Prepare to Repair.

Thomas Baumann
Swiss Mobiliar

Session Code: F10
October 16 2013, 9:45 AM- 10:45 AM | Platform: DB2 for zOS

INTERNATIONAL
DB2 USERS GROUP
Independant • Not-for-Profit • User Run
Founded 1988

At least once during your career as a DBA, you will bei exposed to a situation where your well prepared recovery jobs fail. Better to have a "plan B" ready to execute rather than falling back to reading the manuals and the diagnosis guide. This presentation explains the usage of the Repair utility and shows some typical use cases where Repair is very suitable, for example assisting when performing a recovery of dropped objects

Objectives

- Get an overview (review?) of recovery concepts
- Get an overview of Repair usage
- Learn to use Repair to assist in a recovery of dropped objects
- Learn to use Repair to skip log ranges during recover
- Learn to apply a recovery methodology

Switzerland's most personal
insurance company

Swiss **Mobiliar**
Insurance & Pensions

Swiss Mobiliar mission statement: principles and vision

Principles

Swiss Mobiliar

- is a mutually structured insurance company
- is independent and has a strong capital base
- is geared towards long-term success.

Vision

Swiss Mobiliar

- stands for security, reliability and customer proximity
- is Switzerland's most personal insurance company
- is and aims to stay Switzerland's no. 1 in household, business and pure risk life insurance.

Agenda

- What It Needs to Recover
  - Personal Resources
  - Technical Resources
  - Procedures, Techniques, Methodology
  - A 5-minute Recovery Test For You (and your peers once you'll be back)
- DB2 Components for Recovery
  - What's behind an update?
- Point-in-time Recovery: A Use Case
  - How does it work exactly?
- Another Use Case: Recovery of a dropped tablespace
  - Where REPAIR enters the game
- Summary

Disclaimer

- The Information contained in this presentation has not been submitted to any formal Swiss Mobiliar or other review and is distributed on an 'as is' basis without any warranty either expressed or implied. The use of this information is the user's responsibility.

- The procedures, results and measurements presented in this paper were run in either the test and development environment or in the production environment at Swiss Mobiliar in Berne, Switzerland. There is no guarantee that the same or similar results will be obtained elsewhere. Users attempting to adapt these procedures and data to their own environments do so at their own risk. All procedures presented have been designed and developed for educational purposes only.

Content in a Nutshell

- Review the concepts used for recovery and restart
- How you can use DSN1PRNT, DSN1LOGP and REPAIR
- A recovery methodology for every recovery scenario

What does it need to recover (1/2)?

- Human Resources
  - Know How
  - Experience

- Technical Resources
  - Image Copy
  - Log
  - etc.

- Tools and Procedures
  - Which risks should be covered by standard procedures?

- *A Methodology to control any recovery scenario*

Recovery Control Methodology: A 10-Point-Approach for every recovery scenario.

1. **Save** (Copy, DSN1COPY, VSAM Repro, IDCAMS) the concerned objects
2. Set a defined **time limit** for the end of the recovery process
3. Apply **standard (predefined) procedures** to recover (Recover, Rebuild, Restart, Reorg).

4. Problem **escalation** if time limit or result not reached

5. In Parallel:
   5a) **Analyze** problem reason as   5b) Develop **different**
        far as necessary to be able   **approaches** to attack
        to repair   the problem
6. **Decide** between the different problem solution approaches
7. **Implement** your selected approach
8. **Save** after completion
9. **Information**
10. **In-depth analysis** and lessons learned

The perception of recovery

| Problem Type	Budget	Management Attention	Tools	No. of events/year
Application	**$ 0.1 Mio**	small	Log Analysis Recovery	13
Data Integrity				5
Data Access				8
Medium	**$ 0.5 Mio**	medium	n,	1
Disaster	**$ 2.5 Mio**	High (ORM)	Dual Site	0

In many shops, recovery is a synonym for disaster recovery. Considerable resources are spent there. But aren't there other risks to your databases?

