとしてデータの圧縮技術があります。IBM は従来のデータ圧縮技術に独自の技術を加えたリアルタイム圧縮を実装することにより Network Attached Storage（以下、NAS）や Storage Area Network（以下、SAN）に接続するストレージにおいて、他社にはない優れたデータ圧縮を実現しています。本稿では、IBM が実装したリアルタイム圧縮の技術を解説するとともに、その技術が IT 環境にもたらす意味について解説します。

① はじめに

古いファイルをなかなか捨てられない、こんな経験は誰にでもあると思います。例えば、「3カ月前に作成した資料を消してしまってもいいのだろうか?」「ドラフト・レベルのものは消してしまってもよいのではないだろうか?」と悩むことがあります。しかし、何かの時に使うかもしれないという気持ちから、結局は捨てられずに PC のハードディスクの中に眠ってしまうことがあるのではないのでしょうか。

企業の視点から見ると、年々増加する情報量に対しての情報をいつまで保管すべきか、ビッグデータを見据えて一定期間保管する必要があるのではないかという課題に言い換えられます。データに対する明確な優先順位と保管のポリシーがないと、データはどんどんたまる一方になります。

その課題を解決するための方策は幾つありますが、従来はデータの増加に伴いディスクを増設することが多かったのではないのでしょうか。過去 10 年のストレージ技術の進化は目覚しく、記録メディアの密度が高くなることでストレージのビット単価が下がり、より大容量のストレージを調達できるようになりました。しかし、今後 10 年で既存のストレージ技術は飽和状態になり、今までのようにストレージの容量増加ばかりには頼れなくなるとわれています [1]。そのような状況を見据えた上での解決策の 1 つが、データ圧縮技術の活用です。データが急増しストレージ容量の増加がそれに追いついていかない状況において、増え続けるデータを効率的に保管する方法の 1 つとして注目を集めています。

本稿では、IBM のストレージ製品に搭載されて

いるリアルタイム圧縮技術を紹介し、それが今後の IT システムにおいてどのような効果をもたらすかを解説します。

② IBMのリアルタイム圧縮技術

2010 年 12 月、日本アイ・ビー・エム株式会社は IBM Real-time Compression Appliance（以下、RtCA）を発表しました*1。RtCA は、NAS の 1 次アクティブ・データを、パフォーマンスの低下を最小限に抑えつつリアルタイムで圧縮することによって、ストレージの使用効率を向上させる製品です。

RtCA は NAS に接続する形で活用され、その構成

*1 2013 年 2 月に新製品として性能を向上させた IBM Real-time Compression Appliance STN7800 を発表 [2]。

