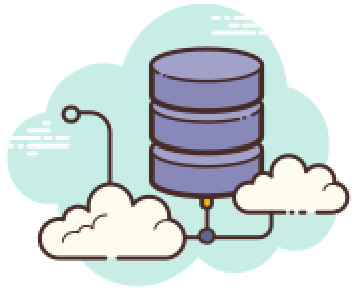
INF115 Lecture 13: *Database Administration*

Adriaan Ludl
Department of Informatics
University of Bergen

Spring Semester 2021

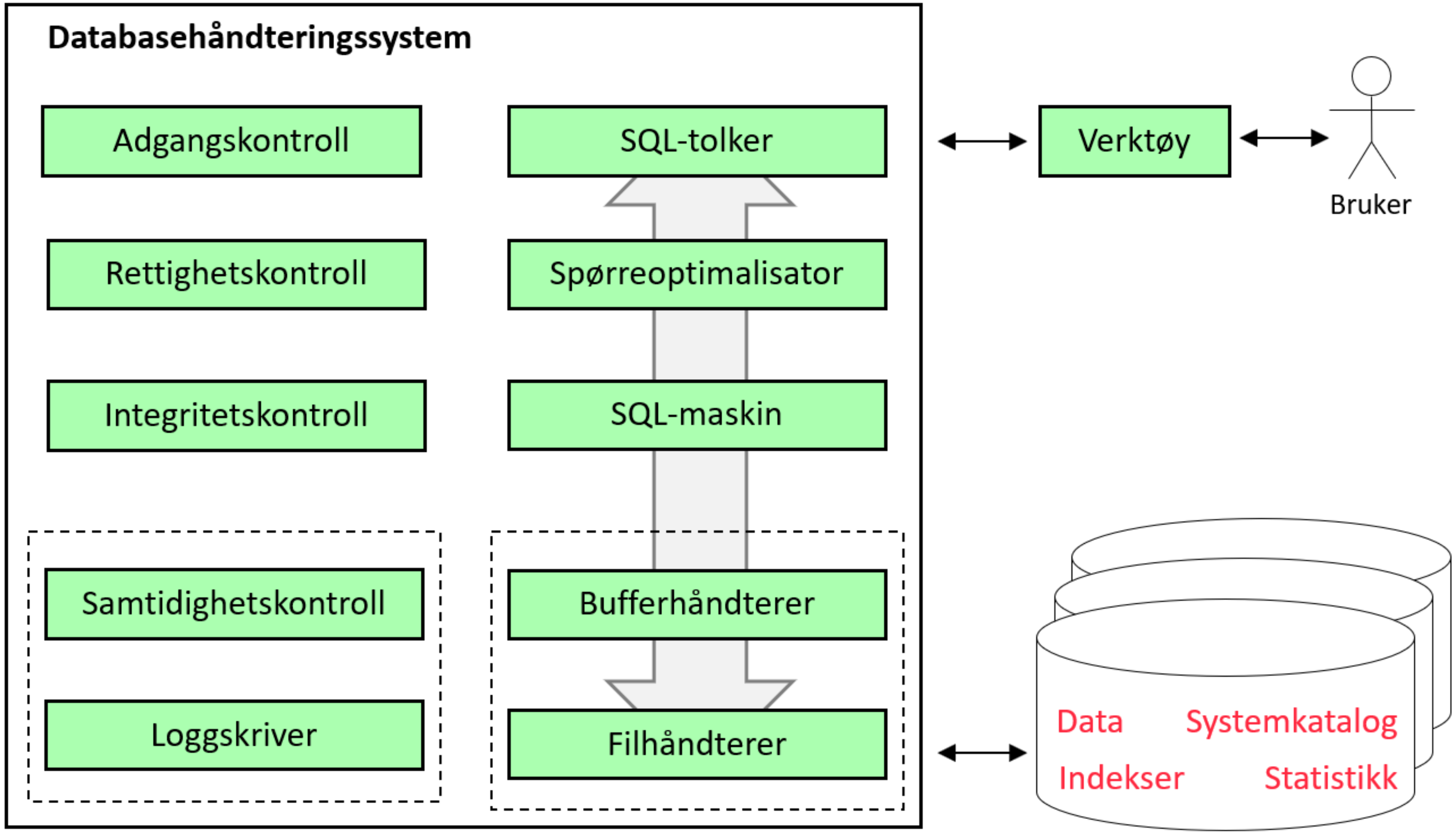
Chapter 11: *Database Administration*



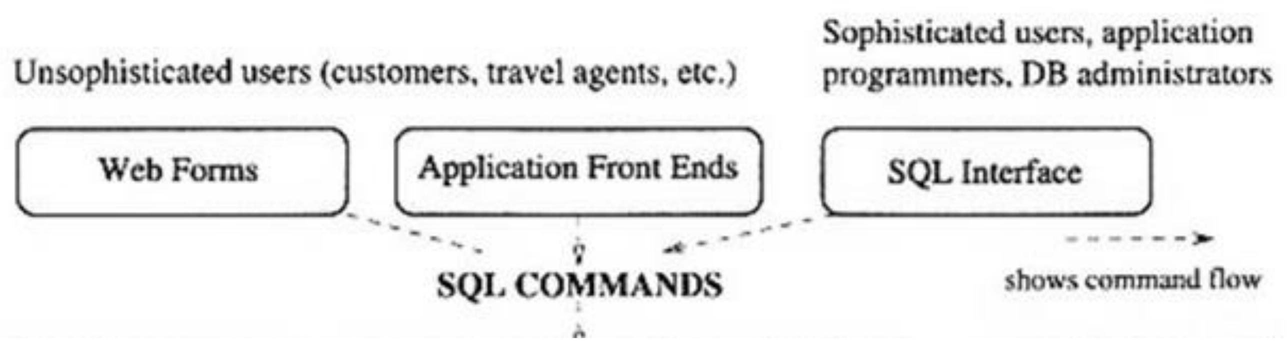
Learning Goals:

- Know the basic **structure** of a database management system (DBMS) and **which tasks it performs**.
- Perform simple **database administration tasks**:
  - **user** administration, **backup** and **recovery**.
- Important concepts in **system architectures** for database solutions
  - **client / server**, **parallel**, **distributed** and **memory** databases.
- **Cloud** databases.
- **Query optimization**.

