

DB2 Backup and Restore Best Practices

Dale McInnis, IBM Canada Ltd,
dmcinnis@ca.ibm.com

Please note



IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM's sole discretion.

Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

Agenda



- Overview

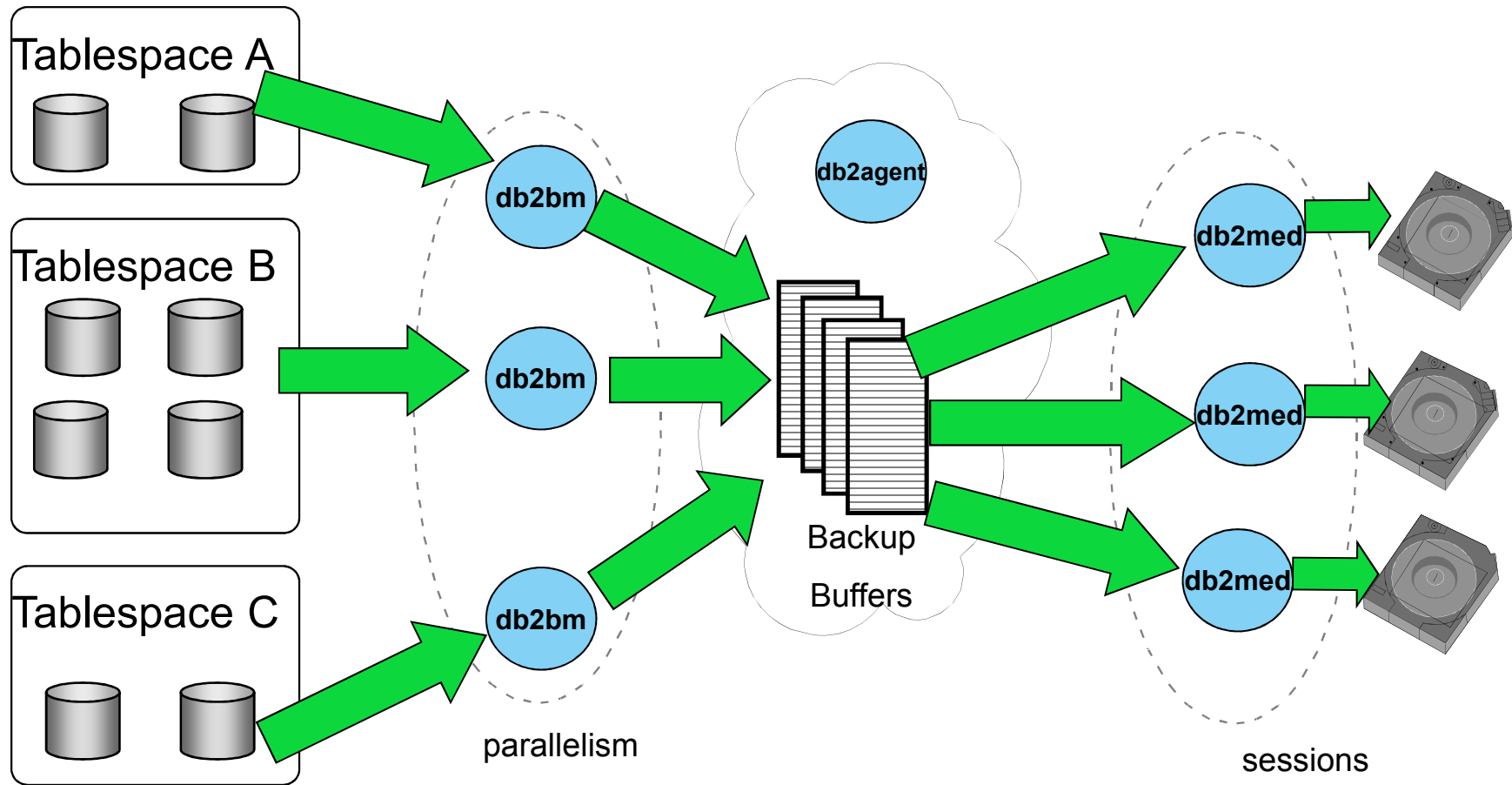
- Technology Review
 - What's new in Backup and Restore
 - What's new in Logging

- Usage Scenarios

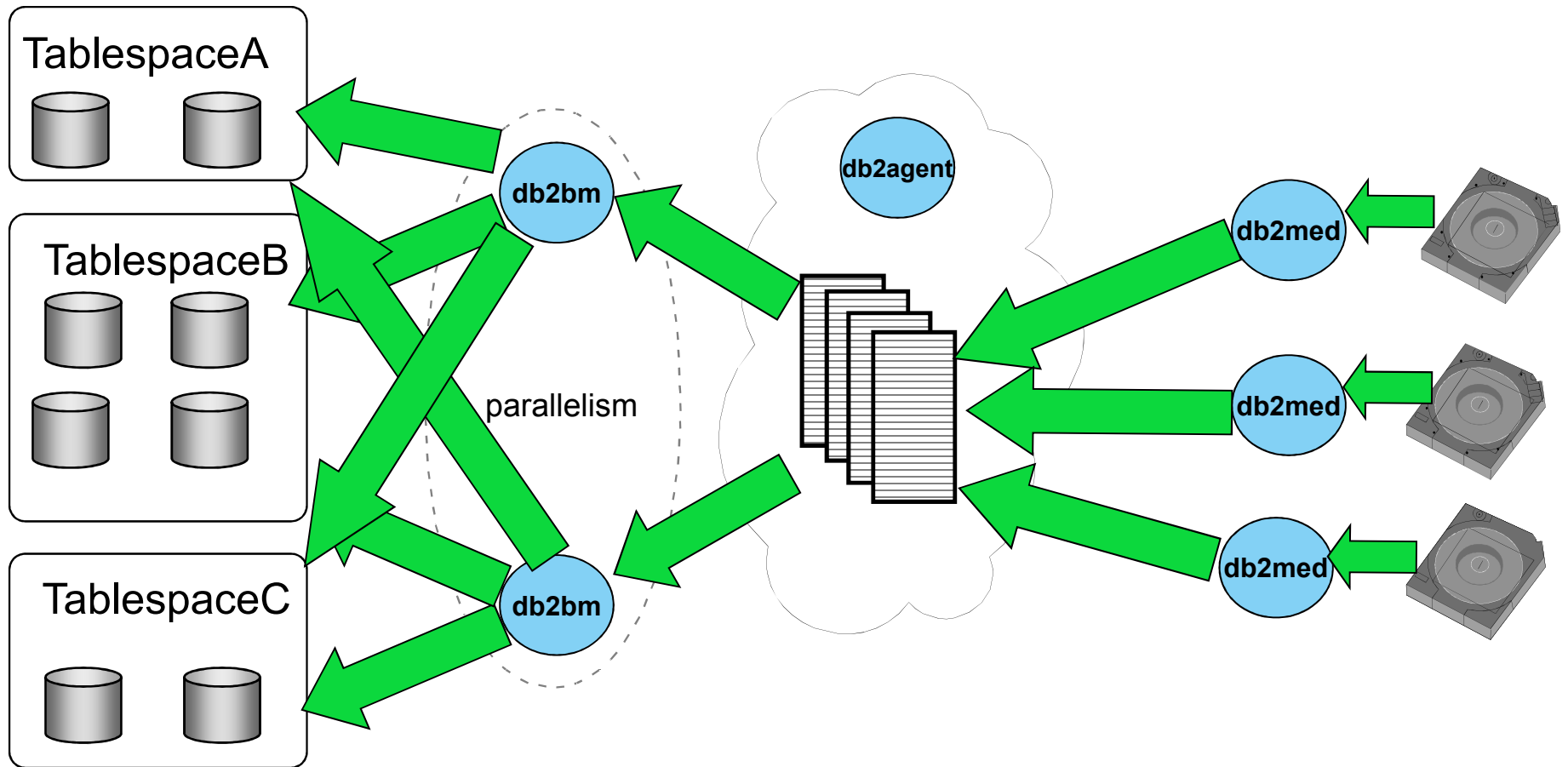
- TSM Recommendations

- Conclusion

Backup Process Model



Restore Process Model



Both backup and restore are self tuning

- If not specified the following settings will be computed
 - Parallelism
 - Buffer size
 - # of buffers

- All setting are dependent largely on the following settings:
 - UTIL_HEAP_SZ
 - # of CPUs
 - # of table spaces
 - Extent size
 - Page size

- Autonomics do handle all possible options
 - Compression
 - Data deduplication
 - ...



How can I make backup run faster?



- **Let the DB tune the backup parameters**
 - Ensure you heap size is large enough (util_heap_sz 50000)
- **If backing up to a file system create multiple objects**
 - If the target device is a raid 5 file system creating multiple backup objects will speed up the process
 - Backup db sample to /mydb/backup, /mydb/backup, /mydb/backup, ...
 - For TSM – open multiple sessions

Logs Stored in Backup Images



- **Ensures backup will be restorable if archived logs misplaced**
Supports all types of online backups, eg:
 - Database, Table space, Incremental, Compressed
- **Mechanism**
 - After the table space data is backed up, the tail log file will be truncated
 - All logs needed to restore the backup and roll forward to the time corresponding to the end of the backup, are placed in the backup image
 - Logs are backed up in parallel, similar to the mechanism used for table space data
 - Logs are placed at end of backup image, and compressed, if backup compression is specified

```
BACKUP DB <dbname> ONLINE to <backup_path> INCLUDE LOGS
```

```
RESTORE DB <dbname> ONLINE FROM <backup_path> LOGTARGET <log_path>  
ROLLFORWARD DB <dbname> TO END OF LOGS AND STOP  
OVERFLOW LOG PATH <log_path>
```

```
// Or, to restore just the LOGS
```

```
RESTORE DB <dbname> LOGS FROM <backup_path> LOGTARGET <log_target>
```


What about backup compression?



- **Compression can be used in 4 different areas**
 1. Table compression
 - Row level compression in the database
 - 10.1 introduced adaptive compression
 2. DB2 Backup compression
 - compressed while backing up
 - Requires additional CPU resources
 - Avoid is backup target is a data deduplication device
 3. TSM Software compression (if you have TSM)
 4. Storage (ie. hardware) level compression
 - Depends on your storage devices

- **Recommendation:**
 - Start with the following:
 - If you have table compression, you may not see a huge benefit with using db2 backup compression as well.
 - If you have storage level compression, then you do may not need to enable to TSM software compression.

- Test the combinations of compression types, to see what is the best fit, it will depend on the resource usage in your environment.

Backup and Recovery are DPF enabled

- **Single System View(SSV) in DPF**
 - Configuration
 - Hierarchical in nature, can be overridden on a partition level
 - Backup
 - Single command resulting in a single point of consistency across all partitions (DPF Aware Backup)
 - db2 backup db foo on all dbpartitionnums
 - db2 backup db foo on dbpartitionnums (0, 30)
 - db2 backup db foo on all dbpartitionnums except dbpartitionnums (10, 20)

- **Logs in Backup is supported in DPF**

- **Rollforward to end of backup simplified:**
 - db2 rollforward db foo to end of backup and stop

The backup history file



- Each database has a backup history file that contains information about database recovery operations
 - BACKUP
 - RESTORE
 - ROLLFORWARD
 - Log file archive

- The history file is pruned periodically to keep its size manageable
 - After a full database backup
 - Using PRUNE HISTORY

- Automatic pruning is managed by two database config parameters
 - NUM_DB_BACKUPS
 - REC_HIS_RETENTN

How to manage the life cycle of backup assets

- A new database configuration parameter that controls whether the underlying logs, backups and other associated objects are deleted when the history file is pruned

- AUTO_DEL_REC_OBJ
 - OFF (default, existing behaviour)
 - ON

- When does this automatic deletion occur ?
 - After a successful backup
 - On an explicit PRUNE HISTORY AND DELETE command

- What is deleted ?
 - Any full database backup images which exceed both NUM_DB_BACKUPS and REC_HIS_RETENTN db cfg parameters will be deleted
 - Any associated incremental backup images, load copy images, table space backup images or log files will be deleted

DB2 Advanced Copy Services (ACS)
PureScale support was added to
DB2 V 10.5 FP4

Review : Manual Flash Copy Backup



- Flashcopy backup and restore a largely manual process:
- Backup
 1. Identify LUN(s) associated with the database
 2. Identify free target LUN(s) for the copy
 3. Establish the flashcopy pair(s)
 4. Issue DB2 SUSPEND I/O command to tell DB2 to suspend write I/Os
 5. Issue storage commands necessary to do the actual flash copy
 6. Issue DB2 RESUME I/O command to return DB2 to normal
- Restore
 1. Restore/copy target LUN(s) containing backup of interest
 2. Issue DB2INIDB command to initialize the database for rollforward recovery
 3. Issue DB2 ROLL FORWARD command

DB2 Database



Flash Copy →



✗ No history file entry
✗ Error prone

Review : Integrated ACS Copy Backup



- Flashcopy backup/restore just like any other DB2 backup
- Backup

1
2 **DB2 BACKUP DB sample USE SNAPSHOT**
3

4. Issue DB2 SUSPEND I/O command to tell DB2 to suspend write I/Os
5. Issue storage commands necessary to do the actual flash copy
6. Issue DB2 RESUME I/O command to return DB2 to normal

- Restore

1 **DB2 RESTORE DB sample USE SNAPSHOT**
2
3 **DB2 ROLLFORWARD ...**

rest
or rollforward

DB2 Database



Flash Copy →



- ✓ History file record
- ✓ Simple !
- ✗ Wide (but not exhaustive) storage support

ACS Differences between V 9.7 and V 10.1+



Feature	V 9.7	V 10.1+
AIX	Y(5.3 & 6.1)	Y (6.1 & 7.1)
Linux	Y RHEL 5 & SLES 10 (nSeries/NetApp only)	Y RHEL 5&6 SLES 10&11 (all storage devices)
HPUX IA64	N	Y
Solaris SPARC	N	Y
DS8K	Y	Y
XIV	Y (Gen 2)	Y (Gen 2 & 3)
SVC	Y (2.1 – 4.3.1)	Y (4.3.0 – 6.2)
ESS800 (shark)	Y	N
DS6K	Y	N
Storwize V7000	N	Y
Storwize V5000	N	Y (10.5.0.5/10.1.0.4)
IBM N-Series	Y	Y (10.5.0.5/10.1.0.4)
NetApp	Y	Y (10.5.0.5/10.1.0.4)

Mapping of DB2 databases to storage subsystem volumes for using snapshot backups



- To use SNAPSHOT backup and restore functions DB2 databases need to be carefully configured on storage system volumes:
 - **Location** of database path and table spaces
The database path as well as all system, user and temporary table spaces need to be assigned to a set of storage system volumes that do not overlap any other DB2 database or even another database partition of the same database.
 - **Location** of active database logs
The active and mirror (mirrorlogpath) database logs need to be defined on a separate volume group. This ensures that the recovery logs are not overwritten during a snapshot restore.
 - **Location** of archive logs
Log files archived to local disk need to be defined on a separate volume group to avoid being overlaid by a snapshot restore

Managing snapshot backup objects – db2acsutil



- The **db2acsutil** command is used to manage snapshot backup objects.
- To list available snapshot backup objects, use **QUERY** parameter.
For example, to list available snapshot backup objects for the database manager instance named dbminst1, use the following syntax:

```
db2acsutil query instance dbminst1
```
- To check the progress of a given snapshot backup operation, use the **STATUS** parameter.
For example, to see the progress of snapshot backup operations that might be currently running on a database called database1, use the following syntax:

```
db2acsutil query status db database1
```
- To delete a particular snapshot backup object, use the **DELETE** parameter.
For example, to delete all snapshot backup objects for the database called database1 that are older than 10 days, use the following syntax:

```
db2acsutil delete older than 10 days ago db database1
```

DB2 V 10+ ACS Installation Change



- **Prior to V 10.1 DB2 ACS was automatically install in**
 - AIX
 - Linux
- **As of V 10.1 DB2 ACS will be automatically installed on AIX only**
 - sqllib/acs directory will contain Linux tar ball images**

Scripted Interface for ACS Copy Backup



- Flashcopy backup/restore just like any other DB2 backup
- Backup
 - Identify LUN(s) associated with the database
 - Identify from target LUN(s) for the copy
 - DB2 BACKUP DB sample USE SNAPSHOT SCRIPT**
 - '/myscript.sh'**
 - Issue storage commands necessary to do the actual flash copy
 - Issue DB2 RESUME I/O command to return DB2 to normal

DB2 RESTORE DB sample USE SNAPSHOT SCRIPT
'/myscript.sh' TAKEN AT <timestamp>

DB2 ROLLFORWARD ...