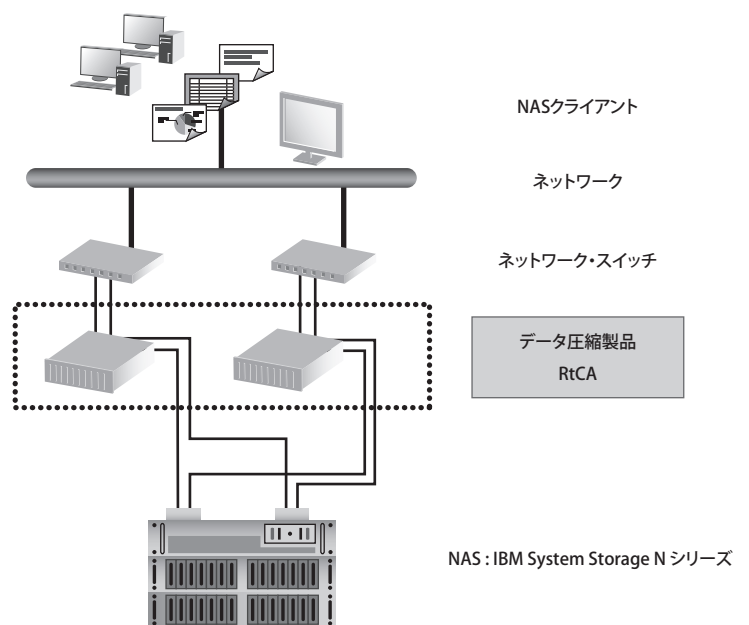


図 1. RtCA 構成例

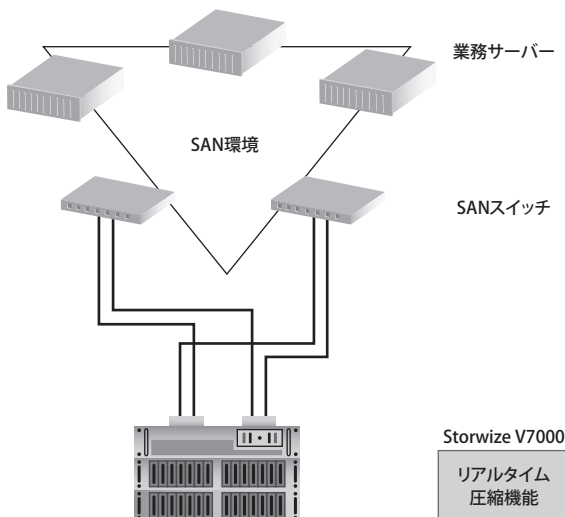


図 2. Storwize V7000 構成例

例は図 1 のようになります。

RtCA を既存のネットワークと NAS の間に挟み込む形で接続することにより、既存の NAS 環境、特に NAS 周辺のネットワーク構成に大きな変更を加えることなく追加できることが特長となっています。

さらに2012年6月にはIBM Storwize V7000(以下、Storwize V7000)とIBM System Storage SAN ボリューム・コントローラー(以下、SVC)にこの技術を実装することで、SAN 接続のブロック・ストレージ環境におけるデータのリアルタイム圧縮も実現しました。Storwize V7000とSVCには、製品自体にリアルタイム圧縮技術が実装されており、NAS 環境でのアプライアンスと同等の圧縮効果が得られます。また、図 2 で示すように今までと同じようにSANを構成することが可能です。

③ データ圧縮の基本技術

具体的にIBMのリアルタイム圧縮技術に関して解説していきましょう。まずはおなじみのZIPを例に挙げて説明していきます。Windowsなどの作業環境においてファイル、もしくは複数のファイルを圧縮するためにZIPの機能を使用することは多いでしょう。ZIPは複数のファイルを1つのファ

イルとしてアーカイブするためのファイル・フォーマットで、圧縮は各ファイルをZIP形式で格納する時に実行されます。ZIPは一般的にDeflateというデータ圧縮アルゴリズムを使っており、IETF RFC 1951にその機能の仕様が記載されています [3]。

Deflateでは64Kバイトを超えない任意のサイズのデータ・ブロックを単位として圧縮が実行されます。それぞれのブロックでは、LZ77(Lempel-Ziv 77)を改良したLZSSというデータ圧縮のアルゴリズムとHuffman符号の両方を組み合わせてデータが圧縮されます。簡単に言えば、LZSSのアルゴリズムを用いて各ブロック内で同じ文字列の繰り返しを見つけ出して、繰り返しが見つかった文字へのポインター(長さ、繰り返しが見つかった位置)、もしくは文字コードの2種類に符号化します。その符号化された結果をさらにHuffman符号で符号化することによってデータを2段階で圧縮します(図3)。

④ IBM Random Access Compression Engine

前述した圧縮方式でファイルを圧縮しZIPフォーマットで自分のPCに保管する分には、さほど使い勝手に問題は感じないと思いますが、同様の圧縮方式をNASやSAN接続の外部ストレージで実装する場合ではどうでしょうか。例えば、圧縮されたデータの1部分を読み出したとします。その場合、圧縮方式の特性から、該当する1部分のデータのみならず、全体を復号する必要性が出

