Cut memory overheads, CO_2, computing time and data storage costs with COBOL data compression.

Executive Summary

Businesses grow and compliance legislation increases, yet IT budgets never quite seem to keep up. One way to minimize computer overheads and save CO_2 emmitions, is to reduce the volume of data that needs to be processed and stored.

Data compression has been around for a long time but it's almost always offered at the file or volume level. File level compression is fine if you're unlikely to ever want to recover the data again but what if you need an item of information in a huge compressed archive? How would you even find it?

Also, if users discover that they may have to wait days to access archived information, they won't let you archive it. Why should they suffer inconvenience, just to save the IT department some cash?

Given that "*75% of the world's business data is still processed on mainframe applications written in COBOL*." (Gartner Group) and "*Mainframe application workloads have grown 19 percent annually for the past 5 years*." (IBM), archiving and compression is particularly acute in the mainframe COBOL environment.

What's needed is a simple field level compression routine that:

- Keeps data accessible to real-time applications.
- Easy to compress.
- Very fast to decompress.
- Designed for COBOL data formats (COBOL data often contains binary strings which are difficult to compress).

The Redvers Compression Algorithm

Redvers Consulting have engineered a compression / decompression algorithm, specifically designed to produce optimal compression rates for data held in COBOL format.

Data strings selected for compression can be in the form of single fields, parts of a record, complete records or even a file of associated records stacked end-to-end.

Key accessed files can be compressed while continuing to operate as if they were still uncompressed, by excluding the key of each record from compression. This allows indexed file acces to continue as before, followed by a decompression process for the data content of each record.

The algorithm is designed to use minimal computer resources, especially when decompressing, so compressed information can be retrieved on-line with minimal disruption. In fact, for the majority of cases, compressed data retrieval will actually be faster than uncompressed retrieval due to the reduced disk rotation required to deliver the same information.

On-line transactions can also benefit by compressing the data stored in memory, thus allowing more simultaneous tasks.

Probably the most desirable effects of data compression are reduced hardware and electricity bills combined with a reduction in CO_2 emittions.

Key benefits:

- Money saved on DASD hardware
- Fast decompression rate for real-time retrieval
- Reduced machine memory overhead
- Faster disk and tape I-O
- Lower electricity bills
- Reduced carbon footprint
- Compression rates up to 75%
- Runs on any COBOL platform
- Also works with non-COBOL data
- Runs in batch or on-line modes