- Issue DB2 ROLLFORWARD command

DB2 Database



Flash Copy



- ✓ History file record
- ✓ Simple to use !
- ✓ Wider storage support enabled

Scripted Interface for Flash Copy Backup



Backup Command using the solution:
BACKUP DATABASE SAMPLE ONLINE
USE SNAPSHOT SCRIPT /script/libacssc.sh OPTIONS '/repository/'

DB2 LUW



calls

Scripted Interface
for DB2 ACS



invokes

queries



Customer
Script

Writes
Options

Reads
Options

Read
Options

May write
own information

Contents of Protocol Files

- DB and instance name
- DB path
- Container Paths
- Storage Paths
- Log Directories
- Options like include / exclude logs



Protocol
File Repository

Scripted interface for flash copy backup

- Available in DB2 10.5, DB2 9.7 FP 9 and DB2 10.1 FP3

- White Papers published or soon to be published
 - Introduction to the Scripted Interface
 - Implementation of the Scripted Interface for DB2 ACS Using Linux LVM
 - Implementation of the Scripted Interface for DB2 ACS Using IBM GPFS
 - Implementation of the Scripted Interface for DB2 ACS using IBM DS4000 midrange storage system

SET WRITE SUSPEND



New options added in DB2 V 10.5

INCLUDE LOGS

Specifies that writes to the log files are not allowed when the database is in a write-suspended state. This is the default.

EXCLUDE LOGS

Specifies that writes to the log files (but not to log file header and mirror log file header files) can occur when the database is in a write-suspended state. This provides a window during which update transactions running against the database can still complete. This can help to reduce the impact on the workload that would normally occur while the database is write suspended. Any copies of the database that are taken while it is write suspended and the `EXCLUDE LOGS` option is specified must not include log files in the copy.

Note: There are some situations in which logged operations can still be blocked from proceeding. This can happen, for example, if the current active log file is full

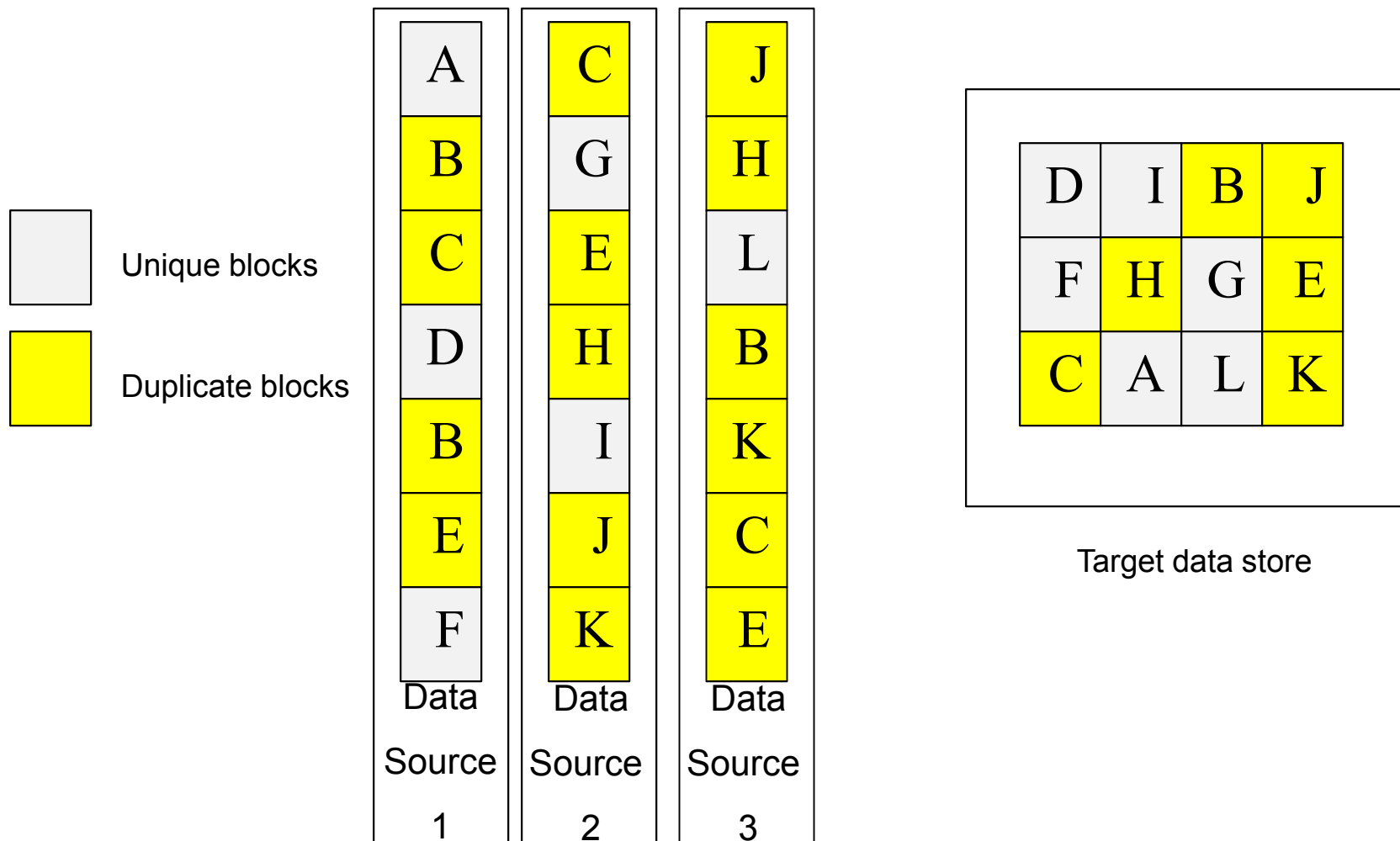
DB2 9.7 Backup and Recovery Enhancements



- Native support for all deduplication devices
- Alters the format the backup image to be dedup friendly
- See White paper:
<http://www.ibm.com/developerworks/data/library/techarticle/dm-1302db2deduplication/>

- What is data deduplication and how does it work
- How a traditional DB2 backup operation works
- How was DB2 modified to support data deduplication devices
- Is compression compatible with data deduplication devices
- Performance recommendations
- TSM Native data deduplication capabilities
 - TSM Server side deduplication
 - TSM server client deduplication
 - TSM deduplication
 - Where to enable TSM data deduplication
- Conclusions

Data Deduplication Concept



Default backup physical format



Tablespace A



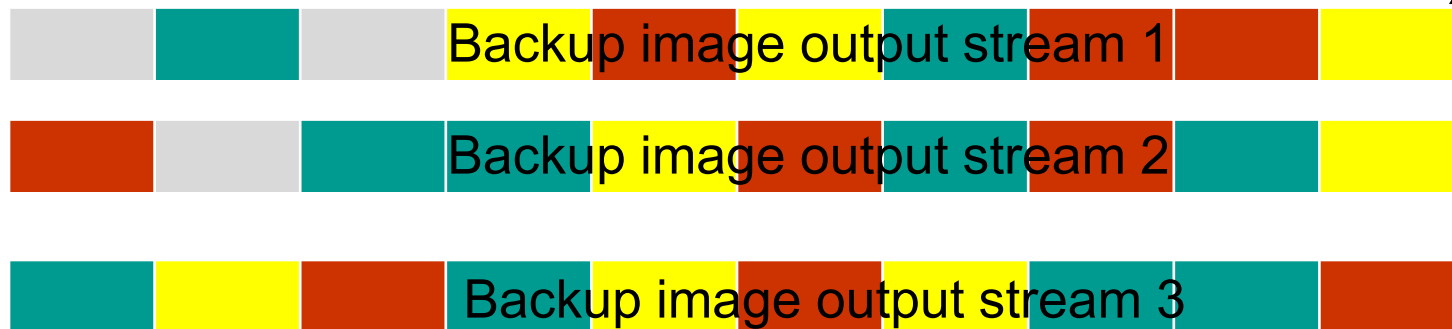
Tablespace B



Tablespace C



DB2 backup db mydb use TSM open 3 sessions



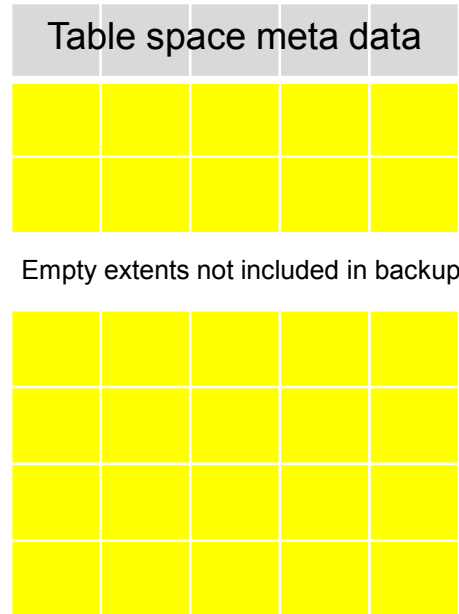
Dedup backup physical format



Tablespace 1



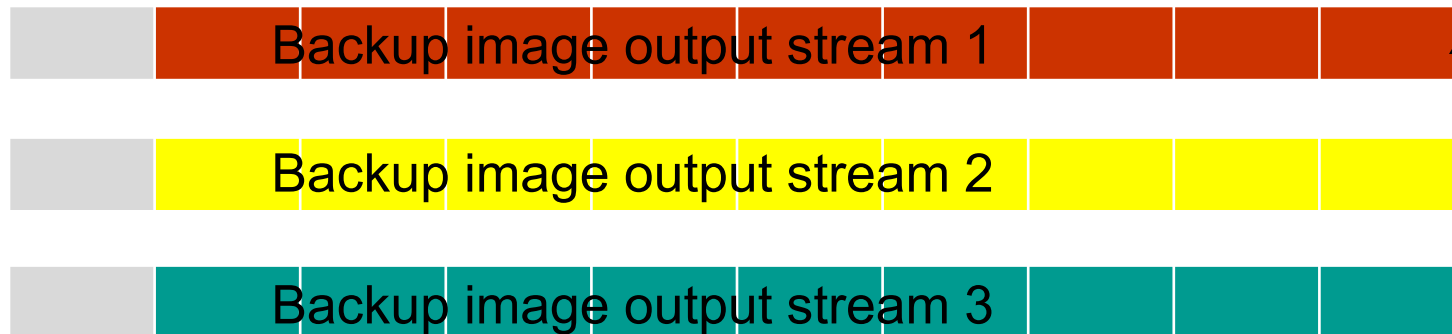
Tablespace 2



Tablespace 3



DB2 backup db mydb use TSM open 3 sessions dedup_device



1 extent

TSM Client Deduplication Results



Command:

DB2 backup db PERF use TSM open 8 sessions dedup_device

Settings:

Util_heap_sz = 50000

Autonomic performance settings:

Backup Buffer Size = 2465

Backup # of buffers = 18

Backup Parallelism = 8

Start	End	Elapsed Time	Actual DB Size in bytes	Dedup Size sent to server in bytes	% Savings in bandwidth
12:22:57 AM	2:32:54 AM	2:09	1.35891E+11	1.3574E+11	0.11%
2:33:02 AM	3:04:13 AM	0:31	1.35891E+11	3501526595	97.42%
3:04:19 AM	3:34:07 AM	0:29	1.35891E+11	1439490051	98.94%
3:34:13 AM	4:03:14 AM	0:29	1.35891E+11	875452538	99.36%
4:03:20 AM	4:32:45 AM	0:29	1.35891E+11	1015678830	99.25%

TSM Client Deduplication Results



Command:

DB2 backup db PERF use TSM open 8 sessions dedup_device

Settings:

Util_heap_sz = 300000

Autonomic performance settings:

Backup Buffer Size = 16384

Backup # of buffers = 18

Backup Parallelism = 8

Start	End	Elapsed Time	Actual DB Size in bytes	Dedup Size sent to server in bytes	% Savings in bandwidth
10:22:00 PM	1:53:47 AM	3:31	1.36499E+11	1.35729E+11	0.56%
1:53:55 AM	2:22:30 AM	0:28	1.36499E+11	1248387123	99.09%
2:22:26 AM	2:49:39 AM	0:27	1.36499E+11	352043759	99.74%
2:49:44 AM	3:17:12 AM	0:27	1.36499E+11	175945343	99.87%
3:17:18 AM	3:45:32 AM	0:28	1.36499E+11	129529760	99.91%

TSM Client Deduplication Results



Command: *DB2 backup db PERF use TSM open 8 sessions dedup_device with 18 buffers buffer 16384 parallelism 8 compress*

Start	End	Elapsed Time HH:MM:SS	Actual DB Size in bytes	Dedup Size sent to server in bytes	% Savings in bandwidth
5:32:19 PM	8:53:02 PM	3:20:43	103683256320	102876574054	
8:53:02 PM	11:54:50 PM	3:01:48	103683256320	9268429	99.9%
11:54:50 PM	3:13:01 AM	3:18:11	103683256320	2875036	99.9%
3:13:01 AM	6:26:13 AM	3:13:12	103683256320	3566182	99.9%
6:26:13 AM	9:22:23 PM	2:56:10	103683256320	3189526	99.9%

Transportable Schema



- The objective of this solution is to simplify the migration of a Sybase database solution to a DB2 database solution.
- The Sybase database architecture includes database objects called dbspaces. Each dbspace is independent and self contained; and Sybase database administrators (DBAs) can easily copy all the data in a dbspace from one database to another. When users migrate a Sybase database solution to a DB2 database, the data in the Sybase dbspaces is mapped to DB2 schemas. However, the DB2 database architecture does not facilitate copying data between DB2 databases by schema.
- Performance objective – 100 GB in under 20 minutes
 - 80GB out of 200 GB backup image consisting of 400 tables
- Customers want to ensure we minimize the amount of I/O that is done
- **Not available in pureScale**

Provide the illusion that we can restore any table space(s) into any database and then recreate all of the logical objects based off of the physical objects in the table space(s).

Essentially Oracle's Transportable Tablespaces feature on steroids (they only transport tables, indexes, views, RI constraints and triggers)

Interface will require the user to specify both the list of schemas to be transported as well as the list of table spaces (for now).