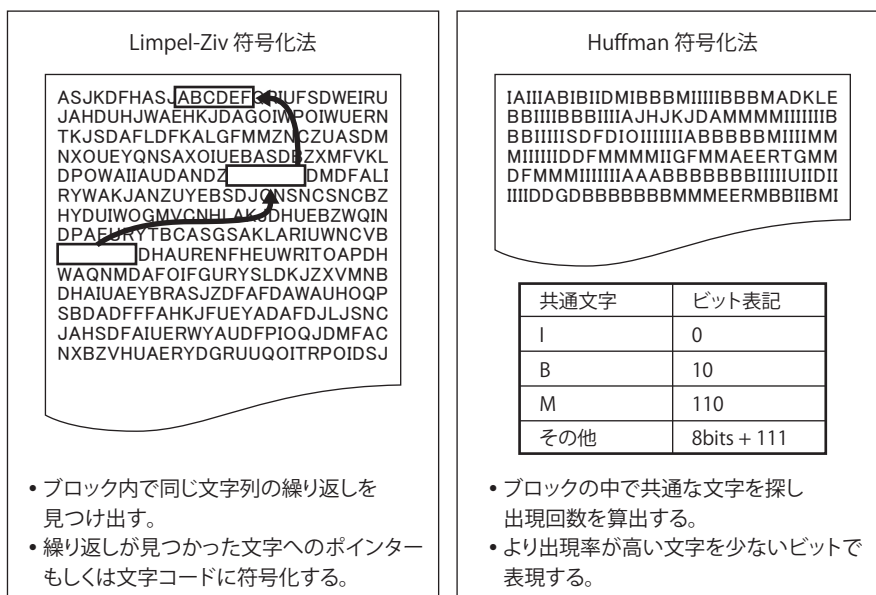
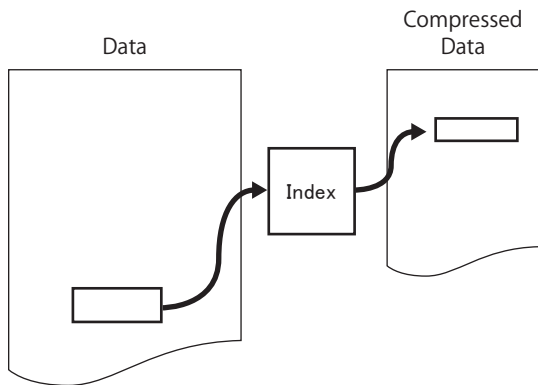


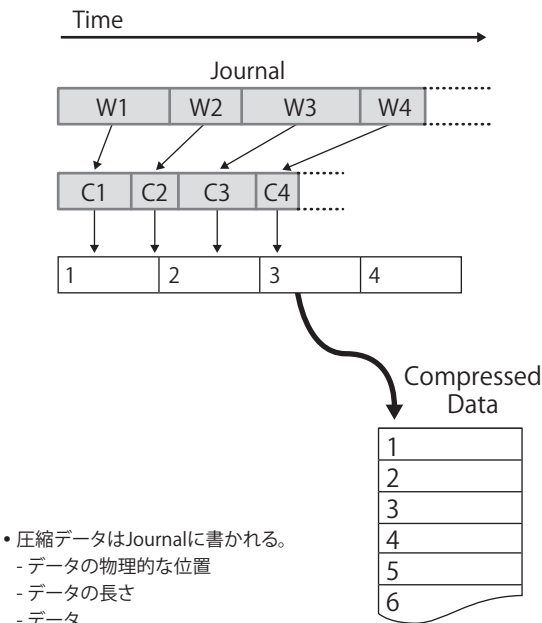
図 3. データ圧縮基本技術の例

できます。同様にデータのある部分に書き込む場合、全体を再度、頭から圧縮しなければならず、結果としてサイズの大きいファイルの読み書きが必要になるとともに、ブロック・アクセスのストレージでは、圧縮・非圧縮処理時



- 元のデータは圧縮データ内の対象データの位置にマップされる。
- このマップを書き換えることで、効率的な圧縮データの更新が可能。
- このIndexによるマップを階層構造で保持しておくことで圧縮データへの高速アクセスと効率的なメモリー使用を実現。
- 少ないI/Oのオーバーヘッドでマップの更新を実装。

図 4. Indexing の概念



- 圧縮データはJournalに書かれる。
 - データの物理的な位置
 - データの長さ
 - データ
 - Journalに書かれる各エントリーは圧縮される。
 - 圧縮データは固定長のブロックとして保持される。
- ▶このJournal機能によりTemporal Compressionを実現
- ▶圧縮されたデータを書く際に、事前のデータの読み込みは必要ない。
- ▶書き込むデータ量が少なくて済む。