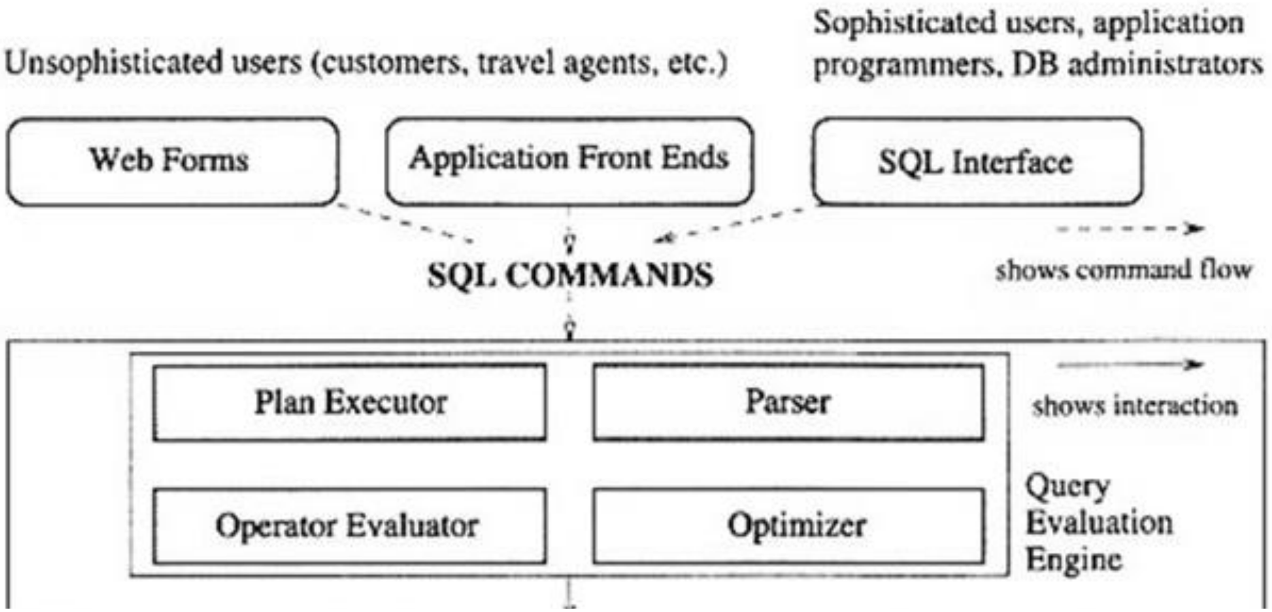
Simplified structure of a DBMS



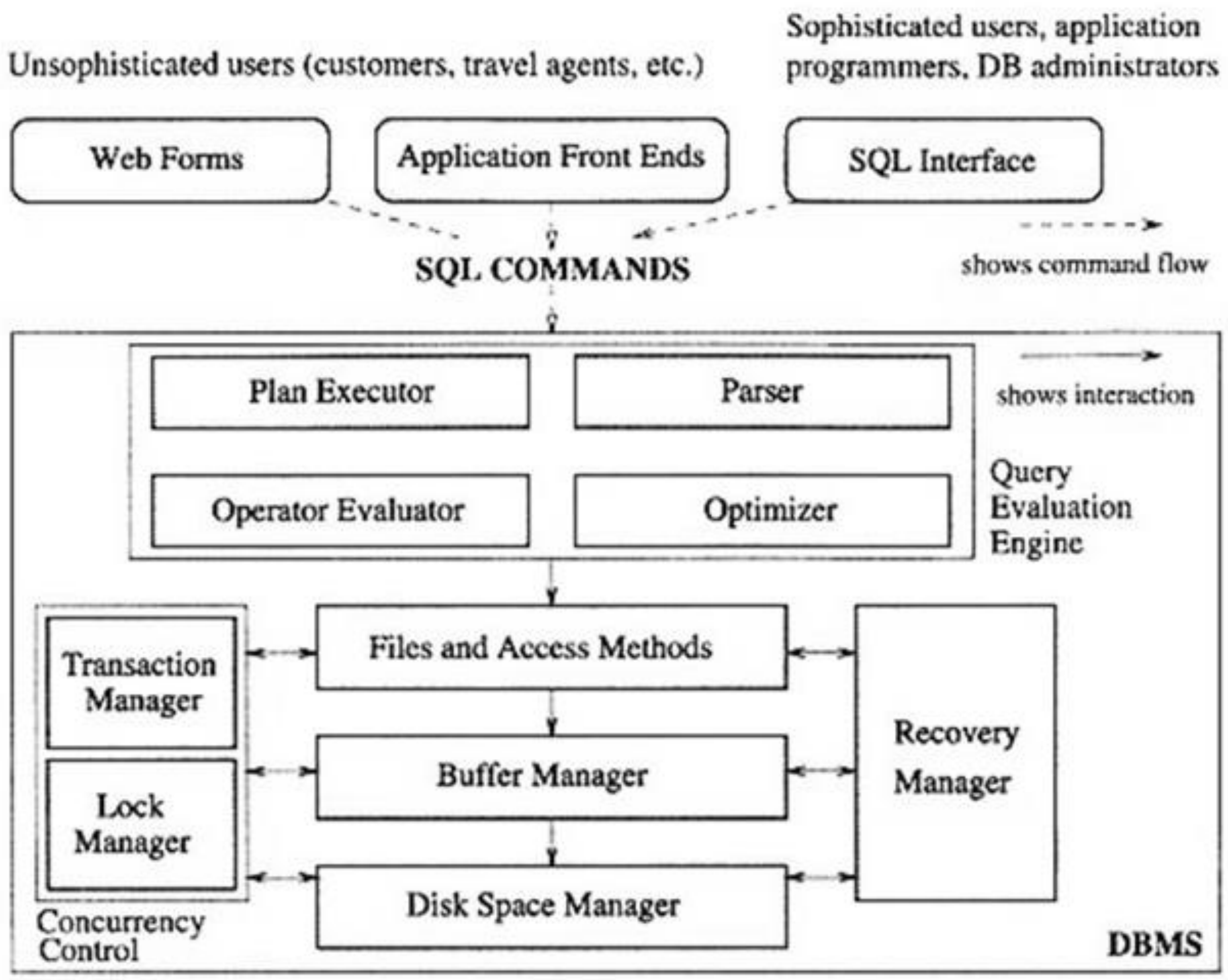
Architecture of a DBMS


Unsophisticated users (customers, travel agents, etc.)
Sophisticated users, application programmers, DB administrators
Web Forms
Application Front Ends
SQL Interface
SQL COMMANDS
shows command flow