IDUG
Leading the DB2 User
Community since 1988

A Recovery / Restart Fitness Check For You (1/2)

1. Catalog and directory tablespaces are image copied on a daily frequence. Does this guarantee that in case of recovery only active logs and no archived logs need to be accessed?   `yes` `no`

2. Is there a guarantee that – for any DML update statement – the log is always wirtten to disk before the data is casted out from bufferpool to disk?   `yes` `no`

3. Does the DB2 catalog store information about the most recent timestamp when base and clone tables' contents habe been switched (EXCHANGE DATA)?   `yes` `no`

4. Is there a utility or SQL statement which allows to inspect log ranges on SYSLGRNX (in order to display log ranges when a tablespace was opened for write access)?   `yes` `no`

5. If you use the RECOVER LOGONLY recovery option, does the log scan start at the CLSN (Commit Log Sequence Number, which means the oldest open unit of work of any tablespace)?   `yes` `no`

A Recovery / Restart Fitness Check For You (2/2)

6. Is ,Recover Index' always faster than ,Rebuild Index' if an index image copy is available?   **yes**   **no**

7. A Space Map Page is reported as ,broken page'. Unfortunately, all availabe image copies are already „infected" with this broken page. Can this scenario be recovered without data loss?   **yes**   **no**

8. Does the zparm value ,DEFER ALL' accelerate DB2 restart processing significantly? (more precisely, does this option decrease the BLR (Backward Log Recovery) phase which rollbacks any open transactions at time of DB2 abend?)   **yes**   **no**

9. Does an *incremental* image copy write all changed pages since the last *full* image copy, and not just the pages changed since the last *incremental* copy?   **yes**   **no**

10. Does DB2 offer the possibility to recover a single tablespace to any arbitrary point in time (for example to a timestamp with uncommitted data changes of this tablespace) while ending up with transactionally consistent data?   **yes**   **no**
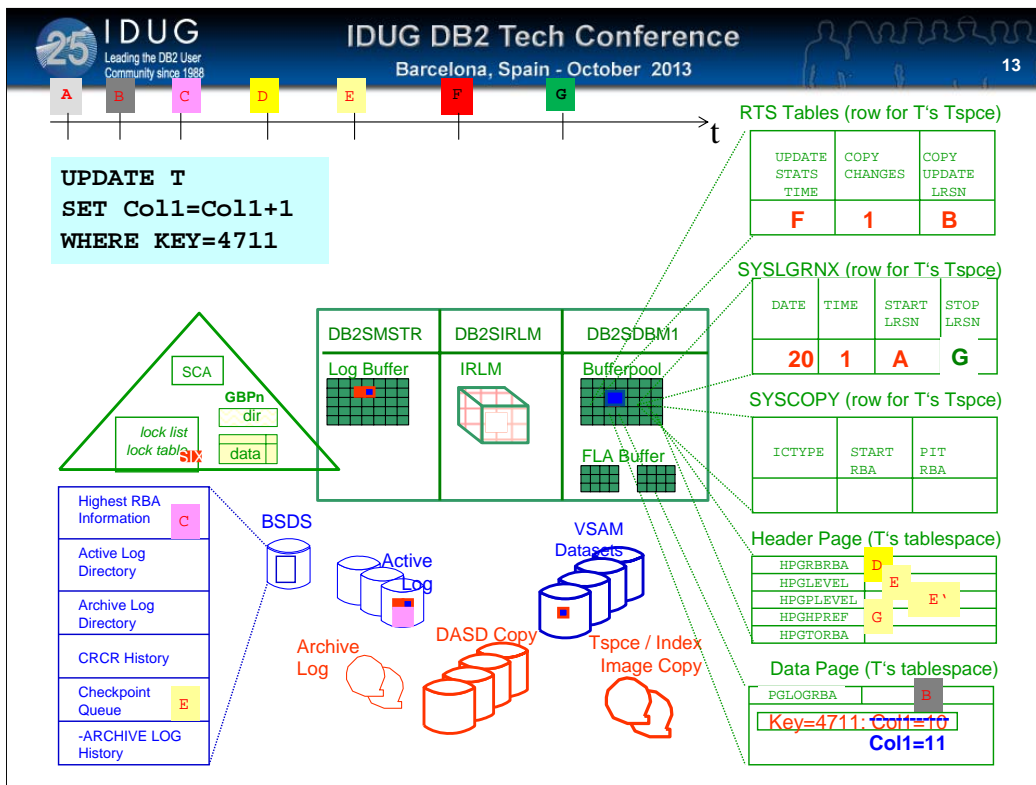
Data Page Offset:

X'01'	PGLOGRBA	Log RBA of the last page update

Header page offsets:

X'10'	HPGHPREF	current high-used page in pageset.
X'18'	HPGTORBA	RBA to which RECOVER utility has recovered pageset
X'62'	HPGRBRBA	Recover BASE RBA (LRSN). Used as start point for LOGONLY recover.

var.	HPGLEVEL	Level ID (down-level detection)
var.	HPGPLEVEL	Previous level ID (down-level detection)

Question

All indexes should be set to COPY YES to allow performing index
image copies:

    a)   *Absolutely, because image copies might be taken without previously executing an ALTER INDEX COPY YES command.*

    b)   *No, only if rebuilding the index takes longer than the SLA (service level agreement) allows, and this index is supposed to be image copied.*

    c)   *No, only indexes with more than a billion (10^9) rows.*

    d)   *Yes, because a LOGONLY recovery becomes possible with COPY YES only.*

Recovery Options

- Standard Recovery (to current)
  - RECOVER TABLESPACE X.Y

- Point-in-time Recovery
  - RECOVER TABLESPACE X.Y TOLOGPOINT X'…..' followed by REBUILD INDEX(ALL) TABLESPACE X.Y

- Some additional options
  - CLONE: Clone-tables only
  - LOGRANGES NO: no lookup in SYSLGRNX table
  - LOGONLY: LOGAPPLY starts at HPGRBRBA point
  - RESTOREBEFORE X'…': avoid application of more recent image copies
  - TOLAST[FULL]COPY: Point-in-time Recover to last [full] copy
  - Details see manual: http://publib.boulder.ibm.com/infocenter/dzichelp/v2r2/index.jsp?topic=%2Fcom.ibm.db2z9.doc.ugref%2Fsrc%2Fugref%2Fdb2z_ugref.htm
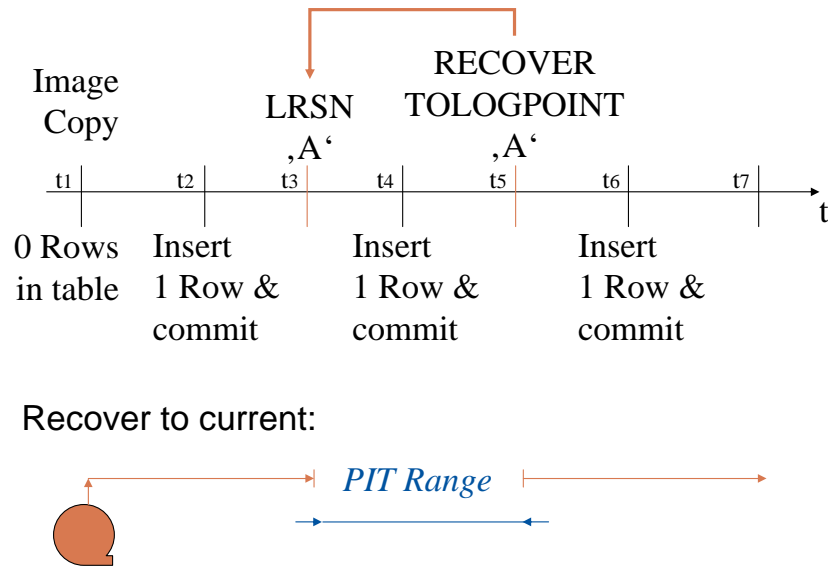
A Recovery Example

RECOVER
Image   LRSN   TOLOGPOINT
Copy   ‚A'   ‚A'

t1   t2   t3   t4   t5   t6   t7

t

0 Rows   Insert   Insert   Insert
in table   1 Row &   1 Row &   1 Row &
           commit   commit   commit

Recover to current:

PIT Range

Table T consists of how many rows at time t_7 ?

Let us start at timepoint t_1: an image copy has been taken. Afterwards (timepoints t_2 and $t_{4)}$), a single row has been inserted at each of these timepoints. At t_5, a point-in-time recovery to timepoint t_3 was performed, which means that

- the most recent image copy before t_3 will be loaded, and the log records until timepoint t_3 will be applied.

- a row will be inserted in SYSCOPY in order to persist the information

  that the log interval between t_3 and t_5 is invalid. This interval is called a

  „PIT Range' , which means an interval that will be skipped by any further

   recovery operation..

After performing this recovery operation, the number of rows of table T is 1 (the one row inserted at timepoint t_2).

