Integrated Accelerator for zEnterprise Data Compression

Explosive amount of data
According to industry experts, data volumes are predicted to reach 163 zettabytes—or 163 trillion GB—by 2025[1]. Data is the lifeblood of every organization. It doesn't matter how large or how small an organization is today, data and the ability to access growing volumes of data is critical to running the business.

Data needs to be kept online for it to carry the most value and it needs to be sent from one system to another – often large data files. High speed compression is needed to help keep more data online and improve the speed the exchange of these cross-platform files.

The zEDC solution
In July 2013, IBM introduced IBM zEnterprise® Data Compression (zEDC) in z/OS® 2.1 along with the IBM zEDC Express adapter for hardware. The combination of these two offer a compression acceleration solution designed for high performance, low latency compression with few processor cycles and little additional overhead.

New Integrated Acceleration for zEDC
IBM introduced a new on-chip accelerator for the IBM z15™, to replace the zEDC Express adapter that was on earlier IBM Z® servers, and they made it standard on the z15 server. Using the new Integrated Accelerator for zEDC, there is up to 8x faster application elapsed time with no additional CPU time over z14 zEDC Express[2]. For the IBM LinuxONE™ III, this compression is 100X faster than software compression on Intel[3].

For easy migration, all data is 100% compatible between the zEDC Express adapter and the new Integrated Accelerator for zEDC.

Industry standard
zEDC is compatible with industry standard, open zlib based compression – used today by Java and other applications. The zlib library can send compression and decompression for acceleration.

The z/OS-provided zlib library can be statically linked into IBM, ISV, or customer applications currently use zlib, enabling additional exploitation of compression through the Integrated Accelerator for zEDC and expanding potential compression opportunities.

On Linux® for IBM Z, sweet spot workloads for the feature are large request sizes and products using gzip/deflate standard compression. zlib is also used by some network protocols and applications such as http/1.1, openssh, openssl, subversion, git, or PNG.

Use Cases for zEDC Compression
BSAM/QSAM
Customers using large sequential BSAM/QSAM extended format, files can use Integrated Accelerator for zEDC to help reduce disk space up to 6X[4] and in some cases shorten elapsed time, reducing batch windows without significant CPU overhead.

Transparent Java exploitation
IBM Java compression applications using java.util.zip.Deflater class can achieve up to a 15x throughput improvement with IBM Java™ 8 SR6 on z/OS with z15 and Integrated Accelerator for zEDC compared to using IBM Java 8 SR5 FP25 on z/OS with IBM z14™ and zEDC Express adapters.[5]

[1] Data Age 2025, David Reinsel, John Gantz and John Rydning, IDC, November 2018
[2] Disclaimer: Measurements were collected in a controlled environment running an IBM developed workload under z/OS comprised of an equal mix of compression and decompression. Individual results may vary. Results are workload dependent.
[3] Performance results based on geometric mean of single threaded Java application runs using java.util.zip.Deflater class compressing classical literature text books in memory using various buffers sizes on LinuxONE III RHEL 7.6 alternate Kernel 4.14 versus Skylake Intel(R) Xeon(R) Gold CPU @ 2.60GHz Ubuntu 18.04 kernel 4.15.
[4] Disclaimer: Measurements completed in a controlled environment. Results may vary by customer based on individual workload, configuration and software levels.
[5] Disclaimer: Measurements were collected in a controlled environment running an IBM developed Java application on z/OS 2.3 that used java.util.zip.Deflater class to deflate in memory text data of classical English books

ZSB03059-USEN-07

Business Partner Data Exchange
zEDC is well suited for an environment where large files created and compressed using zlib on one system need to be transferred to another system for decompression and processing. Using Integrated Acclerator for zEDC on a z15, with up to 4 IFLs, can compress data up to 42X faster compared to using software compression[6].