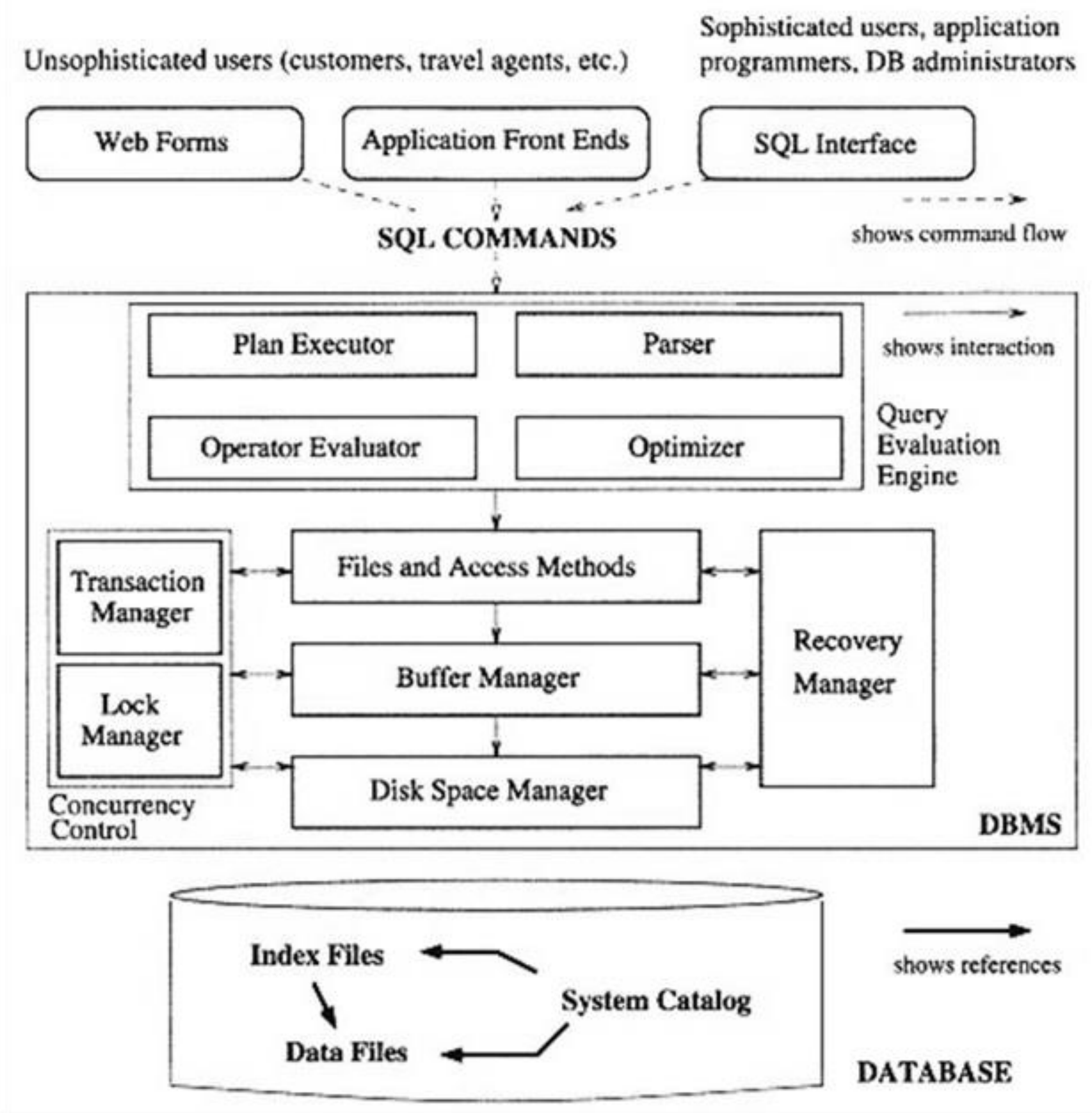
Architecture of a DBMS



Architecture of a DBMS



Architecture of a DBMS



The Tasks of the DBA

A distinction is often made between *data administration* (strategic) and **database administration** (technical).

- Define **guidelines** and **procedures** **:** integrity and validation rules
- ***Evaluation*****,** ***selection*** and ***installation*** of **DBMS** and other tools
- **User** **administration**
- ***Monitoring****: data integrity,* storage requirements, user activity …
- **Training** and **support** for **users**
- ***Backup*** and ***recovery*** after **errors**
- **Capacity** **planning**, **optimization** and **tuning**

DBA Tools

Many of the DBA tasks can be done with SQL.

There are also dedicated «DBA tools». Typical functionality:

- Start and download the database
- Backup and recovery after failure
- Define tables, indices,…
- User administration
- Set limits on users’ resource use
- Organize physical storage structures
- Write and analyze SQL commands
- Start batch jobs
- Visualize the use of disk, buffer
- Online: Make the database available on a network

Quizz on *Database Administration* (part 1)

