How to Employ Compression with Datacom and Presspack

CTC 19 - DBA – Thursday, April 28 2016

Ken Eaton, Kevin Shuma

Abstract

Guidelines and best practices on how to roll out data compression within your environment leveraging Online Compression.

This session will also highlight benchmark performance numbers on savings to DASD, IO and CPU.

Agenda

- New Online Compress ability – TA24
- Demo of Online Compress
- What are the benefits of compression?
- Benchmark in our lab
- Summary of findings

New Online Compress ability – TA24

- Provides the user the ability to alter a table's compression algorithm without interrupting data access
- Avoids a costly data access outage to implement or change compression
- Compression/Decompression activity runs in background in MUF
- Fully restart-able across failures (lack of space, MUF or LPAR)
  - The partially compressed data remains accessible
  - Once the failure issue is resolved, the Online Compress will be restarted and run to completion
- Process can be stopped (REQABORT)
  - Allows user to reverse or change compression setting
    - New process will run to completion and reset all rows to new selection

New Online Compress – Data Blocks

- When decompressing a table
  - Rows may grow and additional data blocks will be used
- When compressing a table
  - Rows will shrink making more space available in each data block
  - Space immediately available for use by new requests
- To complete the process a online reorganization is recommended
  - OLREORG REFGROUP=0 provides best result (DSOP 4/5)
    - May require a data area extension if not enough empty blocks to get rolling
  - OLREORG REFGROUP=2+ provides good result (DSOP 1/2)
    - Will not make the data as dense as REFGROUP=0
    - May require multiple executions

Demo of Online Compress

- Sample table contains 1 million rows from NYC property records
- Rows by default are fixed length (uncompressed)
- The table is loaded densely
- Uncompressed data rows fill 62K data blocks
- After executing TA24 to compress using Presspack Strong
  - 62K data blocks are still in use but each block has significant free space
- After OLREORG REFGROUP=00 data blocks in-use is reduced to 24.5K data blocks (60% reduction)
- Processing (Buffer reuse) significantly improves

DBUTLTY DATASP Report (Before TA24)

- DATASP shows data blocks, data bytes and data rows “IN-USE”

```
Base:   400        ********************************************************************************* Directory: QAMUF4

AREA UNC  BLKSIZE  4,096  TRACKS      5,250  BLOCKS     63,000  URI YES  DSOP 4 RANDOM FOR OLREORG


TBL  CMP  USER COMPRESSION   RECLN        RECORDS
UNC                           237      1,048,576

 GROUP#       BLOCKS        KBYTES             ROWS FREESP  ROWS ROWLEN   ---------- FREESPACE IN BLOCKS -----------
              IN-USE   %    IN-USE   %       IN-USE    MAX   MAX    MAX   0 TO 1/4K       1/2K         1K          2K
              EMPTY    %    EMPTY    %      DELETED    MIN   MIN    MIN          3K         4K          8K         12K
                                           *AVAIL     AVG   AVG    AVG         16K         20K         24K         32K

 TOTALS       61,683   99   242,809   98   1,048,576  4,094    17    237      61,680           1           0           0
                 526    1     6,019    2           0     65     0    237           0         526           0           0
                                              8,943     99    17    237           0           0           0           0
```

Read next program

- Batch program used to read sequentially through the database
  - Reads each row once by master key (data rows not in order)
- Program is executed 3 times
- MUF statistics collected for each run (no other work)
  - Statistics are consistent for each run

Read next batch before TA24

- Execute read next program

TABLE/AREA	REQUESTS	I/O EVENTS - READS	I/O EVENTS - WRITES	LOGICAL READS	LOGICAL WRITES
CXX	0	206	23	865	38
FXX	0	18	17	18	0
LXX	0	3	12	3	4
IXX00006	0	0	0	4	0
IXX00017	0	0	0	8	0
TTM00017	1	0	0	2	3
IXX00500	0	2,553	0	1,051,129	0
UNC00500	1,048,577	68,887	0	1,048,578	0
CBS01006	0	2	0	2	0
IXX01006	0	8	0	9	0
ADS01007	9	0	0	0	0
A0101007	0	3	1	8	3
A0201007	0	6	0	6	0
IXX01007	0	12	0	14	0
SPL01007	0	0	0	0	0
TOTAL	1,048,587	71,698	53	2,100,646	48