図 5. Journaling の概念

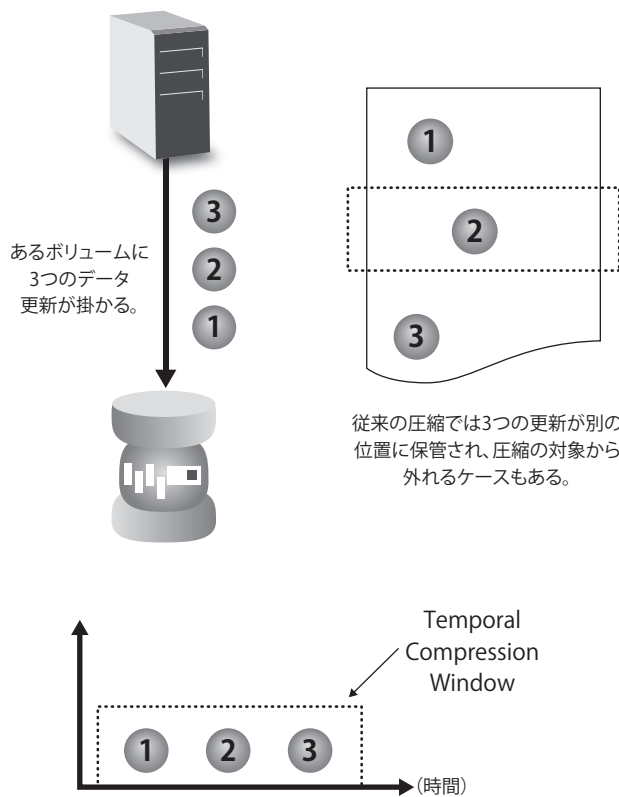
のディスクの I/O や CPU の負荷が高まりパフォーマンスが低下することになります。これが今までストレージ装置に圧縮技術を実装できなかった大きな原因です。

IBM はこの課題を IBM Random Access Compression Engine (以下、RACE) というテクノロジーによって解決しました。RACE は既存のデータ圧縮技術をベースに 35 の特許申請済みの技術を駆使して、以下の 2 点を実現しています。

- 圧縮データへの高速なランダム・アクセス
- パフォーマンスへの影響を与えないリアルタイム圧縮

その技術の特長として、圧縮データの Indexing と Journaling、Temporal Compression を説明します。

- RACEは対象データの物理的な位置ではなく、同時期に書かれたデータに対して実行される。
- RACEは同時期に書かれたデータを処理するため、データの構造と関係性を保持して圧縮する。これは圧縮を解く際に優位に働く。
- 同時に書かれたデータはより関連性が深く、高い圧縮率につながり、さらにはデータの先読みにも寄与する。



従来の圧縮では3つの更新が別の位置に保管され、圧縮の対象から外れるケースもある。

この方式では、3つの更新が時間的にもアプリケーションから見た意味的にも関連付けられる。

図 6. Temporal Compression の概念

Indexing の機能では、図 4 に示すように圧縮前のデータと圧縮されたデータを Index によってひも付けします。この Index を階層化して管理することによって、圧縮されたデータへのランダム・アクセスの速度向上に貢献します。また、圧縮されたデータを更新する際にも Index のひも付けを変更することで、それ以降のすべてのデータを再度圧縮することなく効率的に更新することを実現しています。

Journaling は、図 5 で示すように、データの物理的な位置、長さ、データ自身を、圧縮を実行する際に記録していく手法です。一般的なデータ圧縮においては、固定長（例えば 32KB）のデータを 1 ブロックとして圧縮するため、ブロックによって圧縮の度合いが異なり、圧縮後のデータの長さは可変になってしまいます。一方、RACE ではデータ圧縮の際に可変長のデータを取り込みつつ固定長のデータのブロックに調整し、その圧縮処理の内容を記録していきます。この機能によって、固定長のデータを圧縮するよりも高い圧縮率が期待でき、圧縮後のデータへアクセスする単位が固定長になるのでディスクへのアクセスが少なくなり、安定したパフォーマンスを保つことができます。またデータ圧縮の記録を保持する（この情報も圧縮される）ことにより、圧縮データを更新する際にも事前にデータを読み込む処理の必要性が軽減されます。

Temporal Compression は、直訳すれば時間的な圧縮ということになります。従来の一般的なデータ圧縮では、ディスクに保管されたデータの位置をベースに圧縮していくのに対し、Temporal Compression は、データの保管位置ではなく、接続されているサーバーから同時期に書き込まれるデータを圧縮していく手法で、言い換えれば、サーバーから同時期に送られるデータを時間軸でグループ化し、圧縮を行います。結果として、それらのデータが同一ブロックとしてディスクに保管されます（図 6）。この手法の利点としては、実際のアプリケーションから送られるディスクへの I/O の要求が圧縮される単位に則しているため、同じデータの出現率が高く圧縮率も大きくなることが挙げられます。また、圧縮されたデータを読む際にも先読みがしやすくなるので、より良いパフォーマンスを得ることができます。