Faster database backup
Perform a database backup up to 27x faster and with up to 30x less CPU time for Db2® LUW on one core of a z15/LinuxONE III LPAR using the Integrated Accelerator for zEDC versus a compared x86 platform using software compression[7].

Savings for web traffic
Up to 30x lower latency and up to 28x less CPU utilization on IBM z15 by compressing secure web transaction data before encryption using the Integrated Accelerator for z Enterprise Data Compression instead of using software compression[8].

Integrated Accelerator for zEDC Requirements
The minimum HW requirements are:
- IBM z15 (z15)
- IBM LinuxONE III

The minimum software requirements are:
- z/OS V2.1 or later with zEDC for z/OS feature, and applicable PTFs
- One of the following Linux distributions for IBM Z: Canonical Ubuntu 16.04.03 LTS and later, Red Hat Enterprise Linux (RHEL) 7.3 and later, or SUSE Linux Enterprise Server (SLES) 12 SP3 and later. The new Integrated Accelerator for z15 requires RHEL 8.1.11, SLES 12 SP51, and Ubuntu 19.10.[9]

zEDC candidate analysis for BSAM/QSAM
The IBM z System Batch Network Analyzer (zBNA) is a no charge, as-is tool that analyzes batch windows using SMF records to help determine if you have files that are candidates for zEDC. It can also help estimate the number of hardware features you will need. It is available from: http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/PRS5132?OpenDocument&ExpandSection=5%2C6#_Section5

Summary
The zEDC capability working with the Integrated Accelerator for zEDC offers an improved compression acceleration option. The use of zEDC allows the opportunity to store compressed data on Z more cost effectively and can enable you to more effectively use network bandwidth when sending compressed files.

[6] Disclaimer: Performance results based on IBM internal tests running the minigzip benchmark with compression level -1 from the dfl tcc branch of zlib (downloaded from https://github.com/iii-i/zlib/tree/dfltcc-20190708). Source data files were taken from the Large Corpus (downloaded from http://corpus.canterbury.ac.nz/descriptions). Canterbury.tar contained all files from all corpora. Results may vary. z15 configuration: LPAR with 4 dedicated IFLs, 64 GB memory, 40 GB DASD storage, SLES 12 SP4 (SMT mode).
[7] Disclaimer: Performance results are based on IBM internal tests running database backup with compression on Db2 LUW 11.1.4.4 on a database of size 385 GB using the Db2 built-in software compression on x86, and gzip -1 on z15. On z15 gzip was invoked with a comparable option -1 (compression level 1) and exploited the Integrated Accelerator for z Enterprise Data Compression. Results may vary. z15 configuration: LPAR with 1 IFL, 1.5 TB memory, RHEL 7.6 in SMT mode, database located on IBM DS8000 storage. x86 configuration: 1 Intel® Xeon® Gold 6140 CPU @ 2.30GHz with Hyperthreading turned on, 1.5 TB memory, RHEL 7.6, database located on local SSD storage (RAID 5).
[8] Disclaimer: Performance results based on IBM internal tests running the wrk2 4.0.0.0 benchmarking tool (https://github.com/giltene/wrk2) remotely with a fix transaction rate against a NGINX 1.15.9 web server exploiting zlib (https://github.com/madler/zlib/pull/410) to compress transaction data before encryption versus zlib -1 software compression. Data transmitted via NGINX webserver was the Silesia compression corpus (http://sun.aei.polsl.pl/~sdeor/index.php?page=silesia). Results may vary. z15 configuration: LPAR with 4 dedicated IFL, 32 GB memory, 40 GB DASD storage, 200 GB FlashSystem 900 storage, SLES12 SP4 (SMT mode), running NGINX 1.15.9 with patch https://github.com/nginx/nginx/commit/cfa1316368dcc6dc1aa82e3d0b67ec0d1cf7eebb.
[9] IBM is working with the Linux distribution partners to get the functionality included in their distribution for Linux on Z

ZSB03059-USEN-07