Enabling TA24 in environment (CXX)

- Execute a simple DBUTLTY CXXMAINT function to allow the online compression function
  - Once per CXX

```
//*************************************************************
//TA24ON    EXEC PGM=DBUTLTY,REGION=0M
//SYSIN DD  *
 CXXMAINT OPTION=ALTER,OPTION2=Y_ALLOW_TA24
/*
```

Use DDUPDATE to update Datadictionary and trigger Online Compression

- Using DDUPDATE ensures Datadictionary and CXX remain in synch
- Use standard compression keywords

```
//DDUPDATE  EXEC  DDUPDATE,REGION=4M
//SYSIN     DD  *
-USR  DATACOM-INSTALL,NEWUSER
-UPD  TABLE,UNCOMPRESS(PROD)
1000  ALTER
1500  COMPRESS-EXIT,PRESSPAK
1500  ENCRYPTION-KEY,STRONG
-END
```

Online Compression triggers TA24 activity in MUF

- 1 Million rows compressed in 2 minutes (elapsed)

```
 DSNZMUF# STC01849 EXECUTE  JESMSGLG JES2              CA31
 ------------------------------------------------------------------------------
 30...+....40...+....50...+....60...+....70...+....80...+....90...+....100..+....110..+....120..+....1|0.
MUF#1:DB00903I - OPENED, BASE -   500
MUF#1:DB02823I - TABLE_ALTER_24X7 500 UNC STATUS ACCEPTED TRANSITION COMPRESSION
MUF#1:DB02823I - TABLE_ALTER_24X7 500 UNC STATUS ALTERING COMPRESS NONE TO PRESSPAK (STRONG)
MUF#1:DB02823I - TABLE_ALTER_24X7 500 UNC STATUS STARTED
MUF#1:DB08002I - CA Datacom Presspack Version 15.0
MUF#1:DB00102I - ENDED   JOB-$5ONLCMP NUMBER-27390
MUF#1:DB02824I - TABLE_ALTER_24X7 500 UNC ALTERED ROW 1 OF ABOUT 1,048,576
MUF#1:DB00133I - JOB OPEN $5ONLCMP 27391 EATKE01 JOB01851
MUF#1:DB00102I - ENDED   JOB-$5ONLCMP NUMBER-27391
MUF#1:DB02824I - TABLE_ALTER_24X7 500 UNC ALTERED ROW 1,048,576 OF ABOUT 1,048,576
MUF#1:DB02823I - TABLE_ALTER_24X7 500 UNC STATUS COMPLETE
MUF#1:DB01311I - STATUS_TA24
MUF#1:DB01325I - CONSOLE COMPLETE, STATUS_TA24
MUF#1:DB00813I - ELAPSED SECONDS 120 CPU SECONDS 5.81
MUF#1:DB00814I - RQ/E 2 RD/E 218 WR/E 173 IX/E 3029 DX/E 11352 DT/E 17478
MUF#1:DB00815I - RQ/C 43 RD/C 4521 WR/C 3574 IX/C 62574 DX/C 234485 DT/C 360997
MUF#1:DB00816I - RQ 253 RD 26270 WR 20767 IX 363555 DX 1362358 DT 2097398
MUF#1:DB01311I - STATUS
MUF#1:DB01327I - MULTI-USER AVAILABLE, TASKS - ATTACHED-----0, AVAILABLE--2000       04/14/16 12:45:26
```