How it Works

Compression can be performed in a one-off batch procedure that selects the data suitable for compression. This data is passed to the Redvers compression routine (**RCCMPRES**) which returns the string in its compressed form. The compressed string can then safely replace the original data.
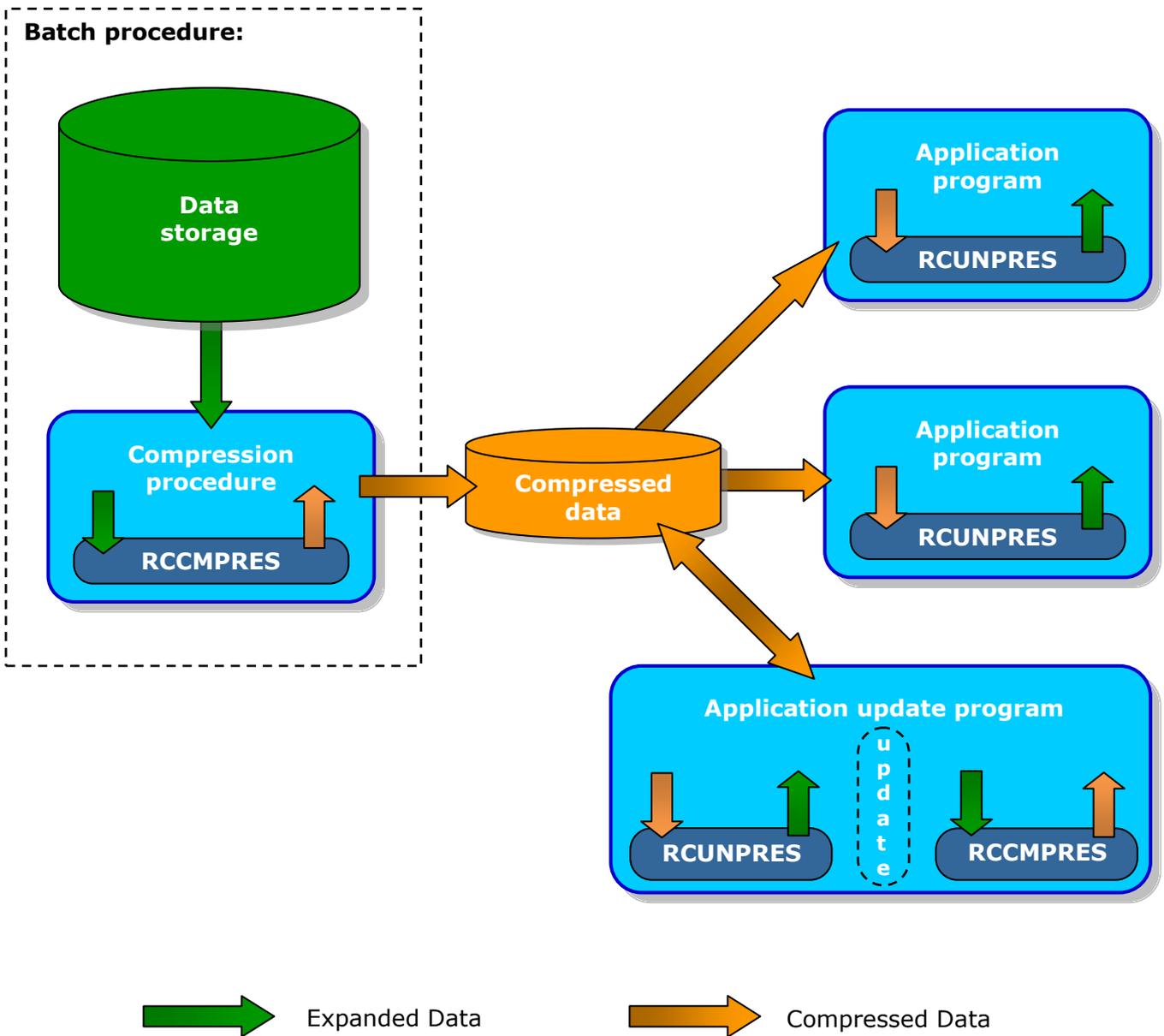
Application programs that require the compressed information, pass the compressed string to the decompression routine (**RCUNPRES**) which returns the data string in its normal form. No other files, keys or parameters are required.

If an application needs to update compressed information, the updated details are passed through **RCCMPRES** and the output rewritten to the compressed data store.

Disk space in databases and indexed files can be saved by leaving key information unchanged, rewriting only the data components in their compressed state.

Compressed data can always be recovered because there are no keys to loose.

The diagram below shows how a compression / decompression procedure might be used in a typical application environment.

Batch procedure:

Data storage

Compression procedure

RCCMPRES

Compressed data

Application program

RCUNPRES

Application program

RCUNPRES

Application update program

RCUNPRES

update

RCCMPRES

Expanded Data

Compressed Data

Reducing CO_2

On April 15th 2009 Redvers Consulting conducted two COBOL application simulations at the IBM Innovation Centre in Hursley Park, Winchester, UK. The simulations ran on a 2-way Power 5+, 16GB System i server running IBM i 6.1.

The first simulation was to prove the **Redvers Compression Algorithm** could be used to reduce the amount of memory required to hold one or more instances of a string of 1,214 bytes.

The software also had to restore the compressed string back to its original form and confirm an exact match.