At timepoint t_6 we insert another row into table T. Any recovery (to current timepoint) performed at timepoint t_7 will end up with a table consisting of 2 rows: Recovery restores the image copy, applies the logrecords until t_3 and between t_5 and t_7 . The log interval between t_3 and t_5 (PIT Range) will be skipped.
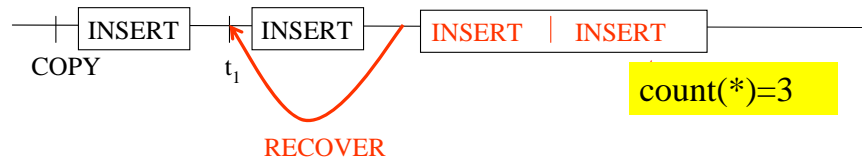
A Point-in-time Recovery Use Case
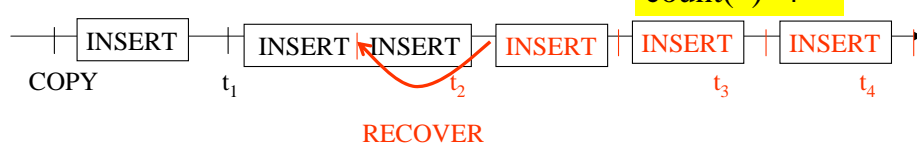
SQL DML
Stmts   Transaction
(followed by commit)
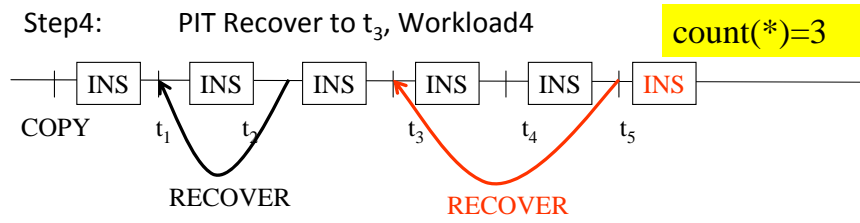
- Step 1: Image Copy, Workload1