DBUTLTY DATASP Report after TA24

- DATASP shows same number of blocks and rows IN-USE
  - But previous 242K bytes now stored in 97K bytes (60% reduction)

```
Base:   400          ********************************************************************************* Directory: QAMUF4

AREA UNC  BLKSIZE  4,096  TRACKS      5,250  BLOCKS     63,000  URI YES  DSOP 4 RANDOM FOR OLREORG


TBL  CMP  USER COMPRESSION    RECLN        RECORDS
UNC       PRESSPAK STRONG       237      1,048,576

 GROUP#       BLOCKS        KBYTES             ROWS FREESP  ROWS ROWLEN   ---------- FREESPACE IN BLOCKS -----------
              IN-USE  %     IN-USE   %         IN-USE    MAX   MAX    MAX   0 TO 1/4K       1/2K        1K          2K
              EMPTY   %      EMPTY   %        DELETED    MIN   MIN    MIN          3K         4K        8K         12K
                                              *AVAIL    AVG   AVG    AVG         16K        20K       24K         32K

 TOTALS       61,683  99     97,225  39      1,048,576  4,094    17    123           0          0         0          0
                 526   1    151,603  61              0  2,139     0     54      61,681        526         0          0
                                             1,612,072  2,496    17     95           0          0         0          0
```

Read next batch after TA24 but before OLREOG

- Execute same batch program, IO processing the same

TABLE/AREA	REQUESTS	I/O EVENTS - READS	I/O EVENTS - WRITES	LOGICAL READS	LOGICAL WRITES
CXX	0	206	23	865	38
FXX	0	18	17	18	0
LXX	0	3	12	3	4
IXX00006	0	0	0	4	0
IXX00017	0	0	0	8	0
TTM00017	1	0	0	2	3
IXX00500	0	2,553	0	1,051,129	0
UNC00500	1,048,577	68,887	0	1,048,578	0
CBS01006	0	2	0	2	0
IXX01006	0	8	0	9	0
ADS01007	9	0	0	0	0
A0101007	0	3	1	8	3
A0201007	0	6	0	6	0
IXX01007	0	12	0	14	0
SPL01007	0	0	0	0	0
TOTAL	1,048,587	71,698	53	2,100,646	48

DBUTLTY OLREORG REFGROUP=0

- OLREORG TABLE=UNC,DBID=0500,REFGROUP=0
- Takes about 5 minutes (elapsed)

```
- STARTING FUNCTION -  OLREORG TABLE=UNC,DBID=0500,REFGROUP=0
- PARM PGMDT=20010901, DBUT6PR  2016/01/14-1532 15.0       RO87686
MUF#1:DB00135I - CONNECT TO DSNZMUF#          MUF#1    STC02338 CA31     M-15.0 I-15.0 0001
MUF#1:DB00101I - STARTED JOB-$4ONLORG NUMBER-27589 CXX=QAMUF#   MUFNAME=MUF#1
- OLREORG BASE 500 TABLE UNC ROW 490,994 OF ABOUT 1,048,576   ROWS MOVED 463,460
- OLREORG BASE 500 TABLE UNC ROW 992,119 OF ABOUT 1,048,576   ROWS MOVED 931,799
MUF#1:DB00136I - DISCONNECT DSNZMUF#          MUF#1    STC02338 CA31     M-15.0 I-15.0 0001
```

DBUTLTY DATASP Report after TA24 plus OLREORG

- DATASP shows number of blocks in use dropped dramatically – free space blocks now available (reduced by 60%)

```
Base:   400        ********************************************************************************** Directory: QAMUF4

AREA UNC  BLKSIZE  4,096  TRACKS      5,250  BLOCKS     63,000  URI YES  DSOP 4 RANDOM FOR OLREORG


TBL  CMP  USER COMPRESSION   RECLN        RECORDS
UNC       PRESSPAK STRONG      237      1,048,576

 GROUP#      BLOCKS       KBYTES            ROWS FREESP  ROWS ROWLEN   ---------- FREESPACE IN BLOCKS -----------
             IN-USE  %    IN-USE   %       IN-USE    MAX   MAX    MAX   0 TO 1/4K       1/2K        1K          2K
             EMPTY   %    EMPTY    %      DELETED    MIN   MIN    MIN          3K         4K        8K         12K
                                          *AVAIL    AVG   AVG    AVG         16K        20K       24K         32K

 TOTALS     24,574  40    97,225  39     1,048,576  4,094    73    123      24,571          0         0          0
            37,635  60   151,603  61             0      0     0     54           1     37,635         0          0
                                        1,587,330  2,496    17     95           0          0         0          0
```

Batch Read next after OLREORG results

- Execute same batch program
  - IXX IO the same, Data IO dropped from 68,887 to 24,676 (74% reduction)