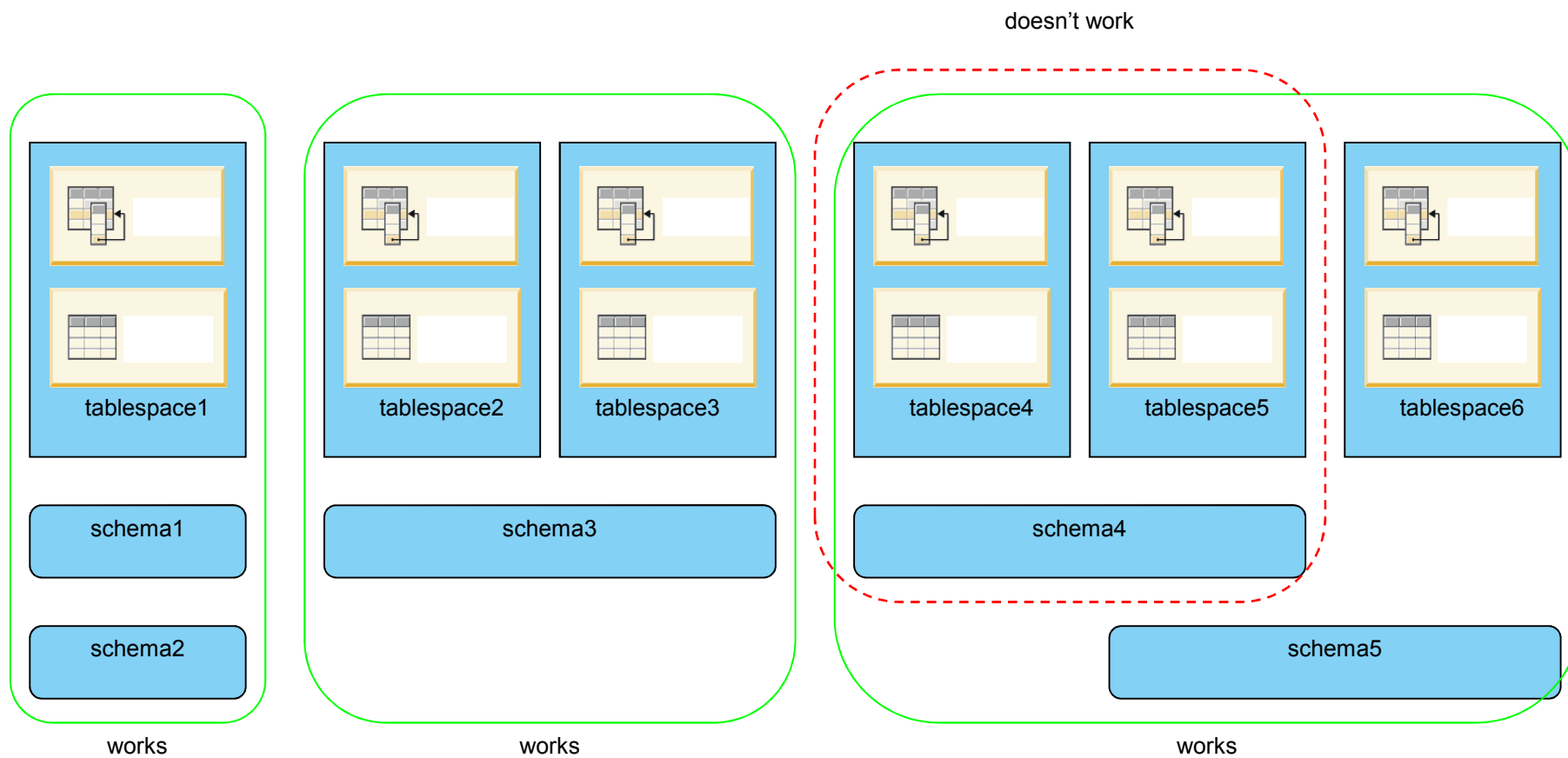
Restore will now do multiple operations

- Restore the syscatspace and specified table spaces from the backup image
- Roll them forward to a consistency point
- Validate the schemas specified
- Transfer ownership of the specified table spaces to the target DB
- Recreate the schema in the targetDB

Restore command changes

- Add “TRANSPORT INTO” option to identify this is a transport option
- Add “Stage into” option to identify which staging DB to use and to not delete it once the transport complete
- Reuse the “TABLESPACE” option to identify which table spaces to transport
- Add Schema list option to identify which schemas will be transported
 - With this release ALL schemas in the supplied table space list must be specified
 - This is added for future extensibility, we do require the call to specify the set of table spaces to transport. In the future we could drop the table space list parameter and determine it ourselves.

Transport Sets



Transportable Schema Example



Transport Schema 1 and Schema 2 to the target DB successfully

- Restore the mydata1 and myindex1 table spaces into TARGETDB

*db2 restore db originaldb tablespace ("mydata1","myindex1")
schema("schema1","schema2") transport into targetdb redirect*

db2 set tablespace containers for 2 using (path '/targetdb/mydata1')

db2 set tablespace containers for 3 using (path '/targetdb/myindex1')

db2 restore db originaldb continue

What will be transported



- Tables / CGTTs / MQTs / 'normal + stats' Views
- Generated columns
 - Expression
 - Identity
 - Row change timestamp
 - Row change token
- UDFs / UDTs + generated functions
- Constraints
 - Check
 - Foreign key
 - Unique / primary
 - Functional dependency
- Indexes
- Triggers
- Sequences
- Procedure – not external routine executable
- Object authorizations / privileges / Security / Access control / Audit
- Table Statistics + profiles / hints (plan lockdown)
- Packages

What will NOT be transported



- Created Global variables
- Aliases
- Jobs
- Index extensions
- Hierarchical tables, typed tables, typed views
- Nicknames
- Structured types
- methods
- Servers
- Wrappers
- Functional mappings & templates
- OLE DB External functions
- sourced Procedures
- XML - sysxmlstrings and sysxmlpaths collisions an issue

DPF and pureScale environments are not be supported

- New db2diag.log entry added at the end of each backup and restore operation
- Contains stats detailing where each BAR thread spent its time
- One entry per db2BM and per db2MED threads
- It was introduced in 9.7 under reg var control, and is enabled by default as of 10.1 FP2 (the reg var is no longer required).

Sample Output



```
2013-07-19-04.13.57.496158+000 E7374506A2396          LEVEL: Info
PID      : 23658576          TID     : 4113          PROC   : db2sysc 0
INSTANCE: db2pb1          NODE    : 000          DB     : PB1
APPHDL  : 0-7          APPID: *LOCAL.db2pb1.130717232706
AUTHID  : DB2PB1
EDUID   : 4113          EDUNAME: db2agent (PB1) 0
FUNCTION: DB2 UDB, database utilities, sqluxLogDataStats, probe:281
MESSAGE : Performance statistics
DATA #1 : String, 1935 bytes
```

```
Number of buffers = 30
Buffer size       = 16781312 (4097 4K pages)
```

<u>BM#</u>	<u>Total</u>	<u>I/O</u>	<u>MsgQ</u>	<u>WaitQ</u>	<u>Buffers</u>	<u>GBytes</u>
000	103510.00	34318.08	68661.04	5.49	129892	2029
001	103509.99	12716.83	61922.75	28477.03	85713	1339
002	103509.99	15396.27	71039.63	16605.96	107371	1677
003	103509.99	12022.06	63610.40	27480.43	86771	1355
004	103509.99	14991.83	59660.31	28477.52	83021	1297
005	103509.99	15170.57	59541.68	28421.50	82117	1283
006	103509.99	13501.96	61204.07	28402.28	87714	1370
007	103509.99	15359.61	59459.05	28312.37	82607	1290
008	103509.99	16304.06	58362.13	28476.03	80317	1254
009	103509.99	11367.38	63259.74	28476.81	88328	1380
TOT	1035100.00	161148.70	626720.86	243135.47	913851	14278

<u>MC#</u>	<u>Total</u>	<u>I/O</u>	<u>MsgQ</u>	<u>WaitQ</u>	<u>Buffers</u>	<u>GBytes</u>
000	103512.33	47805.32	3395.74	0.03	129893	2030
001	75049.26	31252.53	197.54	8.20	85714	1339
002	86913.43	31695.56	329.32	8.20	107372	1678
003	76041.04	29464.31	213.91	8.21	86772	1356
004	75050.68	31783.52	193.83	8.20	83022	1297
005	75099.25	31211.08	199.07	8.21	82118	1283
006	75118.45	41718.12	143.80	8.21	87715	1370
007	75207.78	33519.80	182.74	8.21	82608	1291
008	75046.60	28422.91	208.62	8.20	80318	1255
009	75046.06	30891.54	178.83	8.21	88329	1380
TOT	792084.92	337764.72	5243.45	73.90	913861	14282

Explanation



BM# - the number we assigned to an individual Buffer Manipulator. BM's READ data from the databases tablespace during a backup and place them into buffers.

MC# - the number assigned to an individual Media Controller. MC's WRITE buffers out to the target location.

Total - The total amount of time spent by the process in seconds.

I/O - The amount of time spent either reading or writing data. For the BM's this represents time reading data from tablespace, and filling the buffer. For MC it's time spent reading from buffer and sending it to the target destination.

MsgQ - This is the amount of time we spend waiting to get a buffer. For BM's it's how long is spent waiting to get an empty buffer for filling. For MC's it's time spent waiting to get a full buffer in order to write out.

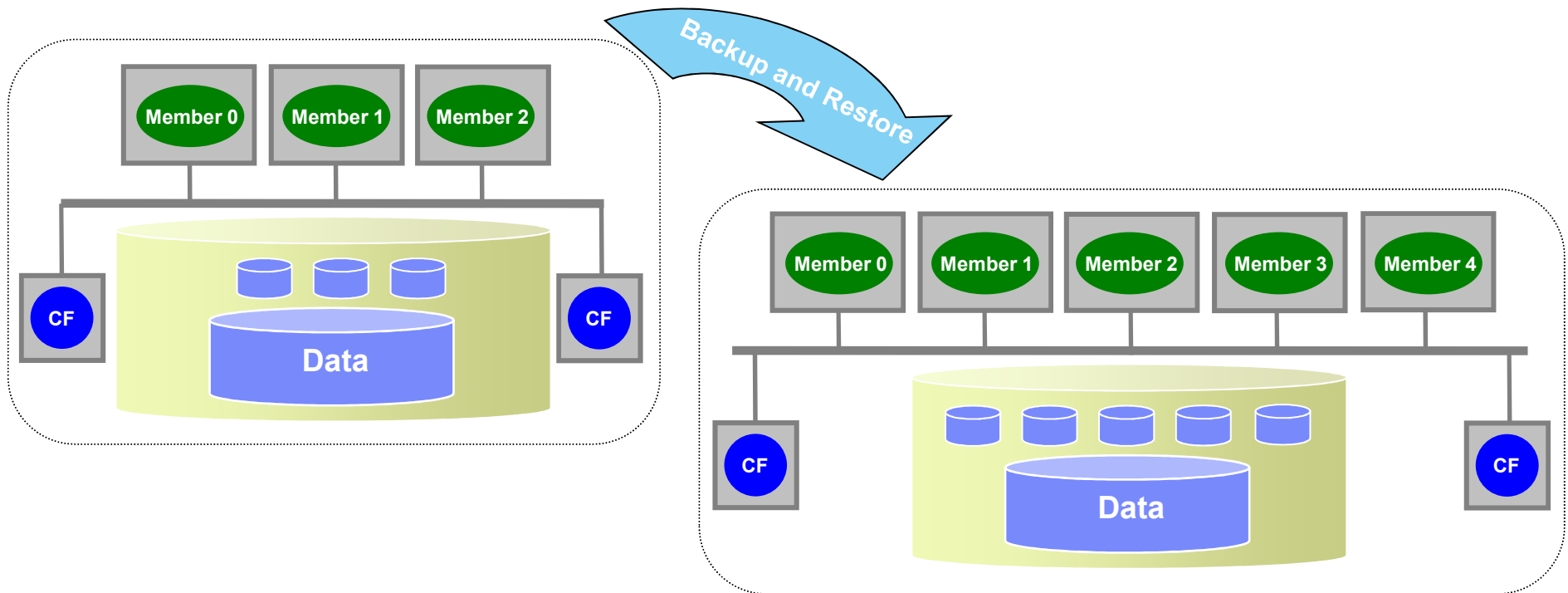
Wait Q - Amount of time spent waiting on directives from the agent overseeing the whole backup.

Buffers - The number of Buffers Processed by a particular BM or MC. A BM filled X number of buffers. An MC wrote out X number of buffers.

GBytes - The amount of data handled by a particular BM or MC in Gbytes.

Topology-Changing Backup and Restore

- Backup and restore between topologies with differing numbers of members
- Backup and restore from DB2 pureScale to non-DB2 pureScale (and vice-versa)



Agenda



- Overview

- **Technology Review**
 - What's new in Backup and Restore
 - What's new in Logging**

- Usage Scenarios

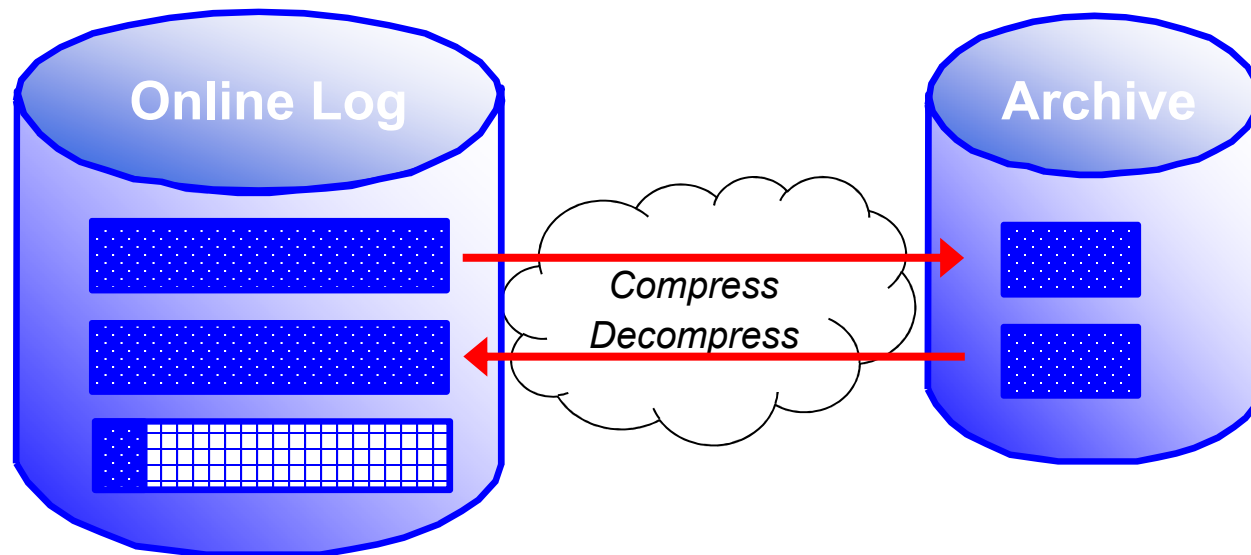
- TSM Recommendations

- Conclusion

Archival Log Compression



- Row and index compression helps reduce log record size
- However, each log record also contains a significant amount of housekeeping information (previous log record pointer, transaction ID, etc, ...)
- New optional database configuration parameter
- Simply turn it on and DB2 does the compression for you
 - logarchcompr1 database configuration parameter set to ON
- Retrieval automatically determines if decompression is needed



Agenda



- Overview

- Technology Review
 - What's new in Backup and Restore
 - What's new in Logging

- Usage Scenarios

- TSM Recommendations

- Conclusion

Why are my backups slow?