    INSERT | INSERT
    COPY t_1 count(*)=2
    count(*)=0

- Step2: PIT Recover to timestamp t_1, Workload2

    INSERT | INSERT | INSERT | INSERT
    COPY t_1 count(*)=3
    RECOVER

- Step3:PIT Recover to timestamp t_2, Workload3   count(*)=4

    INSERT | INSERT INSERT | INSERT | INSERT | INSERT
    COPY t_1 t_2 t_3 t_4
    RECOVER

- Step4: PIT Recover to t_3, Workload4

count(*)=3

COPY t_1 t_2 t_3 t_4 t_5

RECOVER

RECOVER

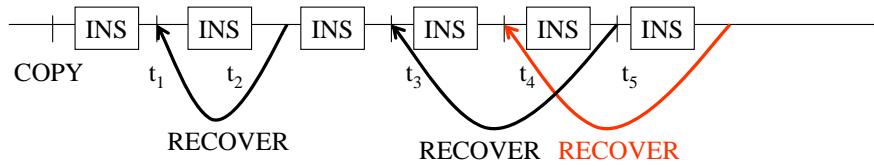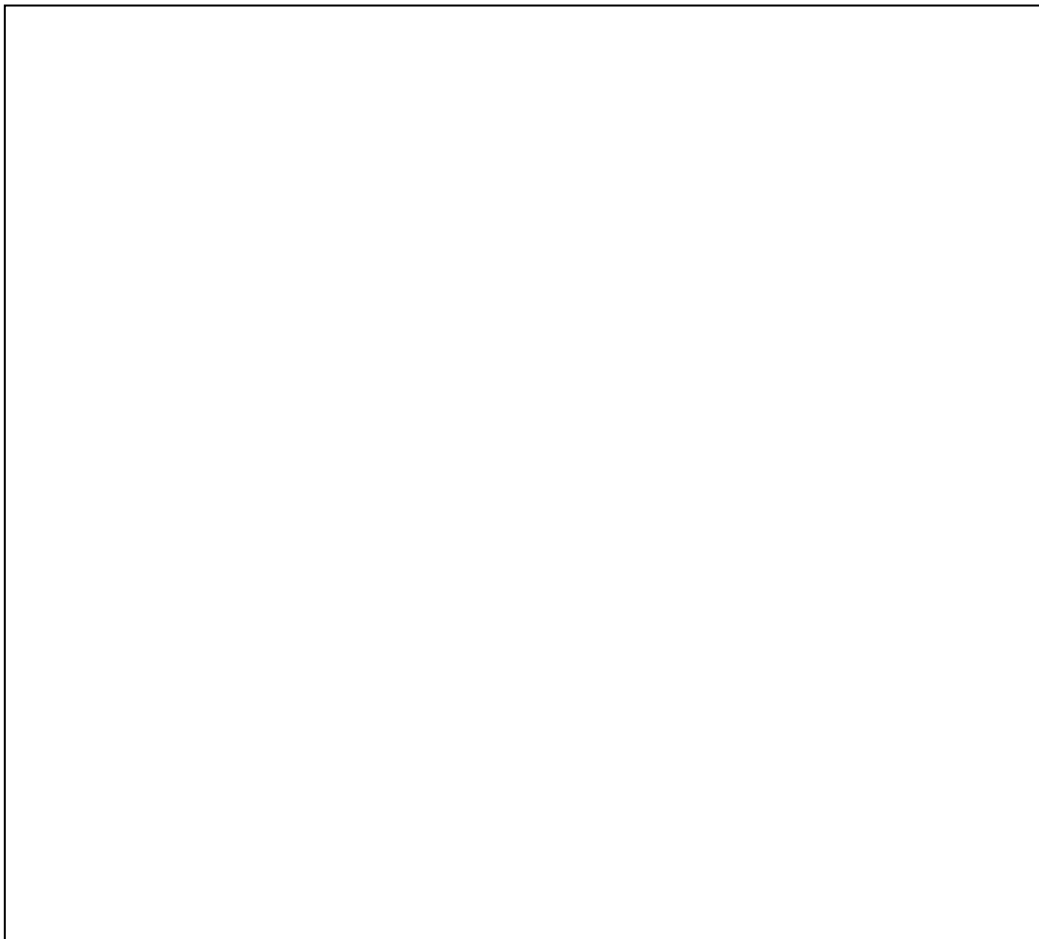- Step5: Let us have a look at SYSCOPY for this tablespace

```
SELECT
TIMESTAMP, ICTYPE,
HEX(PIT_RBA)as T1,HEX(START_RBA)as T2
FROM SYSIBM.SYSCOPY
WHERE DBNAME ='DB900664'
AND TSNAME   ='SRESERV'
ORDER BY TIMESTAMP DESC
```
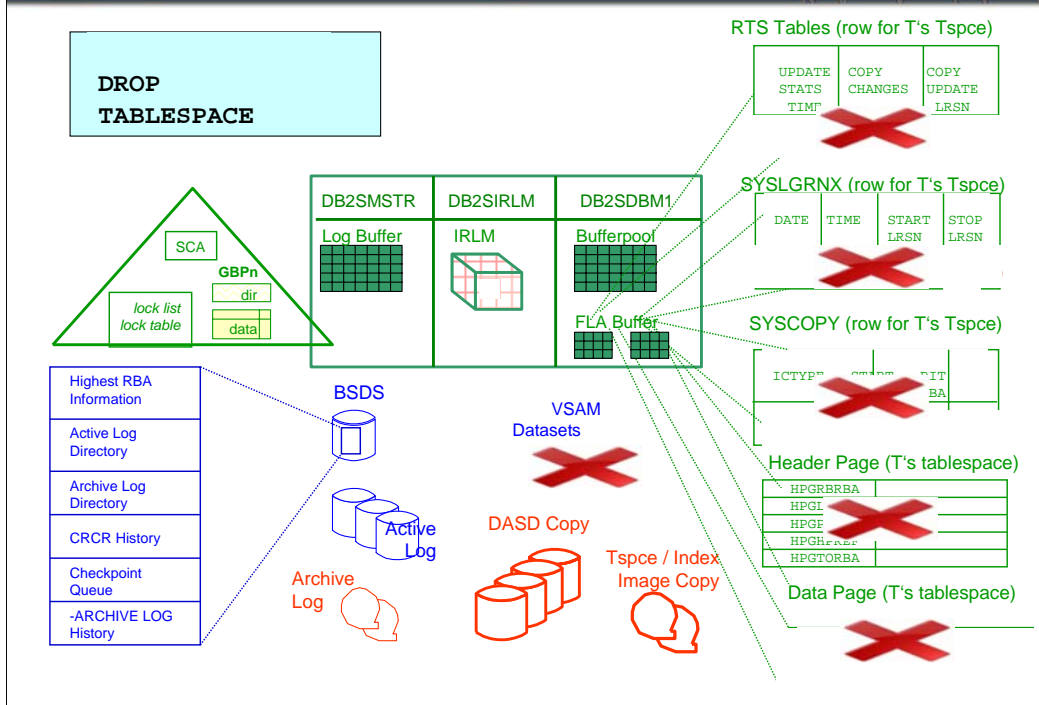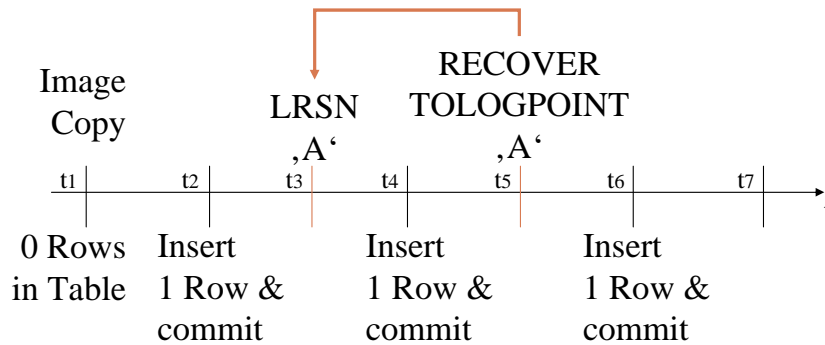
- Step6: PIT Recover to t_4. Does it work?   count(*)=3