TABLE/AREA	REQUESTS	I/O EVENTS - READS	I/O EVENTS - WRITES	LOGICAL READS	LOGICAL WRITES
CXX	0	206	23	865	38
FXX	0	18	17	18	0
LXX	0	3	12	3	4
IXX00006	0	0	0	4	0
IXX00017	0	0	0	8	0
TTM00017	1	0	0	2	3
IXX00500	0	2,553	0	1,051,129	0
UNC00500	1,048,577	24,676	0	1,048,578	0
CBS01006	0	2	0	2	0
IXX01006	0	8	0	9	0
ADS01007	9	0	0	0	0
A0101007	0	3	1	8	3
A0201007	0	6	0	6	0
IXX01007	0	12	0	14	0
SPL01007	0	0	0	0	0
TOTAL	1,048,587	27,487	53	2,100,646	48

Any Questions on TA24 before we look at some benchmark information?

What are the benefits of compression?

- A compressed data block typically has significantly more rows which means
  - Less blocks on DASD to store the same rows
    - Multiplied by number of “copies” of the data – PROD/QA/TEST
  - Better data buffer reuse which reduces IO and improves performance
  - Less memory required to MRDF COVER tables
  - Less log space used per update when LOG_RECORD_USER_CMPRS YES
  - Quicker utility processing for backups and loads (less IO)
- Some other benefits include
  - Can make data less “readable” by non Datacom utilities like TSO Browse
  - Slowing DASD growth and multiple extent management

What are the costs of compression?

TA24 X Requires a data access outage to implement

TA24 X Not easily removed or changed

TA24 X Hard to switch from one vendor to another

zIIP X Uses expensive general purpose processor (GP CPU) instructions to save cheap DASD

Using zIIP for compression

- zIIP processors can be used for in-memory processing
- zIIP processors cannot be used for physical IOs
- zIIP processors can only be used for code you own
  - Cannot be used to process imbedded user/vendor code
    - Example: Non-CA compression code imbedded in MUF

- zIIP to tip the balance “for” using compression
  - Saves DASD, memory and IOs
  - Trades GP CPU used to do IOs for zIIP CPU used to do compression

Benchmark in our lab

- New z13 processor
  - Multiple GPs and zIIPs lightly loaded – should not have contention
- Using the same NYC table mentioned above
- Build 5 versions of the table, each there own database to isolate activity (including IXX activity)
  - Uncompressed
  - DB Compressed
  - Presspack Weak
  - Presspack Strong
  - Presspack Custom

Benchmark in our lab (cont'd)

- Repeatable process
  - MASSADD rows using 10 concurrent tasks (data not in order)
  - READ sequentially using 10 concurrent tasks and 4 different key paths
  - UPDATE
    - Use same sequential READ programs
    - Use a second request area to read and update 1 in 10 rows
      - Use special update mask to pick rows so there is no possibility of an update collision
      - No locking or waits
  - No DELETE processing
    - Typically a small part of MUF processing
    - Would be "all good" with IO savings and no compression/decompression processing

Benchmark in our lab (cont'd)

- Use small SRB/zIIP tasking
  - Demonstrate savings in minimal number of processors
  - SMPTASK 2,2,4,SRB
- 3 buffer/MRDF settings
  - Small buffers
  - Large buffers
  - Large buffers with covered databases
- Repeated multiple times to ensure consistency

CPU/IO consumption MASSADD 1M rows

Small buffers

Base	JARS Elasped	Billable Total CPU	LXX	IXX	Data
UNC	00:01:53.51 *	00:00:05.98 *	31,144	305,420	11,164
DBC	00:01:58.00 *	00:00:05.73 *	29,886	297,412	6,770
PWC	00:01:37.64 *	00:00:05.58 *	30,559	294,682	6,855
PSC	00:01:19.63 *	00:00:05.61 *	29,967	307,856	5,021
PCC	00:01:12.39 *	00:00:05.48 *	27,709	296,417	4,738

- Small savings in Elapsed and billable CPU, good reduction in IO
- Typically a small part of processing