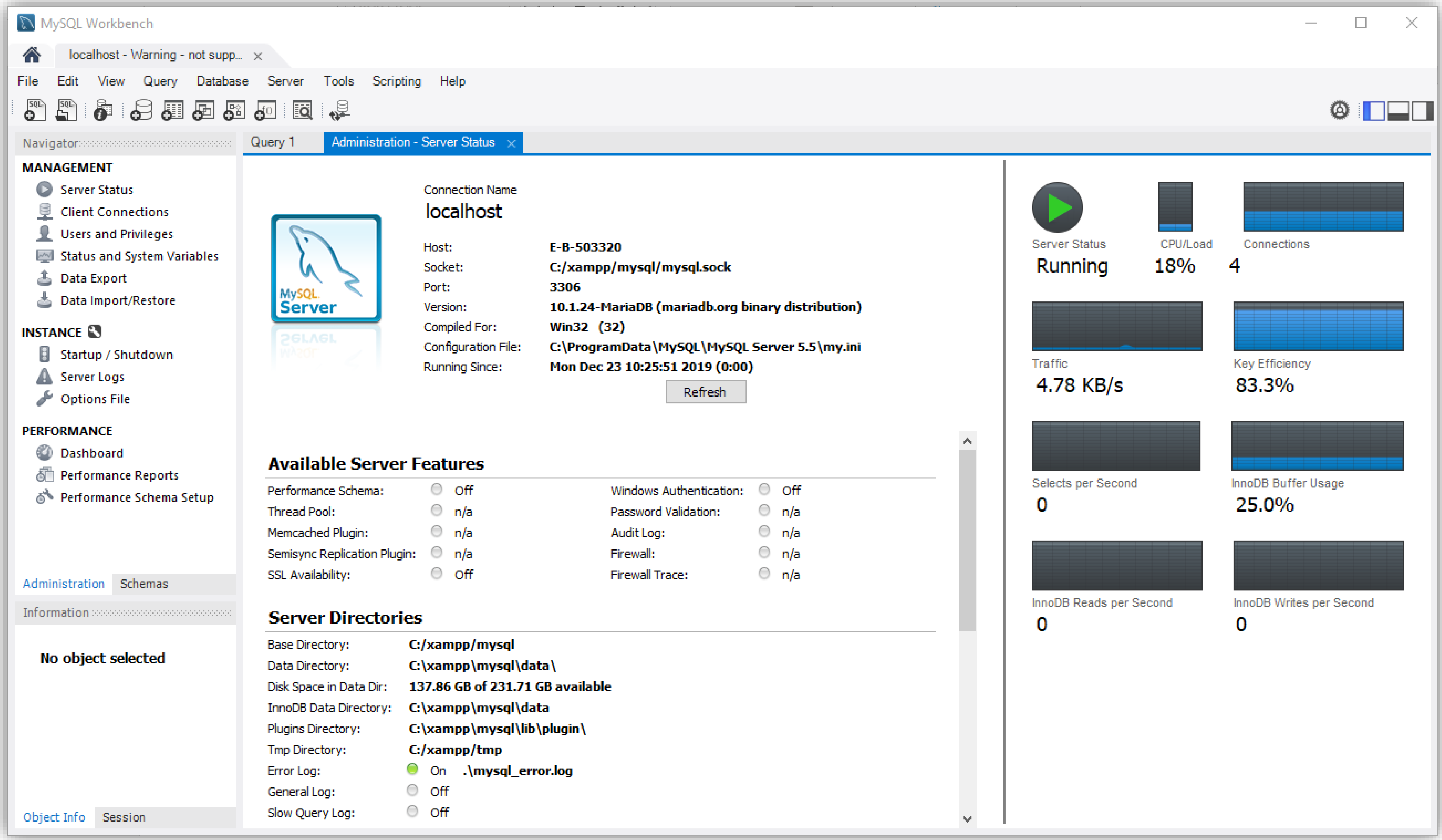
Please answer the practice quizz on mitt.uib now ☺

(you can take it again later if you want)

Link:

➢ https://mitt.uib.no/courses/27455/quizzes

MySQL: Server Administration



Connection Name
localhost

Host: E-B-503320
Socket: C:/xampp/mysql/mysql.sock
Port: 3306
Version: 10.1.24-MariaDB (mariadb.org binary distribution)
Compiled For: Win32 (32)
Configuration File: C:\ProgramData\MySQL\MySQL Server 5.5\my.ini
Running Since: Mon Dec 23 10:25:51 2019 (0:00)

Server Status: Running
CPU/Load: 18%
Connections: 4
Traffic: 4.78 KB/s
Key Efficiency: 83.3%
Selects per Second: 0
InnoDB Buffer Usage: 25.0%
InnoDB Reads per Second: 0
InnoDB Writes per Second: 0