5 リアルタイム=即時性の持つ意味

このように IBM のリアルタイム圧縮は、

RACE の技術を用いることにより、大量のデータに対してランダムなアクセスが発生する外部ストレージにおけるデータ圧縮を実現しています。ここで重要なことはサーバーもしくはクライアントから書き込まれるデータをリアルタイムに圧縮することです。言い換えれば、ディスクに書かれるデータを、実際にディスクに書かれる前に圧縮してからディスクに保管する、ということです。これはストレージのユーザーにとっては大きな意味を持ちます。

リアルタイム圧縮の技術を用いた場合、1GB のデータを半分の 500MB に「ディスクに保管する前に」圧縮できるとすれば、そのデータのために準備するストレージの容量は従来の半分で済むことになります。

では、従来通りにデータがディスクに保管されてからデータ圧縮が実行されるとしたらどうでしょうか。まずは、1G バイトのデータを書き込む領域と、それを圧縮した後のデータ（この場合 500M バイト）のディスク容量が必要になります。単純に考えれば、従来のデータ圧縮技術では、元のデータ+圧縮後のデータの領域がディスクに一時的に求められることとなります。さらに、圧縮を実行する際には、保管されたデータをフルに読みながら圧縮して書き戻す動作が必要になるので、容量的にも I/O 的にもディスクに負荷が掛かることとなります。

6 リアルタイム圧縮の I/O への影響度

IBM の製品はリアルタイムな圧縮を実装することでこれらの課題を解消しているわけですが、リアルタイム圧縮に

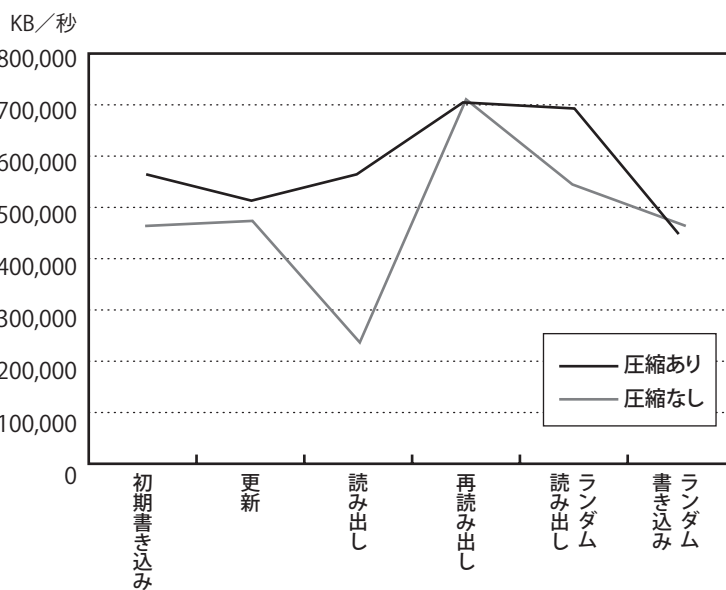


図 7. IOzone によるパフォーマンス・ベンチマーク

よるディスクの I/O パフォーマンスへの影響はどの程度なのでしょう。先にご紹介した RtCA は NAS 用の外付けアプライアンスとして実装することにより、CPU やメモリーなど NAS のリソースを使用しないリアルタイム圧縮を実現しています。また、RtCA はそれ自身にもキャッシュを持っているので、例えばデータを Read する場合、すでにデータがキャッシュにあればレスポンスは早くなります。これは一般的なストレージのキャッシュと同じ効果があります。また、NAS のディスクに読み書きするデータは圧縮後のデータであるため、ディスクへの I/O が少なくなることもパフォーマンス向上に貢献しています。

RtCA と NAS を使用してデータの書き込み、圧縮を実行した際のパフォーマンスについて、IOZone というツールを用いてベンチマーク・テストを行った結果を見ても、リアルタイム圧縮による I/O パフォーマンスへの影響はほとん

ど見られませんでした。それどころか圧縮なしの NAS の構成よりも圧縮を行った方が優位なケースがあることが分かります (図 7) [3]。

しかし、SAN 接続のブロック・ストレージ環境においては、少し状況が異なります。Storewize V7000 と SVC では RACE を用いたリアルタイム圧縮が同一筐体内に実装されているため、リアルタイム圧縮の機能を選択し実行すると、CPU に一定の負荷が掛かります (リアルタイム圧縮用に CPU が割り当てられます)。また、データのアクセスのパターンによってもパフォーマンスやデータの圧縮率に差が出てくるので、事前にどのようなデータを読み書きするのか、どのようなパターンでアクセスするのかを検討した方がよいかもしれません。