Customer XYZ DB2 Backup Analysis



Agenda



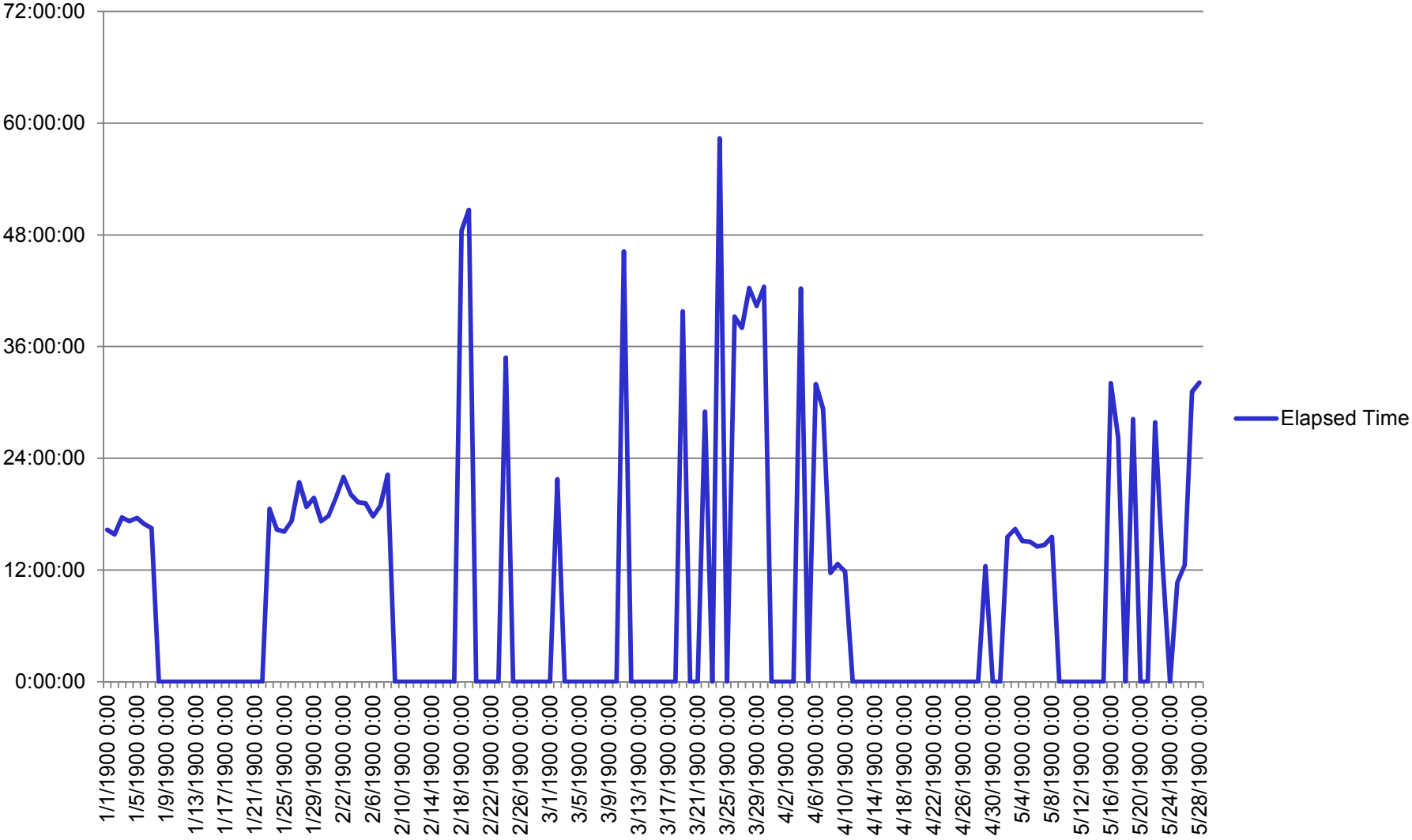
- Current status
- Observations
 - Data distribution
 - db2BM analysis
 - db2MED analysis
- Recommendations
- Update

- SAP R3 Customer
- Backing up to TSM which has an IBM ProtecTIER VTL
- Database size is growing by 150 GB per month
- Enabling the data deduplication parameter for DB2 has elongated the overall elapsed time of the backup beyond the 24 hours window
- Database Backup elapsed time has been erratic

Elapsed time for backups



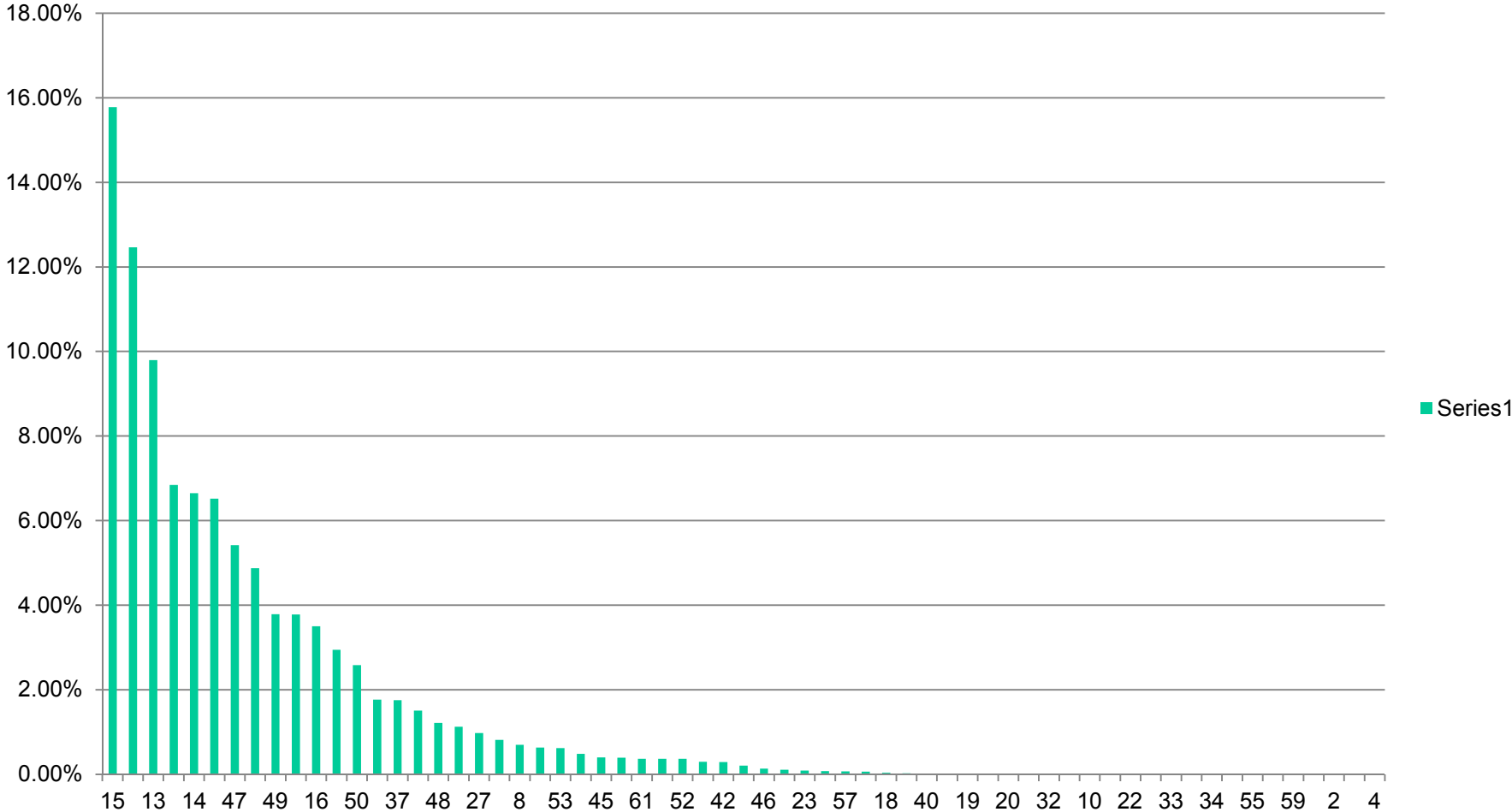
Elapsed Time



Data distribution



- The data is not evenly distributed across all of the table spaces



Backup Detailed analysis



- Options used: DEDUP_DEVICE, 30 sessions, 60 buffers, buffersize 8192
- Started on July 8, 2013 at 23:29:11
- Completed on July 10, 2013 at 10:10:48
- **Elapsed time: 34:41:37**
- Total Backup size = 14.01 TB
- Largest table space = 2.21 TB
- Time to backup largest single table space was 34:41:11
- Through-put 18MB/second over a single stream

db2_bar_stat analysis – Buffer Manipulators

BM#	Total	I/O	MsgQ	WaitQ	Buffers	GBytes	% Time on I/O	% time waiting for buffers	% time waiting for other threads
---	-----	-----	-----	-----	-----	-----			
0	124885.7	31713.7	92632.03	11.98	70748	2209	25.39%	74.17%	0.01%
1	124885.6	15774	90321.45	18361	55240	1725	12.63%	72.32%	14.70%
2	124885.6	12565.6	78445.55	33513.7	44460	1388	10.06%	62.81%	26.84%
3	124885.6	11674.2	63675.79	49269.3	31386	980	9.35%	50.99%	39.45%
4	124885.6	11759.8	61663.76	51205.5	30191	943	9.42%	49.38%	41.00%
5	124885.6	11025.8	61377.62	52229	29637	925	8.83%	49.15%	41.82%
6	124885.6	10744	57109.16	56814.8	24969	779	8.60%	45.73%	45.49%
7	124885.6	9355.42	52296.05	63039.5	22473	701	7.49%	41.88%	50.48%
8	124885.6	6080.38	48480.21	70168.3	17485	546	4.87%	38.82%	56.19%
9	124885.6	6459.55	47083.99	71185.2	17435	544	5.17%	37.70%	57.00%
10	124885.6	10038.2	42897.55	71810	15646	488	8.04%	34.35%	57.50%
11	124885.6	5868.69	42343.08	76553.8	13615	425	4.70%	33.91%	61.30%
12	124885.6	7038.35	38784.47	78960.5	11895	371	5.64%	31.06%	63.23%
13	124885.6	5033.29	34859.28	84922.3	8208	256	4.03%	27.91%	68.00%
14	124885.6	5270.65	33577.73	85969.6	7792	243	4.22%	26.89%	68.84%
15	124885.6	2954.88	33676.33	88194.4	6970	217	2.37%	26.97%	70.62%
16	124885.6	3053.31	31709.32	90072.4	5902	184	2.44%	25.39%	72.12%
17	124885.6	2017.88	31215.53	91608.1	5181	161	1.62%	25.00%	73.35%
18	124885.6	1756.7	28654	94437.6	4501	140	1.41%	22.94%	75.62%
19	124885.6	2193.6	27125.59	95535.2	3733	116	1.76%	21.72%	76.50%
20	124885.6	2523.08	25863.92	96471.6	3232	100	2.02%	20.71%	77.25%
21	124885.6	1369.08	26293.93	97195	3310	103	1.10%	21.05%	77.83%
22	124885.6	1264.64	26314.23	97280.2	3218	100	1.01%	21.07%	77.90%
23	124885.6	1458.65	26120.73	97279.9	3171	98	1.17%	20.92%	77.90%
24	124885.6	2279.59	25301.51	97279.8	2939	91	1.83%	20.26%	77.90%
25	124885.6	2438.22	25131.44	97291.1	2960	92	1.95%	20.12%	77.90%
26	124885.6	1220.39	26358.59	97280.3	3161	98	0.98%	21.11%	77.90%
27	124885.6	2121.5	25528.18	97209.1	3211	100	1.70%	20.44%	77.84%
28	124885.6	1044.03	26533.28	97281.3	3216	100	0.84%	21.25%	77.90%
29	124885.6	1984.92	25592.93	97280.9	3203	99	1.59%	20.49%	77.90%
---	-----	-----	-----	-----	-----	-----			
TOT	3746569	190082	1256967	2295712	459088	14322			

db2_bar_stat analysis – Media Controllers

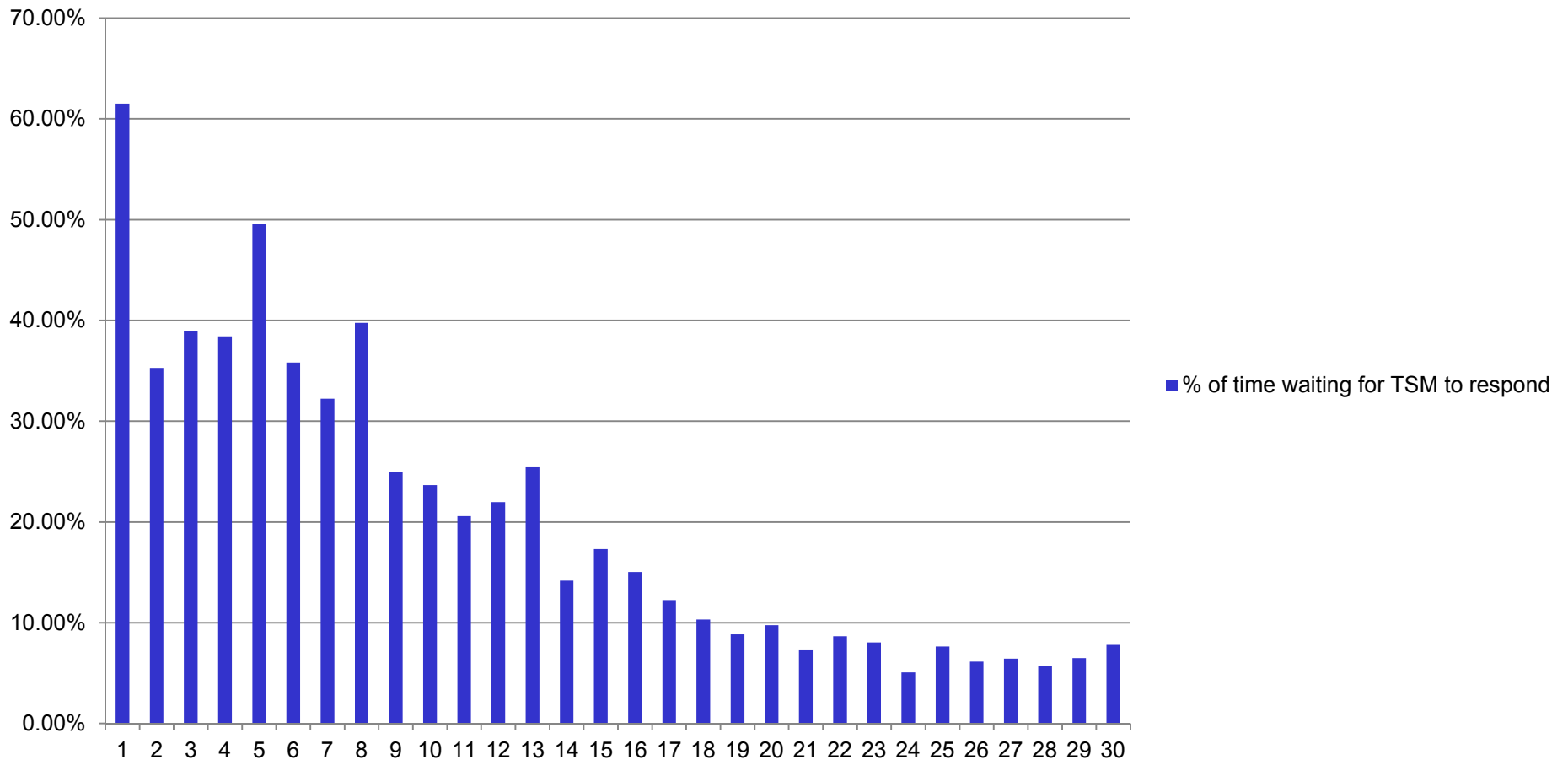
MC#	Total	I/O	MsgQ	WaitQ	Buffers	GBytes	% Time on I/O	% time waiting for buffers	% time waiting for other threads
---	-----	-----	-----	-----	-----	-----			
0	124888.9	76825.5	842.67	0.03	70749	2210	61.52%	0.67%	0.00%
1	106543.5	37585.2	543.94	9.87	55241	1726	35.28%	0.51%	0.03%
2	91390.44	35590.4	289.39	9.87	44461	1389	38.94%	0.32%	0.03%
3	75634.08	29064.1	153.2	9.87	31387	980	38.43%	0.20%	0.03%
4	73700.61	36517.5	72.37	9.87	30192	943	49.55%	0.10%	0.03%
5	72675.22	26042.5	157.67	9.87	29638	926	35.83%	0.22%	0.04%
6	68093.13	21956	152.8	9.87	24970	780	32.24%	0.22%	0.04%
7	61863.59	24598.9	72.07	9.87	22474	702	39.76%	0.12%	0.04%
8	54734.77	13687.5	88.64	9.87	17486	546	25.01%	0.16%	0.07%
9	53717.47	12710.5	88.35	9.87	17436	544	23.66%	0.16%	0.08%
10	53095.89	10927	100.67	9.87	15647	488	20.58%	0.19%	0.09%
11	48348.48	10626.5	70.37	9.87	13616	425	21.98%	0.15%	0.09%
12	45944.12	11687.7	55.88	9.87	11896	371	25.44%	0.12%	0.08%
13	39980.77	5671.74	52.12	9.87	8209	256	14.19%	0.13%	0.17%
14	38933.72	6738.3	50.5	9.87	7793	243	17.31%	0.13%	0.15%
15	36711.24	5515.8	44.08	9.87	6971	217	15.02%	0.12%	0.18%
16	34830.15	4265.31	41.27	9.87	5903	184	12.25%	0.12%	0.23%
17	33296.2	3433.6	35.27	9.87	5182	161	10.31%	0.11%	0.29%
18	30469.33	2692.67	35.07	9.87	4502	140	8.84%	0.12%	0.37%
19	29370.44	2867.86	27.1	9.87	3734	116	9.76%	0.09%	0.34%
20	28437.13	2091.62	27.27	9.87	3233	101	7.36%	0.10%	0.47%
21	27707.99	2397.45	25.55	9.87	3311	103	8.65%	0.09%	0.41%
22	27629.07	2221.45	26.64	9.87	3219	100	8.04%	0.10%	0.44%
23	27626.2	1401.62	28.24	9.87	3172	99	5.07%	0.10%	0.70%
24	27634.34	2110.49	26.77	9.87	2940	91	7.64%	0.10%	0.47%
25	27614.46	1695.52	27.51	9.87	2961	92	6.14%	0.10%	0.58%
26	27626.69	1778.02	27.7	9.87	3162	98	6.44%	0.10%	0.56%
27	27693.92	1572.03	28.13	9.87	3212	100	5.68%	0.10%	0.63%
28	27621.05	1790.85	25.49	9.87	3217	100	6.48%	0.09%	0.55%
29	27626.77	2153.48	26.16	9.87	3204	100	7.79%	0.09%	0.46%
---	-----	-----	-----	-----	-----	-----			
TOT	1451440	398217	3242.89	286.26	459118	14331			