CPU/IO consumption
READ 10-UP 1M rows each

Small buffers

Base	JARS Elasped	Billable Total CPU	LXX	IXX	Data
UNC	00:05:38.82 *	00:00:20.47 *	20	16,796	1,621,985
DBC	00:04:12.57 *	00:00:16.68 *	20	16,880	1,191,894
PWC	00:03:54.57 *	00:00:18.31 *	20	16,817	1,328,618
PSC	00:03:27.02 *	00:00:16.28 *	20	20,666	1,118,444
PCC	00:03:44.61 *	00:00:18.14 *	20	20,854	1,329,996

Large Buffers

Base	JARS Elasped	Billable Total CPU	LXX	IXX	Data
UNC	00:04:48.63 *	00:00:19.90 *	20	14,938	1,326,335
DBC	00:03:42.48 *	00:00:15.34 *	20	15,051	899,087
PWC	00:03:14.20 *	00:00:15.70 *	20	14,959	924,237
PSC	00:02:10.02 *	00:00:10.70 *	20	14,994	562,638
PCC	00:02:06.09 *	00:00:08.45 *	20	14,985	402,835

- Significant savings in Elapsed and billable CPU, good reduction in IO
- Typically the largest part of MUF processing (85%)

CPU/IO consumption
READ 10-UP 1M rows each (cont'd)

MRDF

Base	JARS Elasped	Billable Total CPU	LXX	IXX	Data
UNC	00:01:49.89 *	00:00:03.79 *	20	14,846	58,826
DBC	00:01:25.45 *	00:00:03.33 *	20	14,946	33,498
PWC	00:01:17.64 *	00:00:03.09 *	20	14,865	33,994
PSC	00:01:08.34 *	00:00:02.69 *	20	14,901	23,423
PCC	00:01:10.44 *	00:00:02.47 *	20	14,890	21,797

- Savings in Elapsed and billable CPU
  - Good reduction in IO even when dataset in MRDF

CPU/IO consumption
UPDATE 10-UP 100K of 1M rows updated

Small buffers forced expands

Base	JARS Elasped	Billable Total CPU	LXX	IXX	Data
UNC	00:05:54.70 *	00:00:36.89 *	37,337	25,119	2,796,153
DBC	00:09:51.88 *	00:00:45.82 *	49,818	105,093	2,982,844
PWC	00:10:57.89 *	00:00:43.99 *	48,909	100,899	2,917,078
PSC	00:09:01.55 *	00:00:36.18 *	42,148	97,042	2,278,513
PCC	00:08:31.40 *	00:00:36.86 *	44,914	116,397	2,288,547

Small buffers no expands

Base	JARS Elasped	Billable Total CPU	LXX	IXX	Data
UNC	00:07:20.70 *	00:00:34.83 *	35,065	23,108	2,440,203
DBC	00:06:41.56 *	00:00:32.05 *	32,667	28,549	2,170,885
PWC	00:05:22.23 *	00:00:34.28 *	40,860	23,930	2,449,149
PSC	00:05:05.68 *	00:00:33.99 *	38,458	32,095	2,321,709
PCC	00:04:27.96 *	00:00:31.06 *	37,592	31,186	2,221,882

- Savings fluctuates depending on amount of expansions triggered by update
- Typically the 2nd largest part of MUF processing (10%)

CPU/IO consumption
UPDATE 10-UP 100K of 1M updated (cont'd)

Large buffers no expands

Base	JARS Elasped	Billable Total CPU	LXX	IXX	Data
UNC	00:04:14.99 *	00:00:37.67 *	21,098	14,846	2,692,077
DBC	00:03:26.08 *	00:00:21.21 *	12,499	14,945	1,221,391
PWC	00:03:35.76 *	00:00:23.96 *	14,277	14,864	1,403,587
PSC	00:02:40.08 *	00:00:15.76 *	9,506	14,901	766,733
PCC	00:02:57.00 *	00:00:17.24 *	10,456	14,889	864,681