7 リアルタイム圧縮の効果

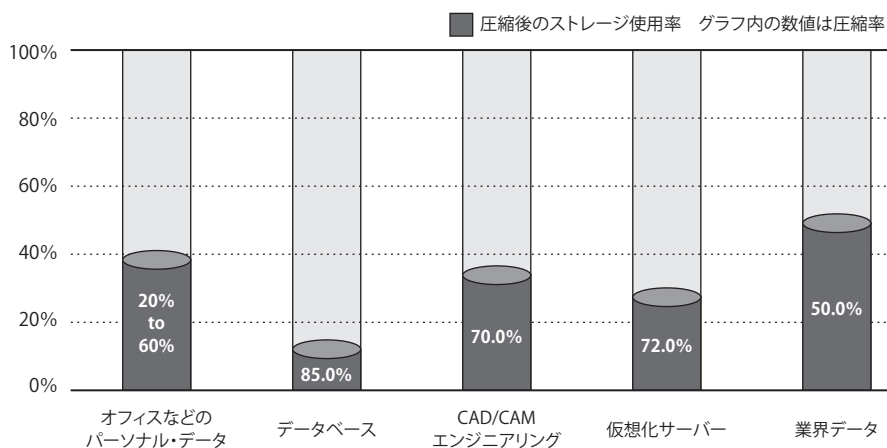


図 8. リアルタイム圧縮の効果

では、リアルタイム圧縮を用いると、実際にデータはどの程度圧縮されるのでしょうか。Windows PC 上で ZIP などを使用してファイルの圧縮の経験がある方は、データの種類によって圧縮率が大きく異なることを認識されていると思います。例えば、テキスト・ファイルや、Excel ファイルなどはかなり大きい圧縮率が期待されます。逆にすでに圧縮されているイメージや動画などのファイルの圧縮率はほとんど期待できません。

表 1. サンプル・ファイルの圧縮率

ファイル名	拡張子	圧縮前のサイズ (KB)	圧縮後のサイズ (KB)	圧縮率 (%)
xxxxxxxxxx.doc	doc	74.0	34.4	53.5
xxxxxxxxxx.pdf	pdf	118.2	112.0	5.3
xxxxxxxxxx.pdf	pdf	194.8	185.3	4.9
xxxxxxxxxx.xls	xls	4049.0	900.5	77.8
xxxxxxxxxx.xls	xls	2382.5	500.6	79.0
xxxxxxxxxx.xls	xls	1144.5	233.7	79.6
xxxxxxxxxx.xls	xls	2954.5	780.5	73.6
xxxxxxxxxx.xls	xls	3460.0	900.5	74.0
xxxxxxxxxx.pdf	pdf	398.6	363.0	8.9
xxxxxxxxxx.pdf	pdf	667.5	534.8	19.9
xxxxxxxxxx.doc	doc	141.0	98.5	30.1
xxxxxxxxxx.doc	doc	102.0	56.7	44.4
xxxxxxxxxx.xls	xls	60.5	27.9	53.8
xxxxxxxxxx.pdf	pdf	622.4	542.7	12.8

前述の通り、IBM のリアルタイム圧縮においてもベースとなる圧縮技術は同じなのでこの傾向は変わりません。図 8 に RtCA を用いてさまざまなデータを書き込んだ際の圧縮率を示します。

これを見て分かる通り、今の IT システムにおいて欠かせないデータ・タイプについては、50 ~ 70% の圧縮率が期待できます。ここで注目すべき

は仮想化サーバー (VMWare) のデータの圧縮率で、VMDK (Virtual Machine Disk: 仮想マシン・ディスク) のファイルでは72%もの圧縮の効果が期待できます。ここでは、一次ディスクの圧縮効果のみを示していますが、それ以外にもそれをバックアップする際のストレージ容量とバックアップにかかる時間も大きく改善することができます。さらには、バックアップしたデータをリストアする時間も短縮できるので、万が一仮想マシンの復旧が必要になった場合においても、復旧時間の短縮に大きく貢献することになります。