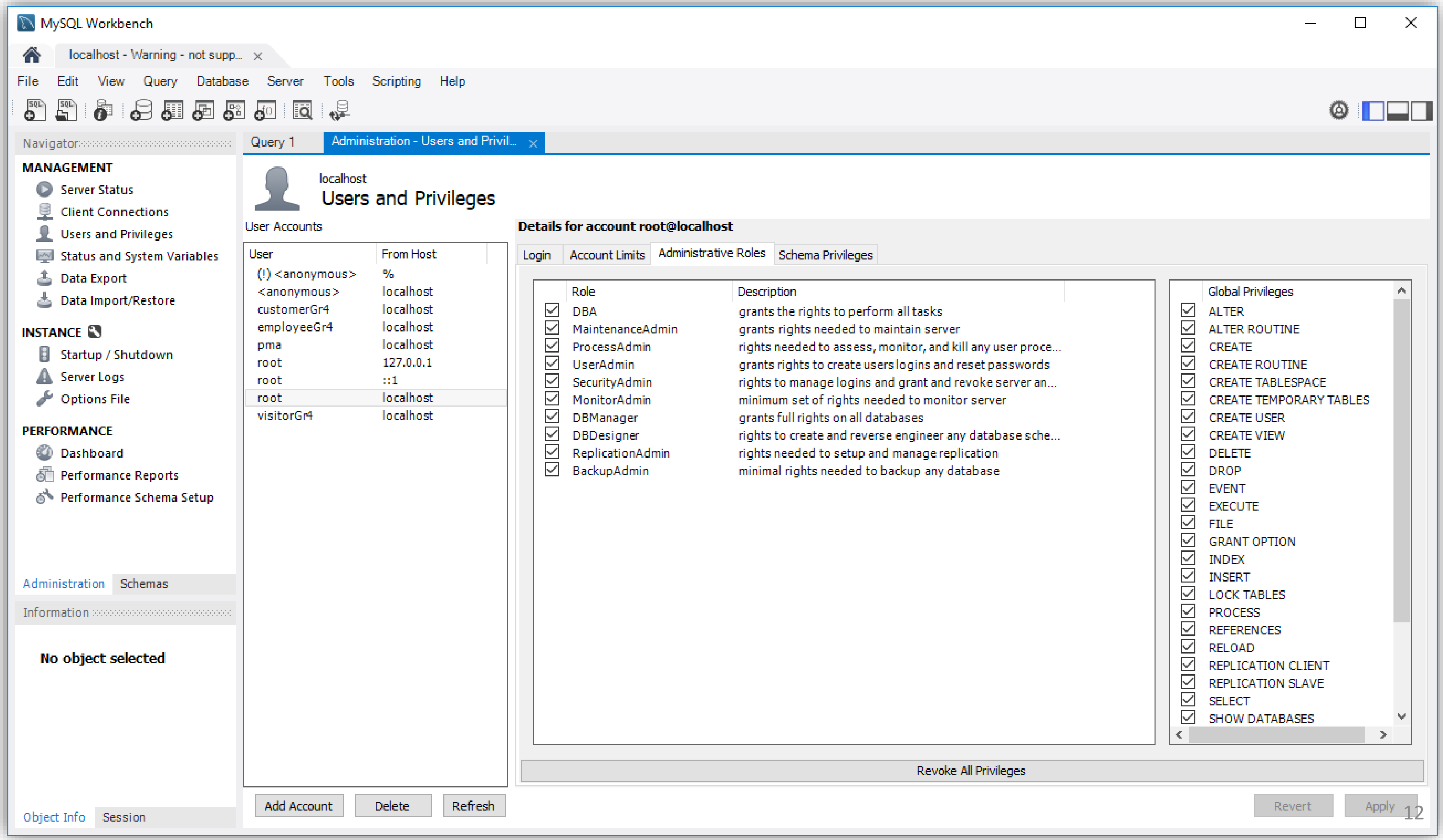
Available Server Features

Performance Schema: ○ Off
Thread Pool: ○ n/a
Memcached Plugin: ○ n/a
Semisync Replication Plugin: ○ n/a
SSL Availability: ○ Off
Windows Authentication: ○ Off
Password Validation: ○ n/a
Audit Log: ○ n/a
Firewall: ○ n/a
Firewall Trace: ○ n/a

Server Directories

Base Directory: C:/xampp/mysql
Data Directory: C:\xampp\mysql\data\
Disk Space in Data Dir: 137.86 GB of 231.71 GB available
InnoDB Data Directory: C:\xampp\mysql\data
Plugins Directory: C:\xampp\mysql\lib\plugin\
Tmp Directory: C:/xampp/tmp
Error Log: ● On .\mysql_error.log
General Log: ○ Off
Slow Query Log: ○ Off

MySQL: Users and Rights



No object selected

localhost

Users and Privileges

User Accounts

User	From Host
(!) <anonymous>	%
<anonymous>	localhost
customerGr4	localhost
employeeGr4	localhost
pma	localhost
root	127.0.0.1
root	::1
root	localhost
visitorGr4	localhost

Details for account root@localhost

	Role	Description
☑	DBA	grants the rights to perform all tasks
☑	MaintenanceAdmin	grants rights needed to maintain server
☑	ProcessAdmin	rights needed to assess, monitor, and kill any user proce...
☑	UserAdmin	grants rights to create users logins and reset passwords
☑	SecurityAdmin	rights to manage logins and grant and revoke server an...
☑	MonitorAdmin	minimum set of rights needed to monitor server