The results showed a memory saving of 751 bytes – a compression ratio of 62%.
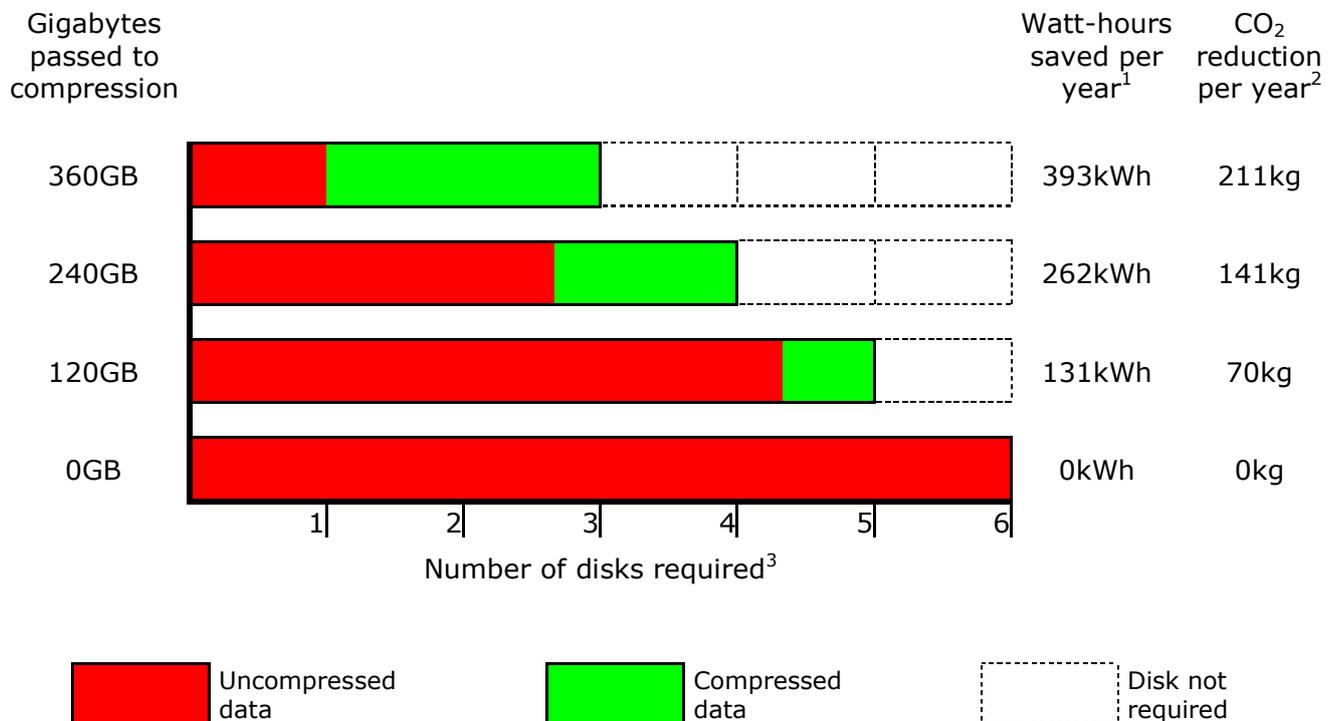
The second scenario simulated the batch compression and decompression of a client master file consisting of one thousand, 1,212 byte records, each with a 16 byte key.

To allow for easy access to the compressed data, this simulation did not compress the record keys. Only the data portion of each record was compressed and a two byte length field was added after each key, providing a length for the compressed area.

Finally, the compressed master file was restored to its expanded state and checked against the original file.

These results proved that a 1.2MB master file can be converted to a 0.5MB file with equal accessibility and data content – a compression ratio of 60%.