DB2 Media Controllers Behavior



db2med threads a lot of time waiting for TSM to respond.

% of time waiting for TSM to respond

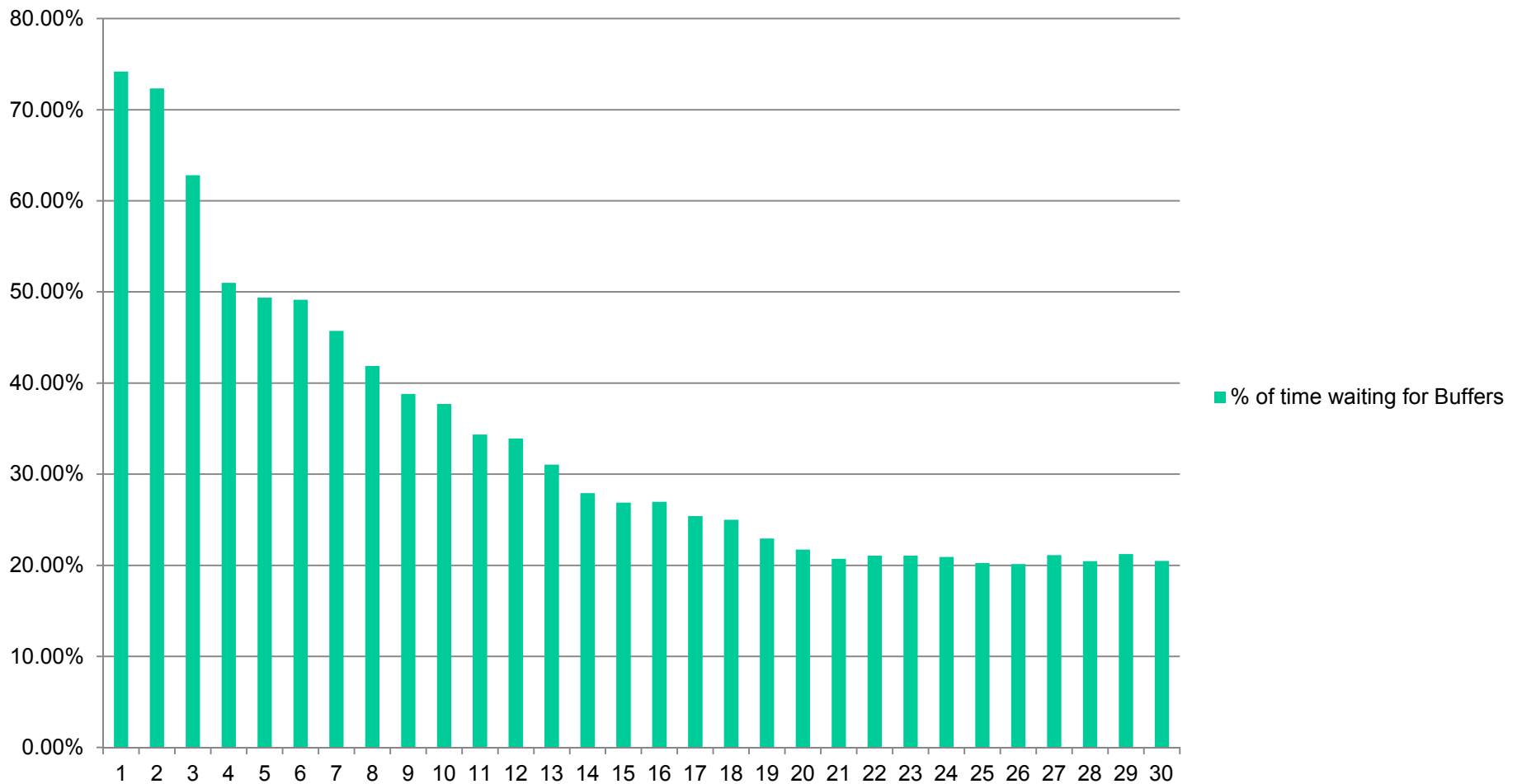


DB2 Buffer Manipulators Behavior



db2BM threads are spending a lot of time waiting for buffers to get freed by db2med threads

% of time waiting for Buffers

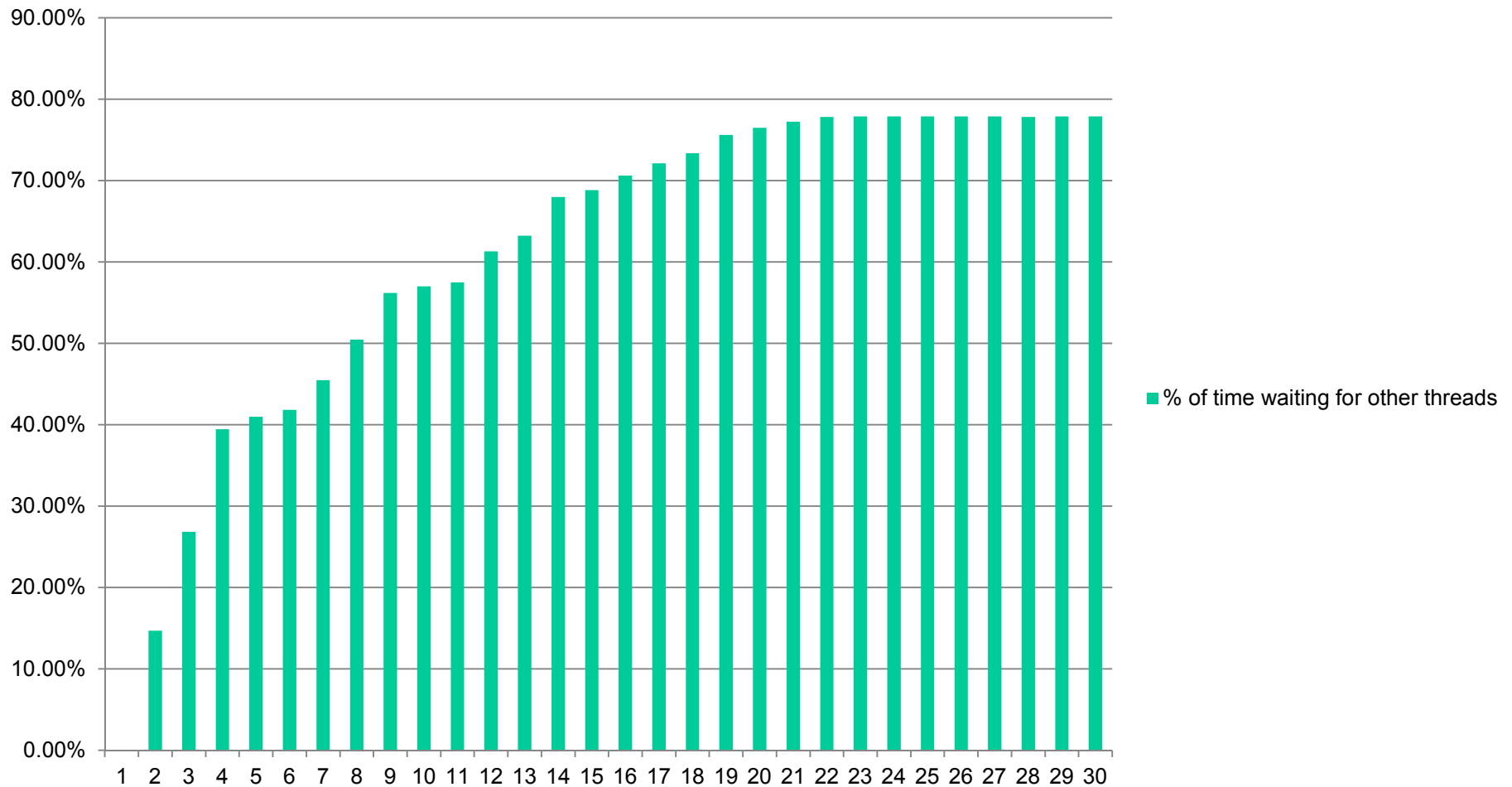


DB2 Buffer Manipulators Behavior



db2BM threads are spending a lot of time waiting for other threads to complete

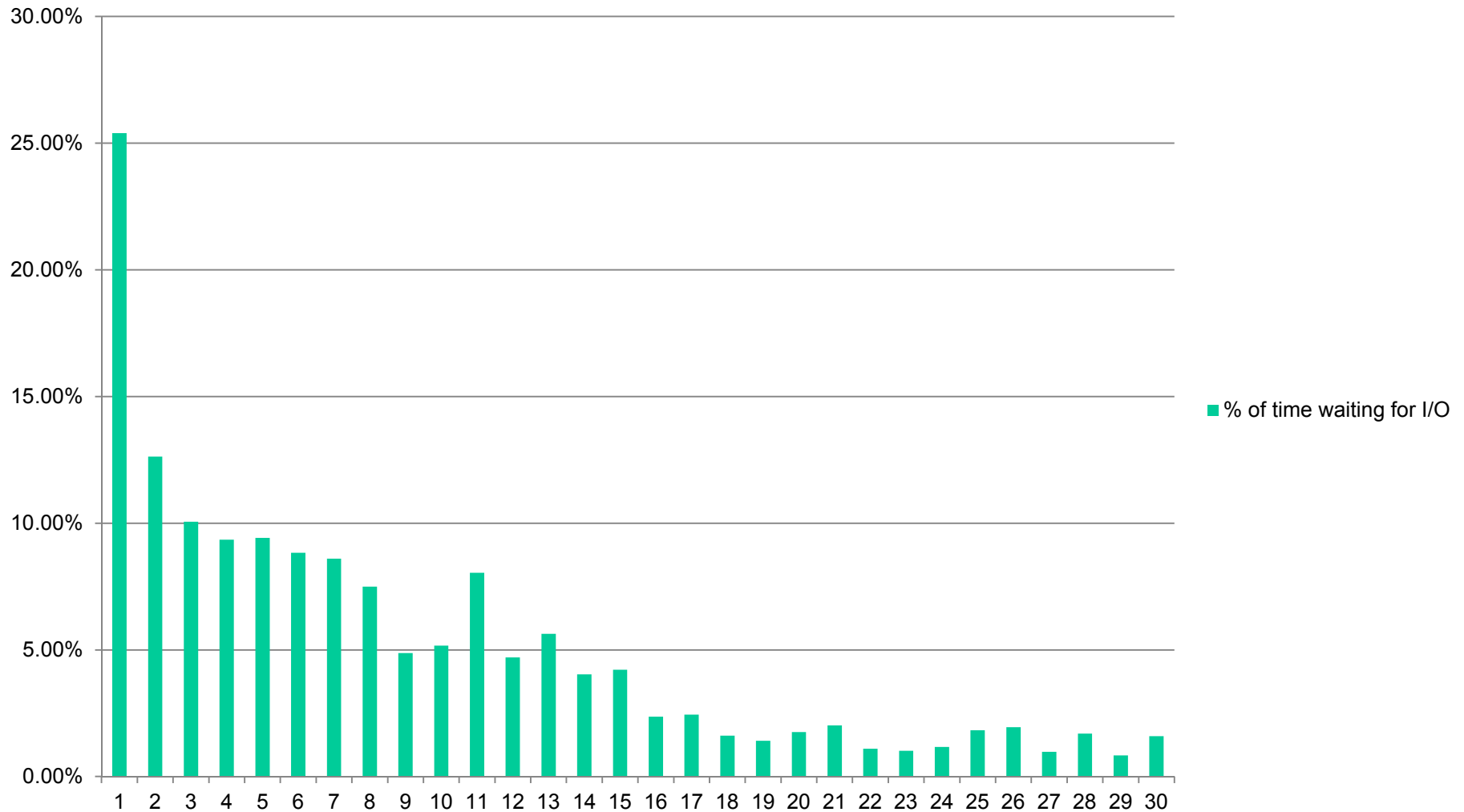
% of time waiting for other threads



DB2 Buffer Manipulators Behavior



db2BM threads somewhat impacted by slow I/O
% of time waiting for I/O



Recommendations



- Analyze TSM Server / PT environment to determine why TSM is responding slowly – Max of 18.5 MB/second over a single session
- Implement DB2 row level compression on all large tables
- Customer should consider implementing an archive and/or purge process to reduce the overall size of the database
- Distribute the data more evenly within the database across the table spaces
- Decrease the number of sessions to 10, this will reduce the amount of time the db2bms are spending waiting
- Decrease the size of the buffer to so that TSM can respond quicker, use a buffer size of 4097
- Ensure there are sufficient buffers, with 10 sessions you should be using 30 buffers
- Consider increasing the NUM_IOSERVERS db cfg parameter to $10 * 14 = 140$ (this may affect runtime performance)