☑	DBManager	grants full rights on all databases
☑	DBDesigner	rights to create and reverse engineer any database sche...
☑	ReplicationAdmin	rights needed to setup and manage replication
☑	BackupAdmin	minimal rights needed to backup any database

	Global Privileges
☑	ALTER
☑	ALTER ROUTINE
☑	CREATE
☑	CREATE ROUTINE
☑	CREATE TABLESPACE
☑	CREATE TEMPORARY TABLES
☑	CREATE USER
☑	CREATE VIEW
☑	DELETE
☑	DROP
☑	EVENT
☑	EXECUTE
☑	FILE
☑	GRANT OPTION
☑	INDEX
☑	INSERT
☑	LOCK TABLES
☑	PROCESS
☑	REFERENCES
☑	RELOAD
☑	REPLICATION CLIENT
☑	REPLICATION SLAVE
☑	SELECT
☑	SHOW DATABASES

Security threats and measures

Accidents:

- Human error
- Software error
- Hardware failure
- Power outage

Attacks:

- Employees / external
- Exploits of vulnerabilities
- Directly to the database
- Indirectly via operating system / network

Security threats and measures

Accidents:

- Human error
- Software error
- Hardware failure
- Power outage

Attacks:

- Employees / external
- Exploits of vulnerabilities
- Directly to the database
- Indirectly via operating system / network

Security threats and measures

Accidents:

- Human error
- Software error
- Hardware failure
- Power outage

Attacks:

- Employees / external
- Exploits of vulnerabilities
- Directly to the database
- Indirectly via operating system / network