To see how 60% compression can produce real power and CO_2 savings by reducing the number of internal disks required to store 438GB on an iSeries installation, see below:
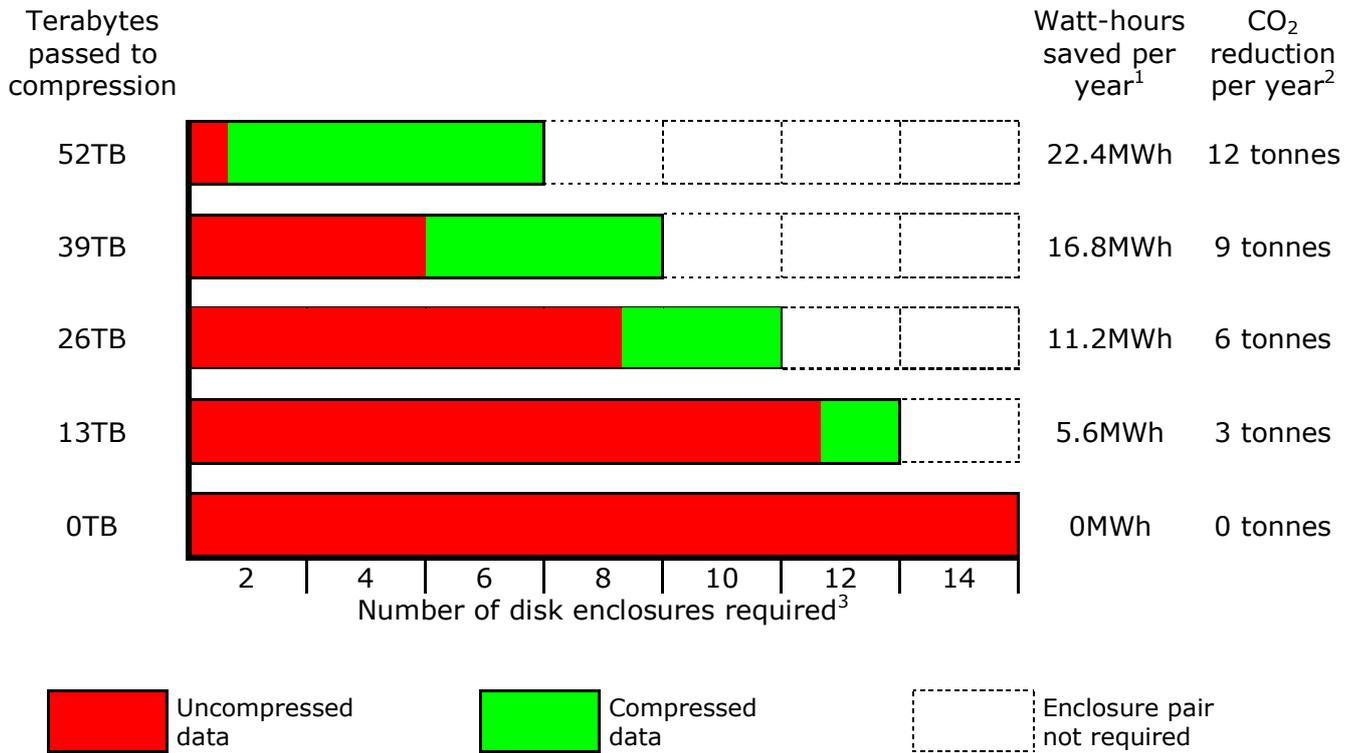
Gigabytes passed to compression	Number of disks required[3]	Watt-hours saved per year[1]	CO_2 reduction per year[2]
360GB		393kWh	211kg
240GB		262kWh	141kg
120GB		131kWh	70kg
0GB		0kWh	0kg

Number of disks required[3]: 1 2 3 4 5 6

Legend:
- Uncompressed data (red)
- Compressed data (green)
- Disk not required (dashed)

[1] 15 additional watts per disk: http://www-947.ibm.com/systems/support/tools/estimator/energy/
[2] 0.537kg per kilowatt-hour: http://www.carbontrust.co.uk/resource/conversion_factors/
[3] IBM iSeries model 520-8203-E4A 73GB Hard Disk Drives (SAS, 15 KRPM)