INS   INS   INS   INS   INS   INS

COPY t_1 t_2 t_3 t_4 t_5

RECOVER   RECOVER   RECOVER

- Step7: Image Copy

DROP
TABLESPACE

RTS Tables (row for T's Tspce)

UPDATE STATS TIME	COPY CHANGES	COPY UPDATE LRSN

DB2SMSTR	DB2SIRLM	DB2SDBM1
Log Buffer	IRLM	Bufferpool

SCA

GBPn
dir

lock list
lock table
data

FLA Buffer

SYSLGRNX (row for T's Tspce)

DATE	TIME	START LRSN	STOP LRSN

SYSCOPY (row for T's Tspce)

ICTYPE	START	PIT BA

Highest RBA Information
Active Log Directory
Archive Log Directory
CRCR History
Checkpoint Queue
-ARCHIVE LOG History

BSDS

VSAM
Datasets

Active
Log

DASD Copy

Archive
Log

Tspce / Index
Image Copy

Header Page (T's tablespace)

HPGRBRBA
HPGL
HPGF
HPGHRKEF
HPGTORBA

Data Page (T's tablespace)

Recovery of dropped objects – 3 available options

- Recreate Objects with same PSID/OBID
  - → DSN1COPY → RECOVER LOGRANGES NO
- Disaster Recovery Site or Cloning
  - → PIT Recovery (Cat/Dir+dropped Object)
- Stop your world
  - → Save Cat/Dir/BSDS/Log (DASD Save) → Cond. Restart
  - → Recover Cat/Dir → Recover dropped Object
  - → Unload → Restore saved Cat/Dir/BSDS/Log
  - → Recreate dropped Object und Reload

Dropped Objects Recovery: Theory and Practise

Image Copy

RECOVER
LRSN TOLOGPOINT
,A' ,A'

| t1 | t2 | t3 | t4 | t5 | t6 | t7 |

t

0 Rows
in Table

Insert
1 Row &
commit

Insert
1 Row &
commit

Insert
1 Row &
commit

- Drop-Recovery at timestamp t_7: Does your recovery procedure detect the PIT range between t_3 and t_5?

Table T consists of how many rows at time t_7 ?

Let us start at timepoint t_1: an image copy has been taken. Afterwards (timepoints t_2 and $t_{4)}$), a single row has been inserted at each of these timepoints. At t_5, a point-in-time recovery to timepoint t_3 was performed, which means that

- the most recent image copy before t_3 will be loaded, and the log records until timepoint t_3 will be applied.

- a row will be inserted in SYSCOPY in order to persist the information that the log interval between t_3 and t_5 is invalid. This interval is called a ,,PIT Range' , which means an interval that will be skipped by any further recovery operation..