Changes Implemented



- Decreased the number of sessions from 30 to 10
 - Most sessions were idle
- Decreased the size of the buffer to 8192 to 4097
 - Allow quicker response from TSM
- Decreased number of buffers from 60 to 30
 - Ensure there are sufficient buffers for all threads
- Changed NUM_IOSERVERS db cfg parameter to $10 \times 14 = 140$
 - Ensure there are enough prefetchers for the backup utility

Post Changes Backup Analysis



- Options used: DEDUP_DEVICE, 10 sessions, 30 buffers, buffersize 4097
- Started on July 17, 2013 at 23:27:07
- Completed on July 19, 2013 at 04:13:57
- **Elapsed time: 28:46:50**
- Total Backup size = 13.95 TB
- Largest table space = 1.98 TB
- Time to backup largest single table space was 28:45:03
- Through-put 17MB/second over a single stream

Analysis of DB2_BAR_STATS output



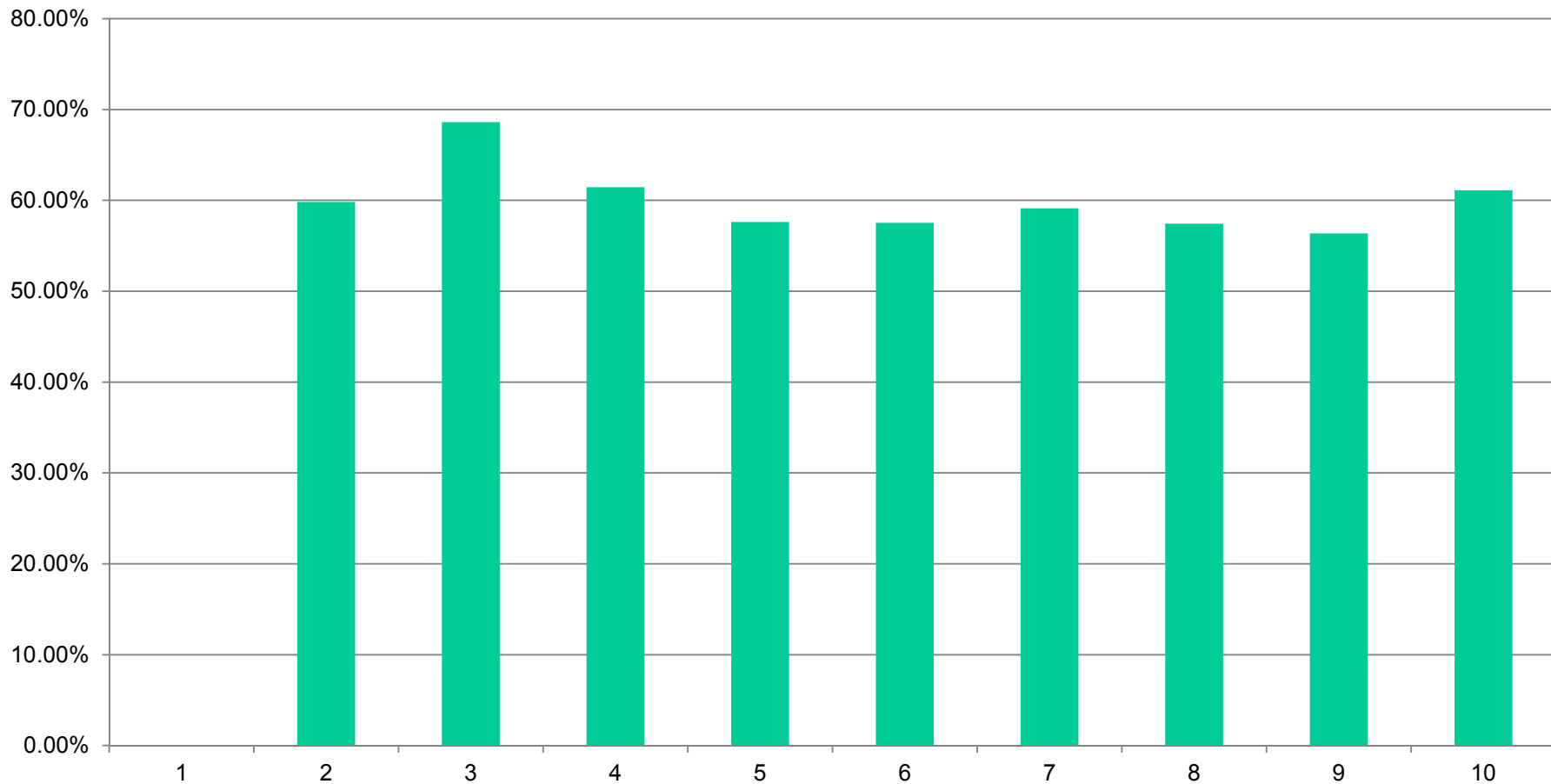
BM#	Total	I/O	MsgQ	WaitQ	Buffers	GBytes	% Time on I/O	% time waiting for buffers	% time waiting for other threads
---	-----	-----	-----	-----	-----	-----			
0	103510	34318.08	68661.04	5.49	129892	2029	33.15%	66.33%	0.01%
1	103509.99	12716.83	61922.75	28477.03	85713	1339	12.29%	59.82%	27.51%
2	103509.99	15396.27	71039.63	16605.96	107371	1677	14.87%	68.63%	16.04%
3	103509.99	12022.06	63610.4	27480.43	86771	1355	11.61%	61.45%	26.55%
4	103509.99	14991.83	59660.31	28477.52	83021	1297	14.48%	57.64%	27.51%
5	103509.99	15170.57	59541.68	28421.5	82117	1283	14.66%	57.52%	27.46%
6	103509.99	13501.96	61204.07	28402.28	87714	1370	13.04%	59.13%	27.44%
7	103509.99	15359.61	59459.05	28312.37	82607	1290	14.84%	57.44%	27.35%
8	103509.99	16304.06	58362.13	28476.03	80317	1254	15.75%	56.38%	27.51%
9	103509.99	11367.38	63259.74	28476.81	88328	1380	10.98%	61.11%	27.51%
---	-----	-----	-----	-----	-----	-----			
TOT	1035099.9	161148.7	626720.8	243135.42	913851	14274			

DB2 Buffer Manipulators Behavior



db2BM threads are spending a lot of time waiting for buffers to get freed by db2med threads

"% of time waiting for buffers to get freed"

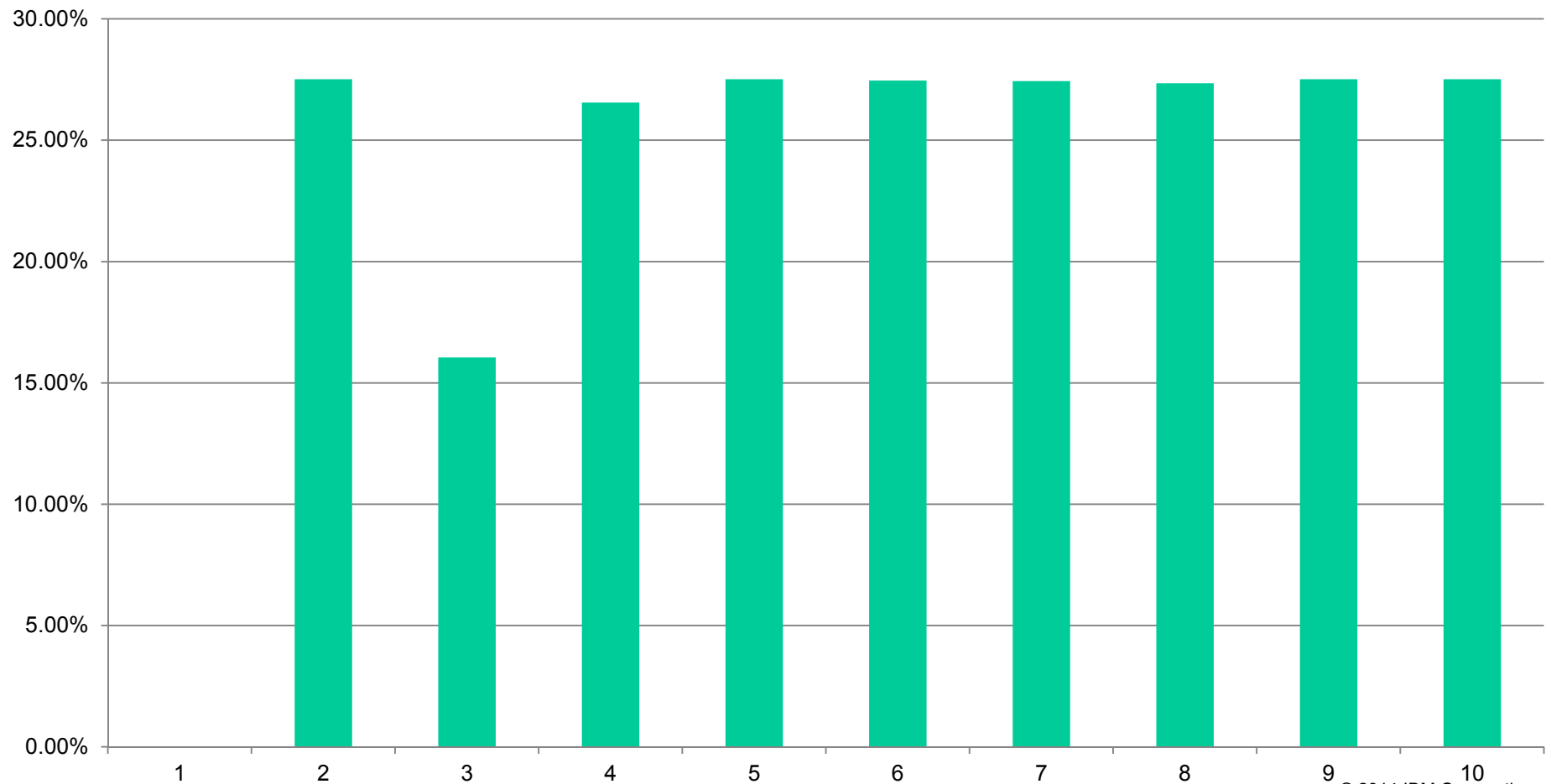


DB2 Buffer Manipulators Behavior



db2BM threads are spending a lot of time waiting for other threads to complete

"% of time waiting for other threads to complete"

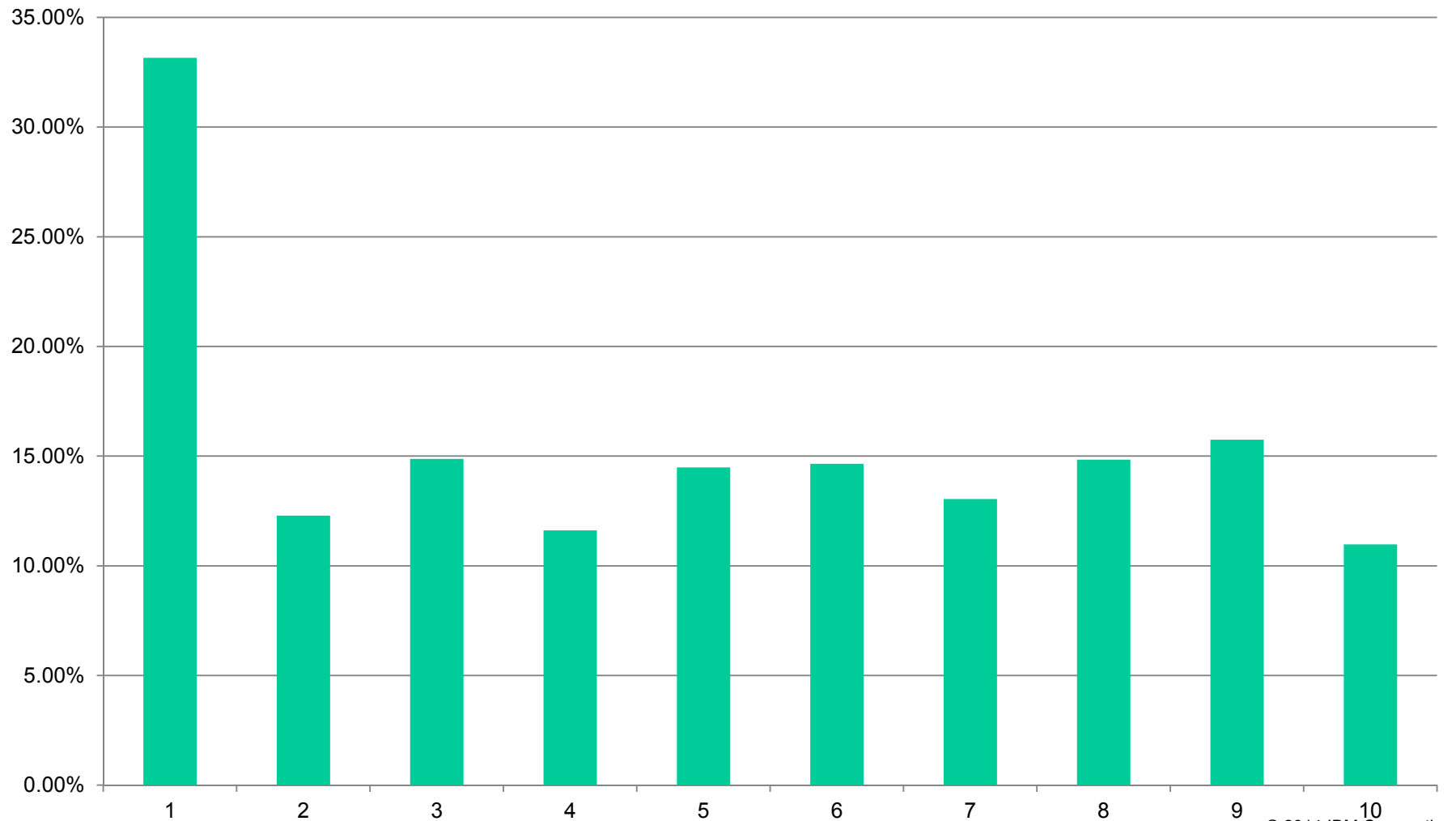


DB2 Buffer Manipulators Behavior



db2BM threads somewhat impacted by slow I/O

"% of time waiting for I/O"



