

# Assignment 4 - DB2

Daniel Brand

Andreas Kostecka

## 1. RECOVERY-METHODS

Whenever an application error, power interruption or storage failure happens the database has to be rebuilt. For this case it is very important to backup the whole database or individual table spaces to rebuild damaged, broken or corrupted data.

There are three types of recovery:

- Crash-Recovery
  - When transactions (also called UOW, units of work) are interrupted unexpectedly the database may have an inconsistent or unusable state. Crash-Recovery deals with such problems.
- Version-Recovery
  - Version-Recovery describes the restoration of a previous version of the database. For this recovery type DB2 needs an image that was created during a backup operation.
- Rollforward-Recovery
  - Rollforward-Recovery can be used to reapply changes that were made by transactions that were committed after a backup was made.

After a power interruption the DB2 Database-Manager starts automatically the Crash-Recovery routine.<sup>1</sup>

## 2. BACKUP AND RECOVERY-STRATEGY

For an backup and recovery strategy we should ask ourselves some questions:

Will the database be recoverable? How much time can be spent recovering the database? How much time will pass between backup operations? How much storage space can be allocated for backup copies and archived logs? Will table space level backups be sufficient, or will full database backups be necessary? and so on.<sup>2</sup>

A strategy for recovering a database should ensure that all the informations needed for the recovery process are available to the system. A strategy should also contain a timetable for periodically backups.

<sup>1</sup>[https://www-01.ibm.com/support/knowledgecenter/?lang=de#!/SSEPGG\\_9.7.0/com.ibm.db2.luw.admin.ha.doc/doc/c0052073.html](https://www-01.ibm.com/support/knowledgecenter/?lang=de#!/SSEPGG_9.7.0/com.ibm.db2.luw.admin.ha.doc/doc/c0052073.html)

<sup>2</sup>[https://www-01.ibm.com/support/knowledgecenter/?lang=de#!/SSEPGG\\_9.7.0/com.ibm.db2.luw.admin.ha.doc/doc/c0005945.html](https://www-01.ibm.com/support/knowledgecenter/?lang=de#!/SSEPGG_9.7.0/com.ibm.db2.luw.admin.ha.doc/doc/c0005945.html)

## 2.1 Recovery-Methods in Detail

- Database-Backup:<sup>3</sup>  
Copying and saving the copy of the data on a different drive. If the original data gets corrupted we can easily restore the database. A simple way to generate a backup is to stop the database to ensure that there are no more active transactions and then creating a backup-image from the original data.
- Database-Recovery:  
The recreation of the database is called recovery. Version recovery is the restoration of a previous version of the database. Rollforward recovery is the re-running of all protocolled transactions after the recovery of a database or table space.
- Recovery after system crash:  
A recovery after a system crash is a so called automatic recovery of the database. Transactions against a database can be interrupted unexpectedly when a system crash happens. When this crash occurs before all of the changes made by the transactions are finished and committed, the database has an inconsistent and unusable state. The automatic crash recovery routine established first a connection to the database. After that the database logs are used to restore the database to a consistent state. Database changes performed by transactions that are incomplete gets rolled back and also completing committed transactions that are still in memory when the crash occurred (Figure 1). Also data page externalization is performed for all committed transactions. At the end of the crash recovery the DB2 database is restored into a transaction consistent state. If necessary damaged table spaces are also fixed.

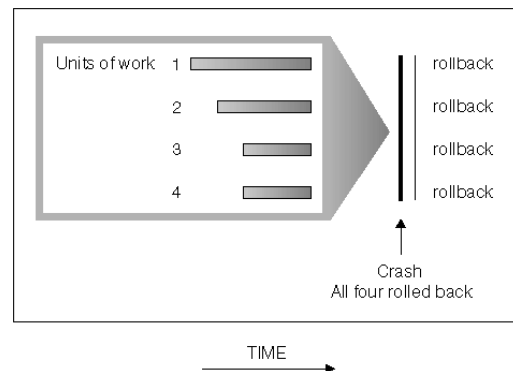


Figure 1: Crash-Recovery

<sup>3</sup>[http://myy.haaga-helia.fi/~dbms/db2/01\\_Classroom/2.1/%20-%20DB2%20backup%20and%20recovery\\_Lab.pdf](http://myy.haaga-helia.fi/~dbms/db2/01_Classroom/2.1/%20-%20DB2%20backup%20and%20recovery_Lab.pdf)