図8の結果はIBMの社内で測定したもの、もしくは理論値ではないかと疑われる方もいらっしゃるかもしれませんが、実際のお客様環境における例をご紹介します。表1は、RtCA STN6500を導入され、本号のインタビューにもご登場いただいている日本写真印刷株式会社様の導入検討の際に検証用に提供いただいたサンプル・ファイルの圧縮結果です。

これらのファイルは、どの企業でもNASなどのファイルサーバーに存在する一般的なビジネス文書ですが、検証の結果、平均して50%程度の圧縮率が期待できることが分かりました。データを50%に圧縮できるということは、現在使用しているNASのHDD数を半分にできることを意味します。HDD数を減らせるということはディスクのエンクロージャーの数も減らすことができ、ひいてはディスク増設に伴うフロアー・コストやストレージが消費する電力、機器の発熱に対するクーリング・コストの軽減にもつながります。具体的な日本写真印刷株式会社様におけるRtCAの導入効果に関しては、本誌8ページ以下のインタビュー①をご覧ください。

データ圧縮の効果を少し違った視点で考えると、プライマリーなストレージでデータを圧縮しておけば、バックアップ用途などで遠隔地に転送するデータ量も小さくできるため、データの転送時間の短縮やネットワーク回線のコストも抑えられるなど、データの遠隔地保管が容易になることが期待できます。

8 より効率的なストレージの利用へ

本稿では、リアルタイム圧縮の技術はどのような特長、優位性があり、具体的にストレージのシステム環境においてどのような意味を持つのかを解説してきました。この技術を有効に活用することにより、ディスクの物理的な使用量を抑えることができ、ストレージにかかわる機器やリソ-

ス、コストを軽減することが可能だということをお分かりいただけたと思います。

ストレージを効率的に使用する手法は今回解説したデータ圧縮の技術だけではありません。例えば、データの重複した部分を削減する重複削減技術やディスクの容量を仮想的に大きく見せるシン・プロビジョニング、アクセス頻度の低いデータをより安価なストレージに保管するストレージ階層化の技術などがあります。ここで重要なのは、リアルタイム圧縮の技術は、IBMが提供するそれらの技術と組み合わせて使用できるように設計・実装されているという事実です。お客様のシステム、データの特徴を見極めてこれらの技術を選択的に有効活用していくことで、より効率的なストレージ環境が実現できるのです。

【参考文献】

- [1] 佐野正和:IBM Smarter Storage ストレージに関するビジョンと展望, P6, [http://www.ibm.com/jp/domino04/pc/support/Sylphd08.nsf/1e97b730bd4fa8f249256a840020d047/4c113941ad5b7d8049257b2c00089beb/\\$FILE/System%20x%E9%83%A8%28%E7%94%9F!%29%20%E7%AC%AC%E5%8D%81%E4%B9%9D%E5%9B%9E%E8%B3%87%E6%96%99.pdf](http://www.ibm.com/jp/domino04/pc/support/Sylphd08.nsf/1e97b730bd4fa8f249256a840020d047/4c113941ad5b7d8049257b2c00089beb/$FILE/System%20x%E9%83%A8%28%E7%94%9F!%29%20%E7%AC%AC%E5%8D%81%E4%B9%9D%E5%9B%9E%E8%B3%87%E6%96%99.pdf)
- [2] IBM: IBM Real-time Compression Appliance STN7800の発表, 日本 IBM ハードウェア発表レター JG13-0017-, http://www.ibm.com/common/ssi/rep_ca/7/760/JAJPJG13-0017/JAJPJG13-0017.PDF
- [3] IETF: DEFLATE Compressed Data Format Specification version1.3, <http://www.ietf.org/rfc/rfc1951.txt>
- [4] EDISON Group: A Validation and Comparison of Storage Efficiency Technology: Compression, <http://public.dhe.ibm.com/common/ssi/ecm/en/tsw03105usen/TSW03105USEN.PDF>



日本アイ・ビー・エム株式会社
システム・テクノロジー開発製造
システムズ・ラボサービス&サポート
シニア・プログラマー

尾関 和弘 Kazuhiro Ozeki

【プロフィール】

1985年日本IBM入社。大和開発研究所でホスト・コンピュータ端末の開発に携わる。その後、LTOテープ装置のファームウェアの開発リーダーとして、アメリカIBM ツーソン研究所との共同開発に従事。2009年からIBMのストレージの先端技術の導入を支援するストレージLab サービスの一員となり、お客様のストレージ環境の構築を支援している。