Analysis of DB2 BAR_STATS output



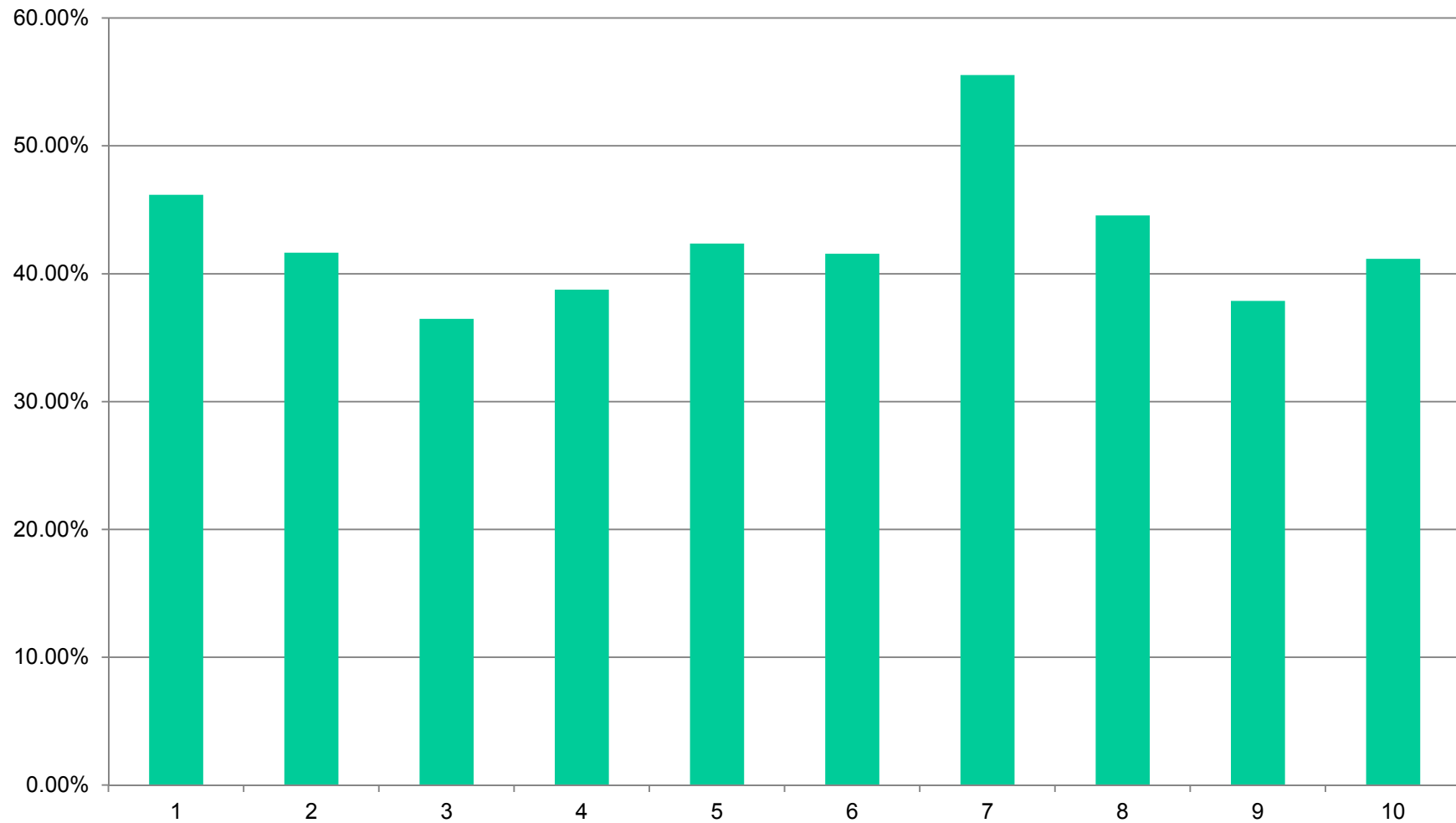
MC#	Total	I/O	MsgQ	WaitQ	Buffers	GBytes	% time on I/O	% time waiting for buffers	% time waiting for agent
---	-----	-----	-----	-----	-----	-----			
0	103512.33	47805.32	3395.74	0.03	129893	2030	46.18%	3.28%	0.00%
1	75049.26	31252.53	197.54	8.2	85714	1339	41.64%	0.26%	0.01%
2	86913.43	31695.56	329.32	8.2	107372	1678	36.47%	0.38%	0.01%
3	76041.04	29464.31	213.91	8.21	86772	1356	38.75%	0.28%	0.01%
4	75050.68	31783.52	193.83	8.2	83022	1297	42.35%	0.26%	0.01%
5	75099.25	31211.08	199.07	8.21	82118	1283	41.56%	0.27%	0.01%
6	75118.45	41718.12	143.8	8.21	87715	1370	55.54%	0.19%	0.01%
7	75207.78	33519.8	182.74	8.21	82608	1291	44.57%	0.24%	0.01%
8	75046.6	28422.91	208.62	8.2	80318	1255	37.87%	0.28%	0.01%
9	75046.06	30891.54	178.83	8.21	88329	1380	41.16%	0.24%	0.01%
---	-----	-----	-----	-----	-----	-----			
TOT	792084.88	337764.7	5243.4	73.88	913861	14279			

DB2 Media Controllers Behavior



db2med threads a lot of time waiting for TSM to respond.

% of time waiting for TSM to respond

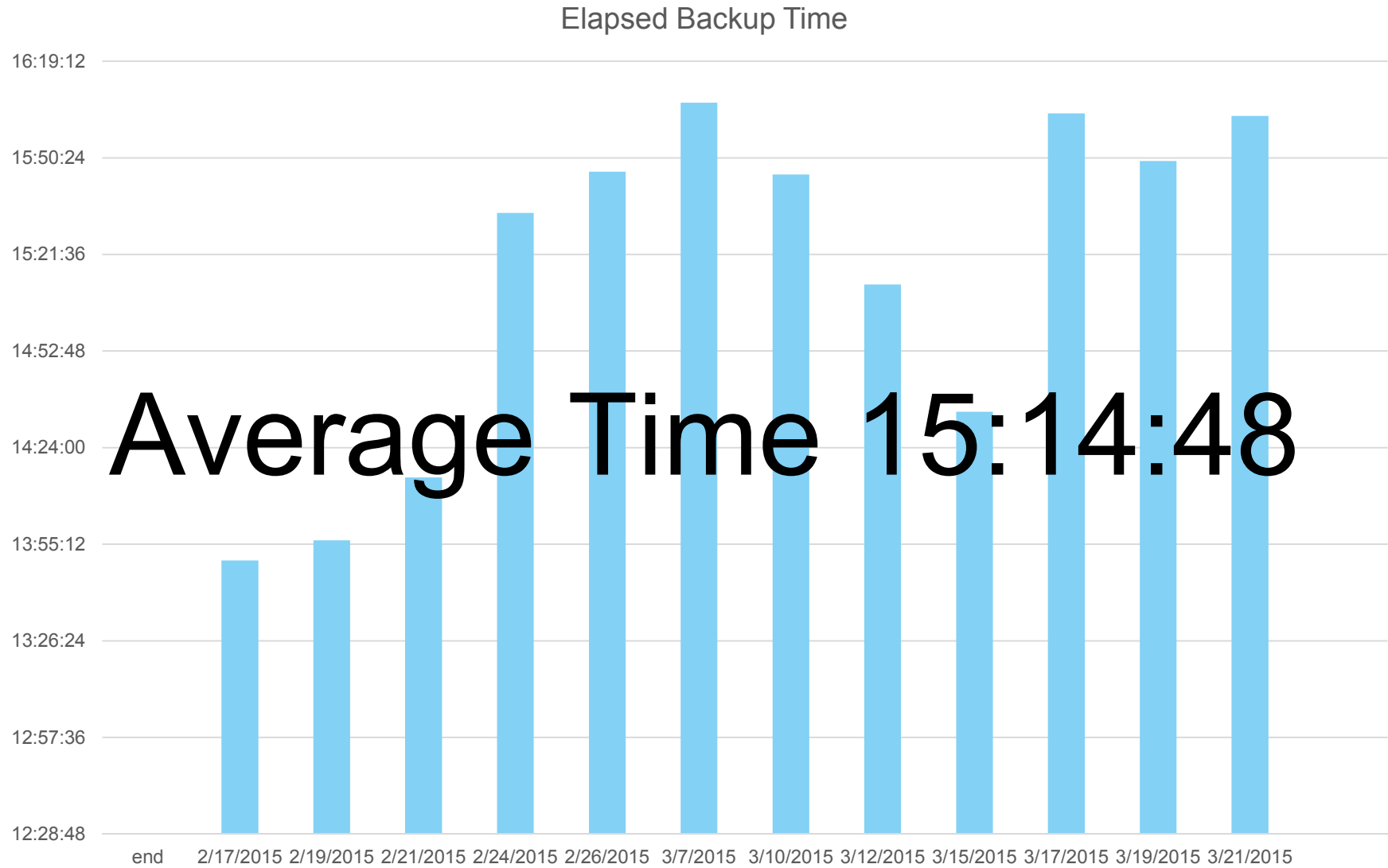


- **Improved overall elapsed time by 6+ hours**
- Observed increase % of wait time for TSM per db2MED thread, however each thread must now process more data. **Overall totals decreased by 15%**
- Observed increased time waiting for buffers, this is expected due to observation #2. **Overall totals decreased by 50%**
- Observed increase % of time waiting for I/O. **Overall totals decreased by 15%.**

- Examine layout of protecTIER meta data layout, currently residing on old storage – Gen 2 XIV storage
- Distribute the data more evenly across the table spaces
- Work with PT, TSM and Network administrators to determine why lan-free through-put is low
- Implement data purging process to reduce overall DB size
- Implement row / adaptive compression to reduce overall DB size

- Distribute the data more evenly across the table spaces using the `admin_table_move` procedure
- Implement adaptive compression to reduce overall DB size on most of the largest tables
- **RESULTS:**
 - Average backup time 15:14:48
 - Compared to 34:41:37 when we started
 - 1.78X faster 😊

Current Elapsed Backup Times



Latest DB2_BAR_STATS output

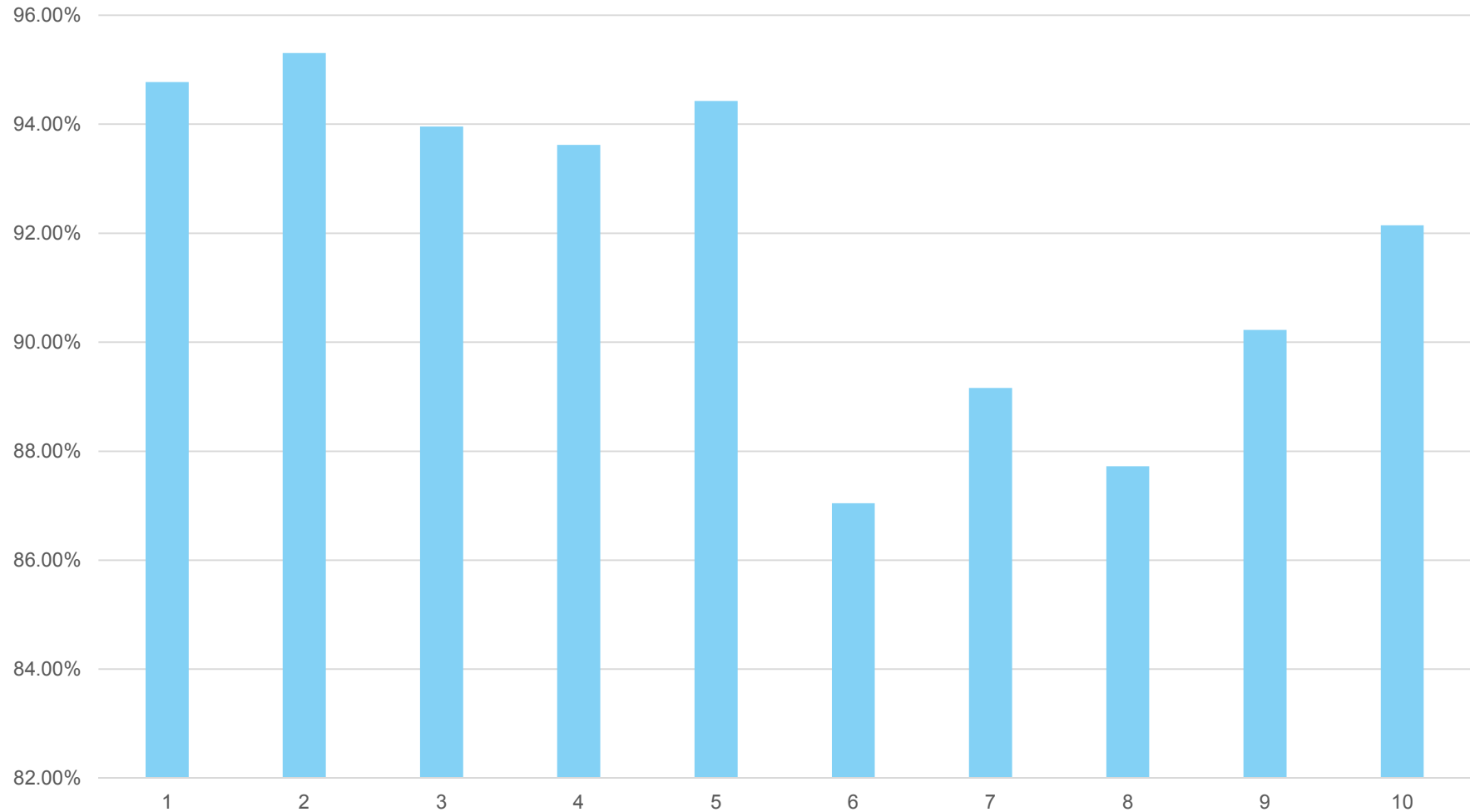


BM#	Total	I/O	MsgQ	WaitQ	Buffers	GBytes	% Time on I/O	% time waiting for buffers	% time waiting for other threads
0.00	57173.28	2517.47	54184.42	251.75	67941	1061	4.40%	94.77%	0.44%
1.00	57173.27	2264.38	54488.35	198.62	69391	1084	3.96%	95.30%	0.35%
2.00	57173.27	2988.10	53717.40	255.06	66566	1039	5.23%	93.96%	0.45%
3.00	57173.27	3180.01	53526.62	255.13	66131	1033	5.56%	93.62%	0.45%
4.00	57173.27	2770.56	53986.24	201.48	66689	1041	4.85%	94.43%	0.35%
5.00	57173.27	6955.30	49765.47	254.92	61101	954	12.17%	87.04%	0.45%
6.00	57173.27	5982.53	50977.12	8.85	63771	996	10.46%	89.16%	0.02%
7.00	57173.27	6640.54	50155.10	176.72	61725	964	11.61%	87.72%	0.31%
8.00	57173.27	5125.88	51583.75	254.97	64750	1011	8.97%	90.22%	0.45%
9.00	57173.27	4012.76	52682.19	255.07	69715	1089	7.02%	92.14%	0.45%
TOT	571732.76	42437.58	525066.71	2112.61	657780	10275			
MC#	Total	I/O	MsgQ	WaitQ	Buffers	GBytes			
0.00	56932.20	26766.03	178.94	0.03	67941	1061	47.01%	0.31%	0.00%
1.00	56987.95	26433.48	178.16	10.91	69392	1084	46.38%	0.31%	0.02%
2.00	56934.07	26269.79	177.43	10.94	66567	1040	46.14%	0.31%	0.02%
3.00	56933.48	22438.73	211.08	10.89	66132	1033	39.41%	0.37%	0.02%
4.00	56991.76	20771.30	232.07	10.94	66690	1042	36.45%	0.41%	0.02%
5.00	56931.78	19318.64	240.55	10.94	61102	954	33.93%	0.42%	0.02%
6.00	57189.49	20111.70	236.53	10.90	63773	996	35.17%	0.41%	0.02%
7.00	57012.08	20191.93	255.09	10.90	61726	964	35.42%	0.45%	0.02%
8.00	56933.05	19882.94	293.40	10.94	64751	1011	34.92%	0.52%	0.02%
9.00	56930.52	26750.77	255.70	10.94	69716	1089	46.99%	0.45%	0.02%
TOT	569776.43	228935.36	2258.99	98.39	657790	10280			