- Log-Files (seen in Figure 2):

- Recovery log files: The DB2 **Recovery log files** tracks all of the changes made against the database. It is also called the transaction log. A transaction is invoked whenever one of the following operations is executed: insert, read, update, delete data. First the record is copied to the buffer pool, where it is modified by the chosen operation. Once a record is modified due to one of these operation a new record reflecting the operation is written to the log buffer. The log buffer is a storage area in the memory. On a delete operation the record containing the original values is written to the log buffer.
- Recovery history files: The DB2 **Recovery history file** is created for each database and updated during various operations. These operations could be: table spaces are backed up, restored or rolled forward; table space is created, altered, renamed, dropped; table is loaded, dropped or reorganized; database is recovered; log file is created or archived.
- Table space history files: The **Table space history files** keeps track of which logs need to be included and processed for each table space. For example DB2 uses the table space history and log files for recovering a dropped table (is done via table space level restore). When we create a backup also the history files are backed up. Each entry in the history file has an associated status: active, inactive, expired, pending, delete, deleted, and do\_not\_delete.

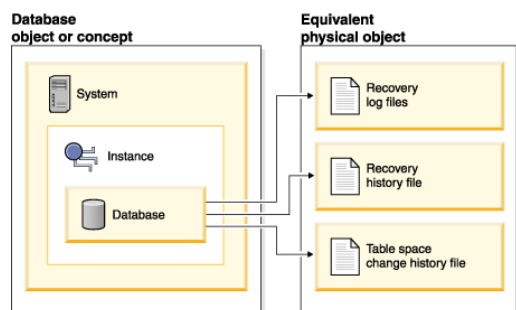


Figure 2: Database-Logs

## 2.2 Automated Backup Operations

It is very time consuming to determine whenever a backup-operation should be executed. DB2 offers a Configure Automatic Maintenance wizard for doing automatic backups. With automatic maintenance, you specify your maintenance objectives, including when automatic maintenance can run. It is still possible to perform a manual backup.

## 2.3 Monitoring

To monitor the progress of backup, restore and recovery operations DB2 offers the following command<sup>4</sup>:

<sup>4</sup>[https://www-01.ibm.com/support/knowledgecenter/?lang=en#!/SSEPGG\\_9.7.0/com.ibm.db2.luw.admin.ha.doc/doc/t0011701.html](https://www-01.ibm.com/support/knowledgecenter/?lang=en#!/SSEPGG_9.7.0/com.ibm.db2.luw.admin.ha.doc/doc/t0011701.html)

## LIST UTILITIES SHOW DETAIL

This command displays the progress information of all active utilities on the corresponding instance. The information contains the type of the activity (backup, recover, restore, rollforward), database name, description for the type (eg. offline backup), start time, state (eg. executing, finished ...), who invokes the operation (eg. user, system ...), priority. Also some information about the progress monitoring: percentage complete, total work, already completed work.

## 3. BACKUP

For a simple backup operation we only need the alias of the database and the following command:

```
db2 backup db alias_name_db
```

If not specified the backup is stored in the directory where the command is executed. It is possible to save backups in:

- A directory (for backups to disk or diskette)
- A device (for backups to tape)
- A Tivoli Storage Manager (TSM) server
- Another vendor's server

The recovery history file is updated automatically with summary information whenever you invoke a database backup operation. This file is created in the same directory as the database configuration file.

With the tool **db2ckbckp** we can display information about existing backup images. This tool can test the integrity of backups, display the backup-header and display information about the objects and the log file header in the backup image.

## 4. RECOVER VS RESTORE AND ROLL-FORWARD

With **RESTORE**, you specify the backup image you want to restore (indicated by the timestamp the backup was TAKEN AT) the media you are restoring from and the database name you are restoring to. Then if you want to roll the database forward to end of logs or a point in time you follow the restore with a **ROLLFORWARD** command.

The **RECOVER** command combines the two statements above and makes it also simpler to specify a certain backup-goal.

For example if you simply want to get the database back to the end of the log files that exist you simply run:

```
db2 RECOVER DATABASE db_name TO END OF LOGS
```

DB2 searches automatically for the last corresponding backup and invokes the **RESTORE** command with all the needed parameter. After the **RESTORE** is executed the **RECOVER** command invokes **ROLLFORWARD** to the end of logs.