MRDF no expands

Base	JARS Elasped	Billable Total CPU	LXX	IXX	Data
UNC	00:01:45.61 *	00:00:06.00 *	15,209	14,846	93,397
DBC	00:01:37.10 *	00:00:05.10 *	11,269	14,945	54,322
PWC	00:01:34.95 *	00:00:04.71 *	12,000	14,864	57,050
PSC	00:01:15.93 *	00:00:04.26 *	12,593	14,901	44,278
PCC	00:01:16.28 *	00:00:03.98 *	12,646	14,889	42,473

- Savings significant with large buffers and MRDF (no expansions)
- Typically the 2nd largest part of MUF processing (10%)

What if there is no zIIP processor READ 10-UP 1M rows each

Small buffers

Base	JARS Elasped	Billable Total CPU	LXX	IXX	Data
UNC	00:06:21.78 *	00:00:43.94 *	20	18,375	1,673,276
DBC	00:04:00.78 *	00:00:40.41 *	20	22,203	1,041,159
PWC	00:07:00.70 *	00:00:51.85 *	20	18,239	1,373,905
PSC	00:07:52.50 *	00:00:47.88 *	20	14,762	850,822
PCC	00:10:53.46 *	00:00:57.46 *	20	18,220	1,000,476

Large Buffers

Base	JARS Elasped	Billable Total CPU	LXX	IXX	Data
UNC	00:01:33.65 *	00:00:20.58 *	11,733	60,471	10,041
DBC	00:01:37.63 *	00:00:19.42 *	12,702	60,254	6,139
PWC	00:01:30.11 *	00:00:22.92 *	12,807	60,143	6,215
PSC	00:01:05.90 *	00:00:20.76 *	12,880	60,089	4,587
PCC	00:01:09.90 *	00:00:21.07 *	12,655	59,973	4,337

- Savings depends on good reduction in IO
- May or may not be “worth it”

What if there is a 3rd party compression

MRDF w/Presspack

Base	JARS Elasped	Billable Total CPU	LXX	IXX	Data
UNC	00:01:25.79 *	00:00:07.44 *	20	14,819	58,826
DBC	00:01:12.81 *	00:00:06.96 *	20	14,807	33,497
PWC	00:01:23.17 *	00:00:06.36 *	20	14,728	33,989
PSC	00:01:11.49 *	00:00:05.37 *	20	14,749	23,424
PCC	00:01:27.86 *	00:00:03.62 *	20	14,710	21,796

MRDF w/3rd party JOEPACK

Base	JARS Elasped	Billable Total CPU	LXX	IXX	Data
UNC	00:01:18.68 *	00:00:07.47 *	20	14,819	58,826
DBC	00:02:12.50 *	00:00:06.73 *	20	14,807	33,497
PWC	00:03:27.58 *	00:00:46.22 *	20	14,728	33,989
PSC	00:03:32.58 *	00:00:53.67 *	20	14,749	23,424
PCC	00:05:30.92 *	00:00:50.64 *	20	14,710	21,796