To see how 60% compression can produce real power and CO_2 savings by reducing the number of DS8000 disk enclosures required to store 54.6TB, see below:

Terabytes passed to compression	(chart)	Watt-hours saved per year[1]	CO_2 reduction per year[2]
52TB		22.4MWh	12 tonnes
39TB		16.8MWh	9 tonnes
26TB		11.2MWh	6 tonnes
13TB		5.6MWh	3 tonnes
0TB		0MWh	0 tonnes

Number of disk enclosures required[3]
(x-axis: 2 4 6 8 10 12 14)

Legend: Uncompressed data | Compressed data | Enclosure pair not required

[1] 640 additional watts per disk enclosure pair (270W x 2) + (25W x 4) – see "Power consumption and environment" section of IBM System Storage DS8000: Architecture and Implementation

[2] 0.537Kg per kilowatt-hour: http://www.carbontrust.co.uk/resource/conversion_factors/

[3] Containing 32 x 15 KRPM, 300GB disks running 7 + P RAID 5 configuration (7.8TB data capacity)

The Product Package

A perpetual license for the **Redvers Compression Algorithm** can be provided for a one-off fee. Alternatively, the software can be leased on an annual basis for 20% of the perpetual license cost (minimum two years).

All licenses include:

- Product source code ("cloaked")
- Sample COBOL calling program
- User Guide
- Corporate level software license
- Two year warranty
- Product upgrades and support via email*

Additional options:

- 24 x 7 telephone hotline support
- Software escrow agreement with Software Escrow Solutions

Software and documents are shipped in the form of email attachments unless otherwise requested. Installation is performed by copying the source code text into your COBOL source code library and running your standard site compiler.

Full pricing details can be found at:
http://www.redversconsulting.com/data_compression_pricing.php

* Free for the first two years followed by a minimal annual fee.

About Redvers Consulting

Redvers Consulting have been providing top quality products and services for COBOL applications since 1988. Our clients are primarily large financial institutions in Europe and North America, although we also have customers in many other business and geographical areas.

Our ability to deliver software in COBOL source code form, gives customers reliable, efficient and perfectly integrated solutions to business needs. Source code distribution also means our software will run on all hardware platforms and operating systems: *EBCDIC, ASCII, big endian or little endian*.

We are business partners with **IBM**, **HP** and **Fujitsu Siemens**, and our development team are members of the **Professional Contractors Group**. In 2009 we won the Thames Gateway **Best use of Technology Award**.

Our client list includes:

Agora (FR)
ANZ (AUS)
Barclays Life Assurance (UK)
Canada Life Assurance (UK)
Deutsche Bank (USA)
Deutsche Rentenversicherung Bund (DE)
FirstBank (USA)
Fiserv (USA)
GMAC Insurance (USA)
Hanesbrands (USA)
John Deere (USA)
LBS / Finanz Informatik (DE)
J P Morgan (USA)
Oppenheimer (USA)
Pacific Gas (USA)
Network Rail (UK)
R+V Allgemeine Versicherung (DE)
Sasktel (CAN)
SEB (DE)
Standard Life Assurance (UK)
Suncorp (AUS)
SunGard / FIS (USA)
WorkSafeBC (CAN)
Zurich Insurance (UK & SUI)

Contact: https://www.redversconsulting.com/contact.php

Development Office:

Redvers Consulting Ltd
44 Broadway,
London E15 1XH,
UK

Tel: +44 (0)203 130 0773
Fax: +44 (0)700 603 8655

Accounts Office:

Redvers Consulting Ltd
1st Floor, 48 Dangan Rd,
London E11 2RF,
UK

Tel: +44 (0)870 922 0633
Fax: +44 (0)707 505 5472

German Office:

Redvers Consulting Ltd
Postfach 30 03 26,
50773 Köln,
Deutschland

Tel: +49 (0)221 1704 9000
Fax: +49 (0)221 271 1016