Db2bm's waiting for Buffers to get freed up



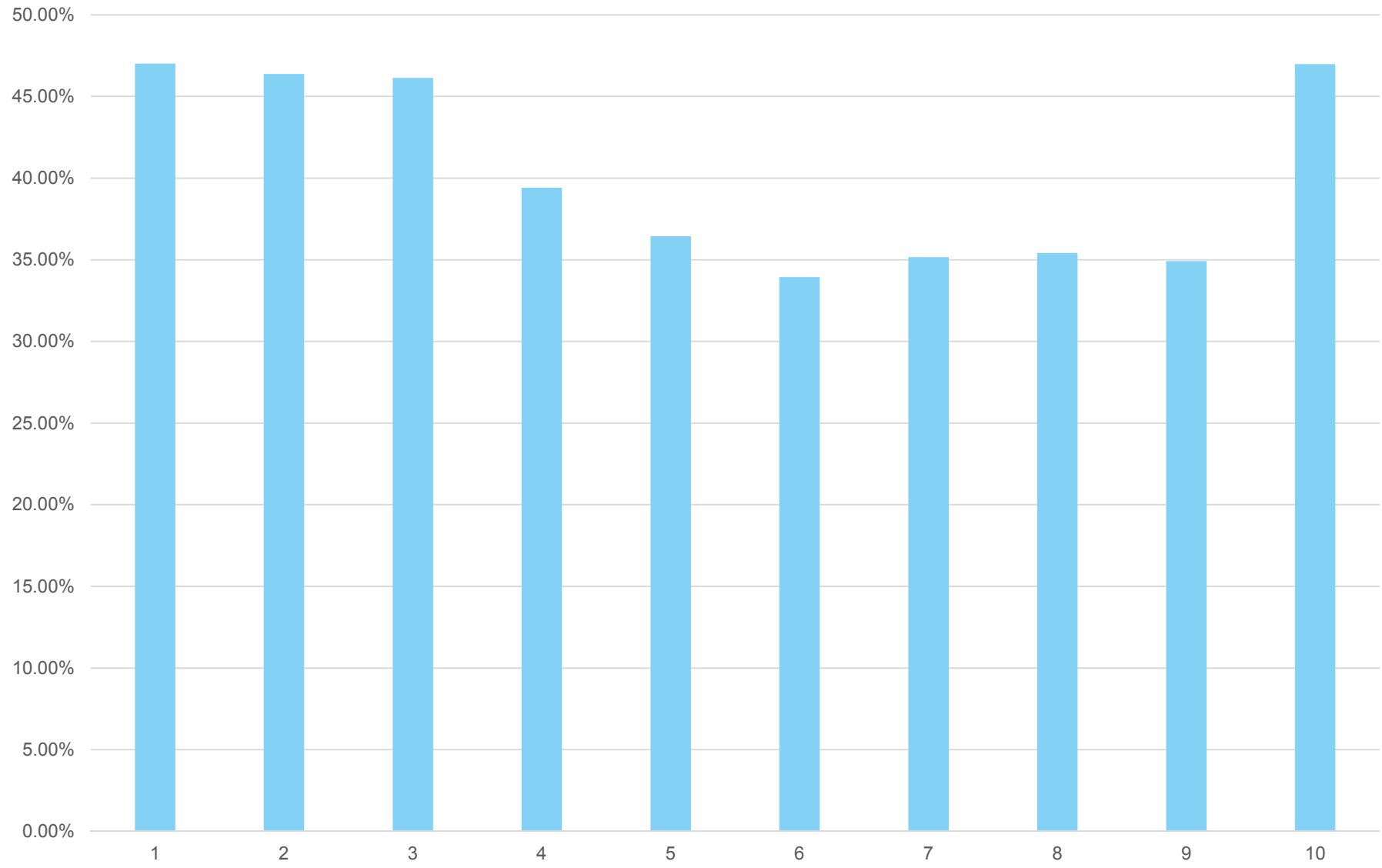
% time db2bms are waiting for buffers



Db2med's waiting for TSM to respond



% of time db2med waiting for TSM to respond



Agenda



- Overview

- Technology Review
 - What's new in Backup and Restore
 - What's new in Logging

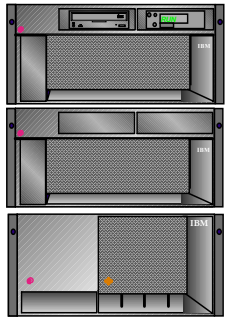
- Usage Scenarios

- TSM Recommendations

- Conclusion

Restoring to a Different Node Using TSM

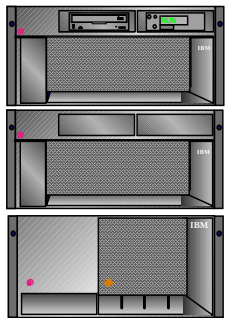
NODE: PRIMARY, DB2 Instance:db2inst1



db2 backup db sample use tsm
db2adutl grant user myinst on all for database sample



TSM Server



db2 restore db sample use tsm options
"-fromnode=PRIMARY - fromowner=db2inst1"



NODE: STANDBY, DB2 Instance:myinst

Exploiting Proxy Authority Support



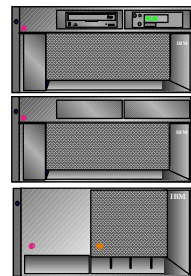
- Provides the ability for 1 or more nodes to appear as a single node to the TSM server.
- Very useful in a HADR environment where either machine may be the primary at any one time
- 2 Step procedure
 - Issue the grant proxy node from TSM Admin command line
 - Set database configuration parms
 - Set Vendoropts to “-asnode=xxx”; logarchopts to “-asnode=xxx”

Proxy Node Example

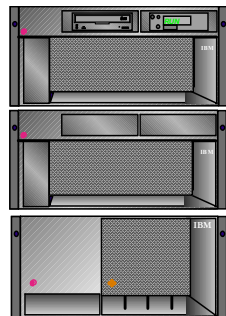
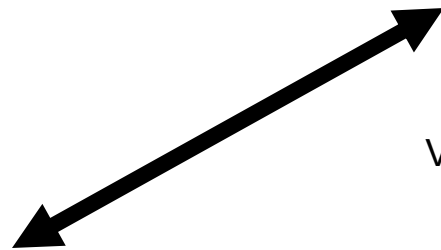


TSM Server

Nodes registered = dale, austin, mcinnis
Grant proxynode target=mcinnis agent=dale
Grant proxynode target=mcinnis agent=austin

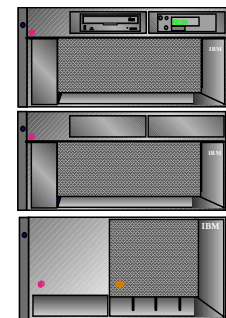
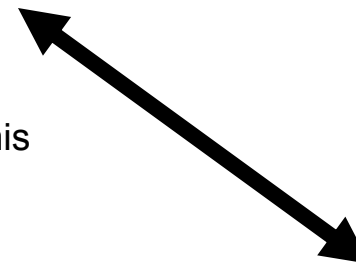


Virtual node mcinnis



Hostname: Dale

Vendoropts="-asnode=mcinnis"
Logarchopts="-asnode=mcinnis"



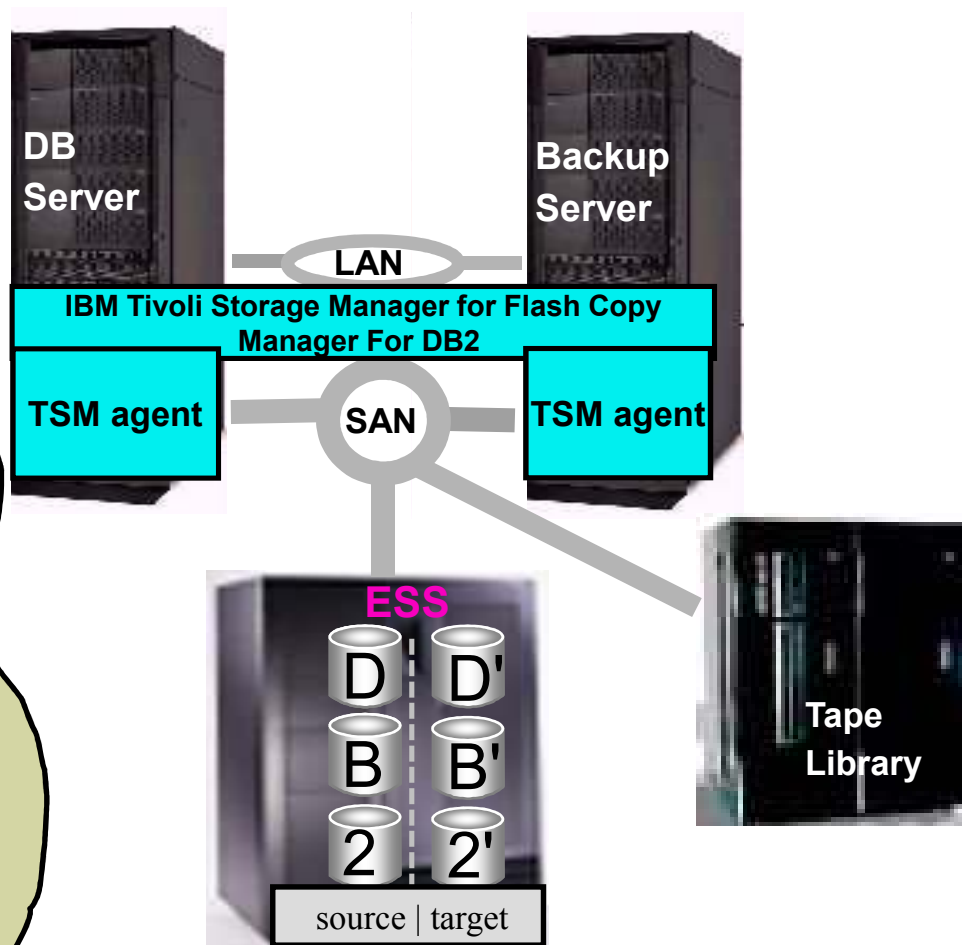
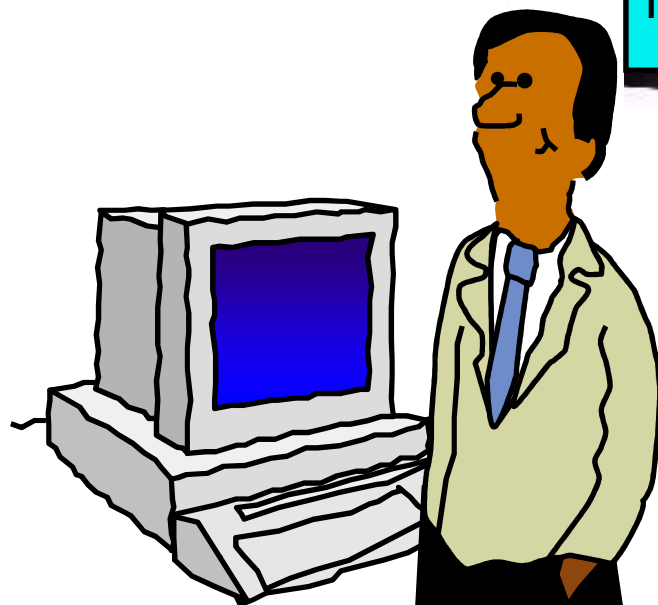
Hostname: Austin

Vendoropts="-asnode=mcinnis"
Logarchopts="-asnode=mcinnis"

IBM Tivoli Storage Manager for Flash Copy Manager For DB2



Product-based
end-to-end
solution



TSM Server configuration recommendations



- Ensure commtimeout and idletimeout values are set large
 - Required when doing “redirected restore” since we must allocate and initialize all of the disk containers
 - Values should be set longer than it takes to initialize all of the containers – e.g. 60000

TSM Server configuration recommendations



Backup Copy Group

- Used by DB2 backups and load copies
- All db2 backup images are uniquely named
- TSM retention policy not enforceable
- Db2adutl utility will delete objects immediately

Archive Copy Group

- Used by DB2 logs
- As of V 8.2 all archived log files will be uniquely named – includes the log chain ID
- Retention controlled by TSM policy definition
- Can use db2adutl to delete objects immediately

Agenda



- Overview

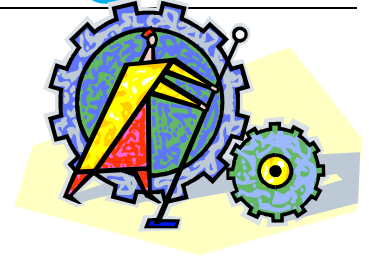
- Technology Review
 - What's new in Backup and Restore
 - What's new in Logging

- Usage Scenarios

- TSM Recommendations

- Conclusion

Best Practices – Your database design



- Leverage intelligent database design practices!
 - All referential tables are in the same table space, or set of table spaces.
 - All staging tables (for load) are in a separate table space
 - Active vs. In-active data – separated by table space
 - Range partitioning – further identify active vs. inactive by designing ranges
 - Local vs. Global Indexes – if you have global indexes (ie. Prior to v9.7), then keep those in a table space that you back up with it's matching data table space. Rebuilding your indexes can be time consuming!
- Goal: you want to be set up to restore only what you need.

Best Practices: Backup recommendations



- Backup Images:
 - Online database backups
 - Online table space backups
 - Incremental backups
 - Compression is good *
- Use the Single System View option. Let DB2 manage the backups and timestamps for you.
- Always use lan-free option to write to tape drives directly via SAN
 - DO NOT archive directly to tape!
 - configure FAILARCHPATH in case primary-log-destination becomes unavailable
- Include the log records in each backup image – it's easier that way!
- Operating System files backed up twice per week

Best Practices: Backup recommendations cont'd



- Example:
 - Full online table space backup of hot* and warm table spaces twice a week, include logs
 - Full online database backup of catalog partition daily, logs included
 - Full online database backup of database partitions on quarterly or monthly basis
 - Full table space backup after adding a new table space

Best Practices - Logging



- Use archive logging – not circular logs for your production environment
- Goal: ease of maintenance
 - Include logs in your backup image
 - Use TSM to handle your archiving for you
- Small transaction logs:
 - Reduce risk since logs contain fewer transactions
 - Increases the number of logs required, since large transactions may span multiple log files, plus more overhead in system resources.
- Large transaction logs:
 - Increase risk since logs contain more transactions
 - Benefit: reduces the amount of system resources used, as less log switching takes place
- Recommendation: Start with 50 primary logs, of about 50MB each.

Best Practices - Recovery recommendations



- For table space recovery, use the restore command with the REBUILD option
 - Use the REBUILD option (v9.5) to get the subset of the system you need up and running immediately
- Use the log manager, not user exits
- Use db2adutl to retrieve log files from tape to disk – keep ahead of the log files you need .

- **If a DB2 online backup is not feasible due to conflicts with other utilities then consider SNAPSHOTS in DB2 9.5**

OR

- **IBM Tivoli Storage Manager for Advanced Copy Services For DB2**
 - http://www.ibm.com/developerworks/tivoli/library/t-acbdb2_1/index.html
- **When using TSM always use Lan-Free option**
- **Review SAP 20 TB BI Benchmark Results**
 - Full backup in < 7 hours
 - <http://www.ibm.com/support/techdocs/atmastr.nsf/WebIndex/WP101012>
- **Best Practices for backup and recovery in ISAS**
 - <http://www.ibm.com/developerworks/data/bestpractices/isasrecovery/index.html>

More Information



- DB2 LUW Best Practices:
<https://www.ibm.com/developerworks/data/bestpractices>
- Best Practices: Building a Recovery Strategy for an IBM Smart Analytics System Database
<http://www.ibm.com/developerworks/data/bestpractices/sasrecovery/index.html>
- Best Practices: Multi-Temperature Data Management
<http://www.ibm.com/developerworks/data/bestpractices/multitemperature/index.html>
- Article: DB2 instance recovery for IBM Smart Analytics System
<http://www.ibm.com/developerworks/data/library/techarticle/dm-1010db2instancerecovery/index.html?ca=drs->

- **On-line communities, User Groups, Technical Forums, Blogs, Social networks, and more**
 - Find the community that interests you ...
 - **Information Management** bit.ly/InfoMgmtCommunity
 - **Business Analytics** bit.ly/AnalyticsCommunity
 - **Enterprise Content Management** bit.ly/ECMCommunity
- **IBM Champions**
 - Recognizing individuals who have made the most outstanding contributions to Information Management, Business Analytics, and Enterprise Content Management communities
 - ibm.com/champion

THANK
YOU