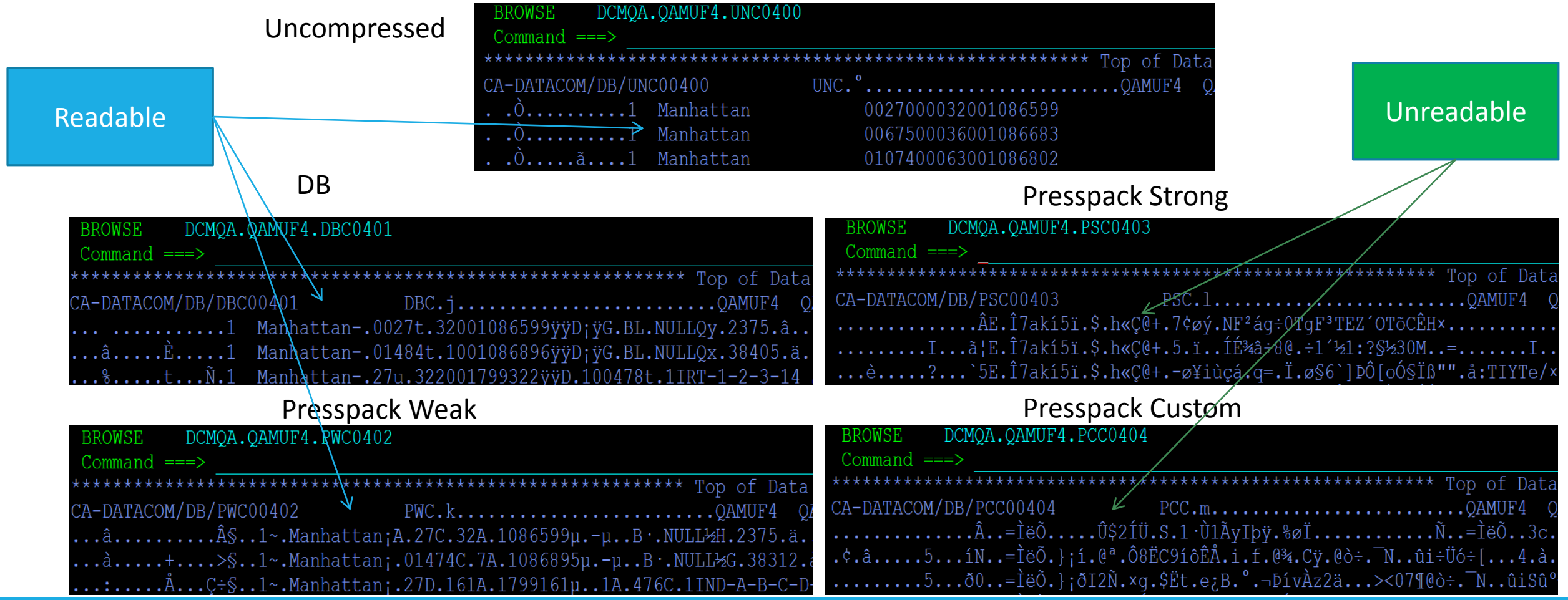
Do not use 3rd party compression and SMPTASK x,x,x,SRB (zIIP)

- Same exact work – same performance for UNC and DBC
- JOEPACK causes swapping between SRB (normal) and TCB (compression)
  - 10x worse performance

Summary

- Online Compression (TA24) removes several of the key drawbacks of compression, now
  - No longer requires a data access outage to implement
  - Can be easily removed or changed
  - Easily switch from 3rd party vendor to CA
- When compression (Presspack) combined with zIIP (SRB) processing, the TCO tips in favor of Presspack (CA) compression
  - Uses cheaper zIIP (CPU) instructions and saves DASD, IO and memory

One extra benefit – Presspack Strong compression can make data unreadable to external tools



FOR INFORMATION PURPOSES ONLY

Terms of this Presentation

This presentation was based on current information and resource allocations as of April 2016 and is subject to change or withdrawal by CA at any time without notice. Notwithstanding anything in this presentation to the contrary, this presentation shall not serve to (i) affect the rights and/or obligations of CA or its licensees under any existing or future written license agreement or services agreement relating to any CA software product; or (ii) amend any product documentation or specifications for any CA software product. The development, release and timing of any features or functionality described in this presentation remain at CA's sole discretion. Notwithstanding anything in this presentation to the contrary, upon the general availability of any future CA product release referenced in this presentation, CA will make such release available (i) for sale to new licensees of such product; and (ii) to existing licensees of such product on a when and if-available basis as part of CA maintenance and support, and in the form of a regularly scheduled major product release. Such releases may be made available to current licensees of such product who are current subscribers to CA maintenance and support on a when and if-available basis. In the event of a conflict between the terms of this paragraph and any other information contained in this presentation, the terms of this paragraph shall govern.

Certain information in this presentation may outline CA's general product direction. All information in this presentation is for your informational purposes only and may not be incorporated into any contract. CA assumes no responsibility for the accuracy or completeness of the information. To the extent permitted by applicable law, CA provides this presentation "as is" without warranty of any kind, including without limitation, any implied warranties or merchantability, fitness for a particular purpose, or non-infringement. In no event will CA be liable for any loss or damage, direct or indirect, from the use of this document, including, without limitation, lost profits, lost investment, business interruption, goodwill, or lost data, even if CA is expressly advised in advance of the possibility of such damages. CA confidential and proprietary. No unauthorized copying or distribution permitted.

Questions?