After performing this recovery operation, the number of rows of table T is 1 (the one row inserted at timepoint t_2).

At timepoint t_6 we insert another row into table T. Any recovery (to current timepoint) performed at timepoint t_7 will end up with a table consisting of 2 rows: Recovery restores the image copy, applies the logrecords until t_3, and between t_5 and t_7 . The log interval between t_3 and t_5 (PIT Range) will be skipped.

Dropped Objects Recovery step by step

- Recreate Object (rerun DDL), Compare DBID, PSID, OBID
- DSN1COPY: Restore tablespace from image copy
- Recover LOGONLY LOGRANGES NO
- Rebuild Indexes and **Copy Tablespace**

This procedure describes the steps necessary to perform a recover of a dropped tablespace when PIT ranges can be excluded.

A prerequisite is the ability to recreate the dropped tablespace by generating the same PSID.

PSID and OBIDs of the dropped tablespace and tables might be found in the header page of the dropped tablespace's last image copy (if the tablespace consists of many tables, the table OBIDs are to be found in the corresponding data pages). While recreating tables, the OBID can be defined directly as part of the SQL Create Table DDL statement.

If even after several attempts the recreation of the same PSID for the tablespace is not successful, this procedure can not be applied and a „Plan B" must be applied: PIT recovery of the subsystem, application of ISV products, etc..

However, it is not a problem to include SHRLEVEL CHANGE image copies in this process, as the RECOVER LOGONLY operation will take this into account and applies the log records from the begin timepoint of the image copy.

Also, FULL and INCR image copies are both supported by DSN1COPY.

Dropped Objects Recovery *with PIT range* step by step

- Recreate Object (rerun DDL), Compare DBID, PSID, OBID
- DSN1COPY: Restore tablespace from image copy
- Search for PIT ranges:
  - Older SYSCOPY Image Copy (taken before drop tablespace), or
  - Output/Documentation of earlier Point-in-time Recovery
- *Skip PIT Ranges (iteratively, if more than one):*
  - *Recover LOGONLY TOLOGPOINT LOGRANGES NO*
  - *Repair HPGRBRBA*
- Recover LOGONLY LOGRANGES NO
- Rebuild Indexes und Copy Tablespace

```
REPAIR LOCATE TABLESPACE DB900660.SRESERV PART 1 PAGE 0
              VERIFY  OFFSET X'62' DATA X'BC44056BAF9C'
              REPLACE OFFSET X'62' DATA X'BC43D8476354'
```

If the log can or should not be applied without any log records to be skipped (for example because of a PIT range or another reason that forces a skip during log apply), then the log apply phase has to happen the following way:

1.RECOVER TOLOGPOINT X'timepoint t' LOGONLY LOGRANGES NO, where *t* is the last log record to be applied before the log skip

2.REPAIR HPGRBRBA: Replace the restart-point of LOGONLY recovery by any arbitrary logpoint selected by yourself.

3.Repeat step 1 until the next interval to be skipped.

Look for DBID, PSID, OBID in dropped tablespace's last image copy (DSN1PRNT FORMAT,PRINT(0*)))

```
PGCOMB='10'X  PGLOGRBA='C9880FAA82FD'X  PGNUM='00000000'X  PGFLAGS='18'X
HPGOBID='01130062'X  HPGHPREF='00005777'X  HPGCATRL='00'X  HPGREL='D4'X
HPGZLD=
HPGCATV='00'X  HPGTORBA='C98412A7B4D8'X  HPGTSTMP='20120419153011117384'X
HPGSSNM='DB2S'  HPGFOID='0061'X  HPGPGSZ='1000'X  HPGSGSZ='0000'X
HPGPARTN='000
HPGZ3PNO='000000'X  HPGZNUMP='00'X  HPGTBLC='0001'X  HPGROID='0063'X
HPGZ4PNO='00000000'X  HPGMAXL='0100'X  HPGNUMCO='0005'X  HPGFLAGS='0008'X
HPGCONTM='20120419153053047766'X  HPGSGNAM='SYSDEFLT'  HPGVCATN='DB2S     '
HPGRBRBA='C984223D7565'X  HPGLEVEL='C984223D7565'X  HPGPLEVL='C9841B240546'X
HPGCLRSN='C9880FAAFE01'X  HPGSCCSI='01F4'X  HPGDCCSI='0000'X
HPGMCCSI='0000'X
HPGFLAG2='00'X  HPGEPOCH='0000'X  HPGRBLP='000000000000'X  HPGDNUMB='01'X
HPGDNUMC='0100'X  HPGDFSG='00000000'X  HPGDLSG='00000000'X
HPGSISP='00000000'X
FOEND='N'
```