Mitigation/Protective Measures:

- Backup, logging and rebuilding
- Monitoring
- Encryption

- Access control:
  - Physical control
  - Password
  - Rights

- Duplication of equipment
- Emergency unit
- Routines, « fire drills »

Backup and restore

Types of **backup**

- Full / incremental
- Hot / cold
- Logical / physical
- Backup to the cloud (norsk: skyen)

Dedicated tools for backup and **recovery**

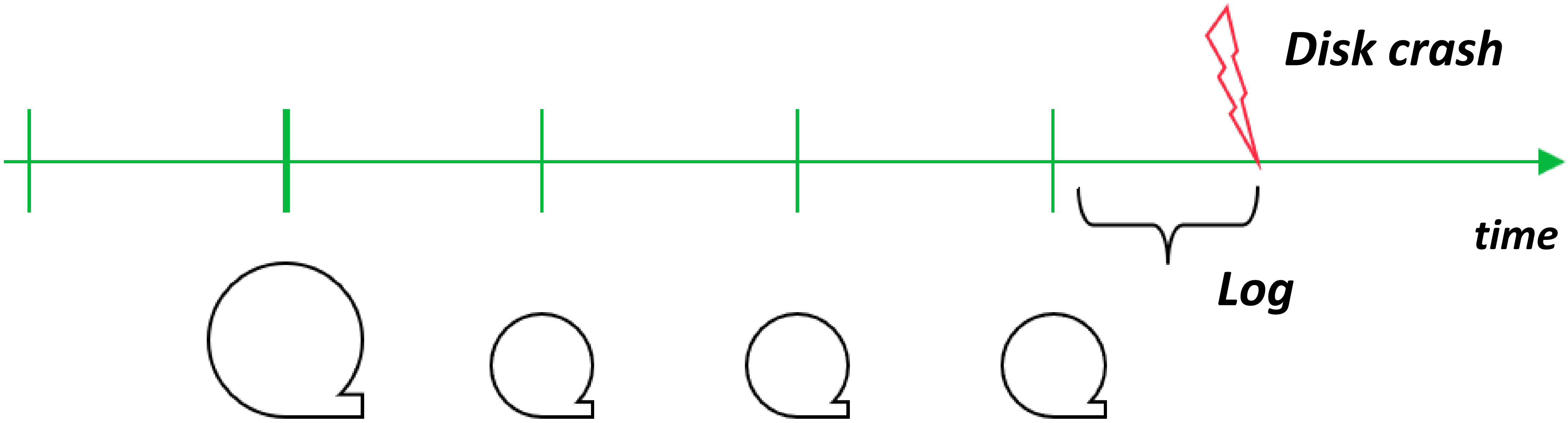
Transactions

- Updates are written first to the **transaction log** then to the database.
- Last Backup + Transaction Log is used for **rebuilding**
  to bring the database back to a **consistent** state.

Backup routines

- Times
- Backup storage

Backups and Rebuilds



Disk crash

time

Log

Full backup

<u>*Incremental*</u> backup

DB Efficiency and Tuning

Choice of hardware

- Number of disks
- Size of RAM
- Network speed

Physical design

- Select indices
- Denormalization
- File structures

Monitoring and optimization

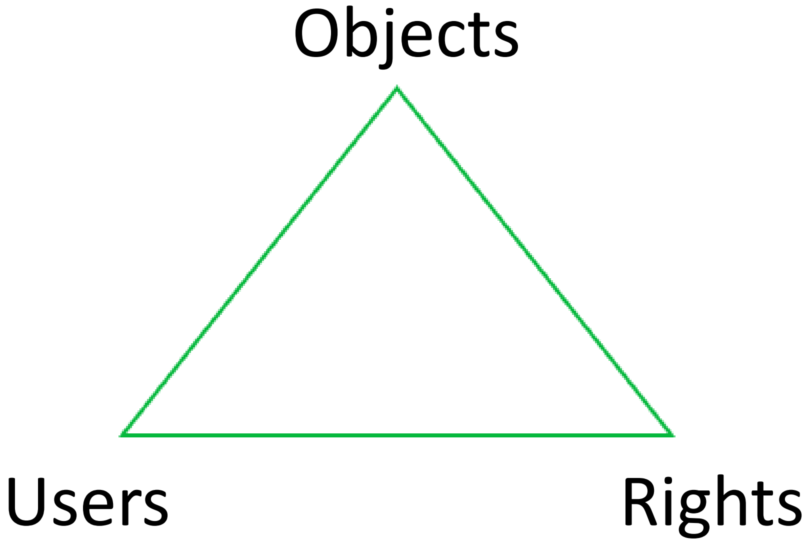
- Write SQL code, give instructions to the DBMS
- Add indexes, change physical storage structure
- Change system parameters

15 minute break!
Lecture resumes at 15:10

User administration

We want to assign a specific **user** the right to perform a specific **operation** on a specific **object**.

- Example: Peter has to be able to read the customer table.