0113 DBID hex value

0062 PSID hex value

0063 table's OID hex value (for tablespace with 1 table only)

*) For partitioned tablespaces, set to first page of pageset

Question

A dropped tablespace

a) *can only be recovered if tablespace recreation ends up with an identical PSID.*
b) *can only be recovered by applying ISV products*
c) *can always be recovered, but – as worst case - with an outage of the complete data sharing group.*
d) *can always be recovered by applying the REPAIR utility*

Recovery Control Methodology: A 10-Point-Approach for every recovery scenario.

1. **Save** (Copy, DSN1COPY, VSAM Repro, IDCAMS) the concerned objects
2. Set a defined **time limit** for the end of the recovery process
3. Apply **standard (predefined) procedures** to recover (Recover, Rebuild, Restart, Reorg).

4. Problem **escalation** if time limit or result not reached

5. In Parallel:
   5a) **Analyze** problem reason as   5b) Develop **different**
        far as necessary to be able   **approaches** to attack
        to repair   the problem
6. **Decide** between the different problem solution approaches
7. **Implement** your selected approach
8. **Save** after completion
9. **Information**
10. **In-depth analysis** and lessons learned

Recovery & Restart Resources

Important Header Page Offsets:
X'10'   HPGHPREF
X'18'   HPGTORBA
X'62'   HPGRBRBA

Important Data Page Offset
X'01'   PGLOGRBA
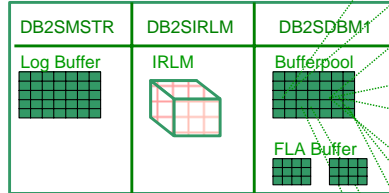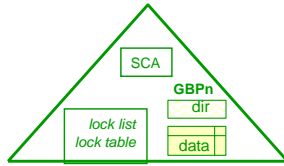
RTS Tables (row for T's Tspce)
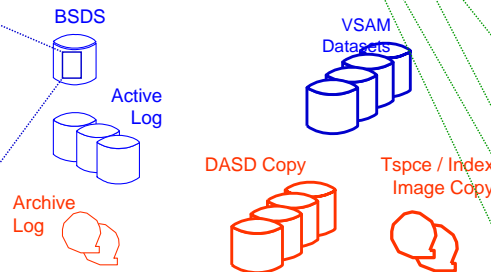
UPDATE STATS TIME	COPY CHANGES	COPY UPDATE LRSN

SYSLGRNX (row for T's Tspce)

DATE	TIME	START LRSN	STOP LRSN

SYSCOPY (row for T's Tspce)

ICTYPE	START RBA	PIT RBA

DB2SMSTR	DB2SIRLM	DB2SDBM1
Log Buffer	IRLM	Bufferpool / FLA Buffer

SCA
GBPn
dir
lock list
lock table
data

Header Page (T's tablespace)

HPGRBRBA	
HPGLEVEL	
HPGPLEVEL	
HPGHPREF	
HPGTORBA	

Data Page (T's tablespace)

PGLOGRBA	

Highest RBA Information
Active Log Directory
Archive Log Directory
CRCR History
Checkpoint Queue
-ARCHIVE LOG History

BSDS

Active Log

Archive Log

VSAM Datasets

DASD Copy

Tspce / Index Image Copy

Thomas Baumann

Swiss Mobiliar

thomas.baumann@mobi.ch

Session F10
Prepare to Repair

INTERNATIONAL
DB2 USERS GROUP
Independent • Not-for-Profit • User Run
Founded 1988

Since 1992, Thomas Baumann has been focusing on understanding how the DB2 engine works. He has a master degree of computer sciences from ETH Zurich, Switzerland, and is currently working both as head of DB2 and as IT performance architect at Swiss Mobiliar Insurance in Berne, Switzerland. If he is not in his office trying how to get the most out of DB2, he is somewhere lecturing on database optimization. Thomas is a certified information systems auditor (CISA), holds a CRISC (certified in risk and information systems controls) certificate, and is a member of the IDUG speaker hall of fame.