Objects

Users

Rights

We distinguish between **object rights** (privileges) and **system rights**.

- Starting and stopping the DBMS is an example of a system privilege.

The user information is stored in one or more **system tables**.

Roles / groups

A **group** (norsk: rolle) is a collection of users who should have the same rights in the database system.

- Example: The role of *Seller*

Using groups:

1. Create groups
2. Assign group rights to different objects
3. Create users
4. Add users to groups

Users **inherit** rights from the groups in which they are members.

➢ Thus, the DBA avoids assigning rights to individual users.

Rights in SQL

Allow Peter to read and update the table Customers (Kunder):

```
GRANT SELECT, UPDATE
 ON Kunder
 TO Peter
```

Peter shall no longer be allowed to update the table Customers:

```
REVOKE UPDATE
 ON Kunder
 FROM Peter
```

Allow Peter to **pass on the rights**:

`WITH GRANT OPTION` - **for object rights**

`WITH ADMIN OPTION` - **for system rights**

A coarse security system

We only distinguish between a few user types.

- **Administrators** : Gets **all rights**, including system rights such as creating other users, starting and stopping the database, creating tables, etc.
- **Change** : End-users who can both **read and change the contents** of the database. Gets rights to read / insert / update and delete data:

  SELECT + INSERT + UPDATE + DELETE.
- **Access** : Gets rights to read the contents of the database: SELECT.
- **Developers** : Working with the **structure** . Get rights to create new tables, queries, slices, and also to change the definition of existing objects. Usually, developers do not work with databases in production.

Here we do not distinguish between specific objects.

A more fine-grained security system

- Different parts of the database are used by different departments / job categories.
- We divide the group of end users. Examples:
  - DepartmentA, DepartmentB,…
  - Secretary, Salesman, Director,…
  - Level1, Level2, Level3,…
- Can be useful to create groups in several "teams".
  - In Oracle, groups can be assigned to other groups. Example: Both the *Seller* groups and the *Secretary* groups are part of the *Access* group.
  - If the *Seller* and *Secretary* are to have many of the same rights, it is labor-saving to assign the rights once to *Access*, and then give *Access* to *Seller* and *Secretary*.
- Can assign rights to **views** that provide access to selected rows / columns in a table.

Example: SQL and metadata in Oracle

For a similar set-up in MySQL, see database **information_schema**

- How are the tables defined?

```
SELECT table_name, column_name, data_type
FROM user_tab_columns
```

- What rights are distributed on Peter's tables ?

```
SELECT *
FROM dba_tab_privs
WHERE owner=‘Peter’
```

- Which rights have been distributed by Peter ?

```
SELECT *
FROM dba_col_privs
WHERE grantee = ‘Peter'
```

Example: SQL and logging in Oracle

Log sample queries to the Employee table:

```
AUDIT SELECT ON Ansatt;
```

- What happened ?

```
SELECT username, timestamp,
       obj_name, action_name
FROM user_audit_object;
```

- *Triggers* can be used to “tailor” monitoring.

Example: Write to a log table each time Peter changes something in the column Ansatt.Lønn.

The system catalogue

Metadata , or "data about data" are descriptions of the database:

  - What tables are there? How are the tables structured?
  - Which indices are defined?

- **Statistical data** is used by the query optimizer:
  - How many rows do the tables have?
  - How many different values are stored in a column?
- Data about **users** and their **rights** .
- The **system catalogue** is a collective term for such data.
- Data in the **system catalogue** is often stored in tabular form (system tables), which means that we can use SQL for **reporting**.

Quizz on *Database Administration* (part 2)

Please answer the practice quizz on mitt.uib now ☺

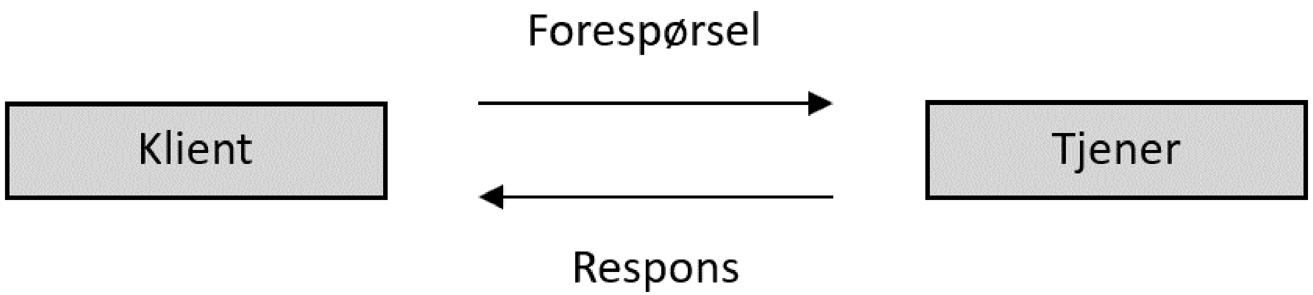
(you can take it again later if you want)

Link:

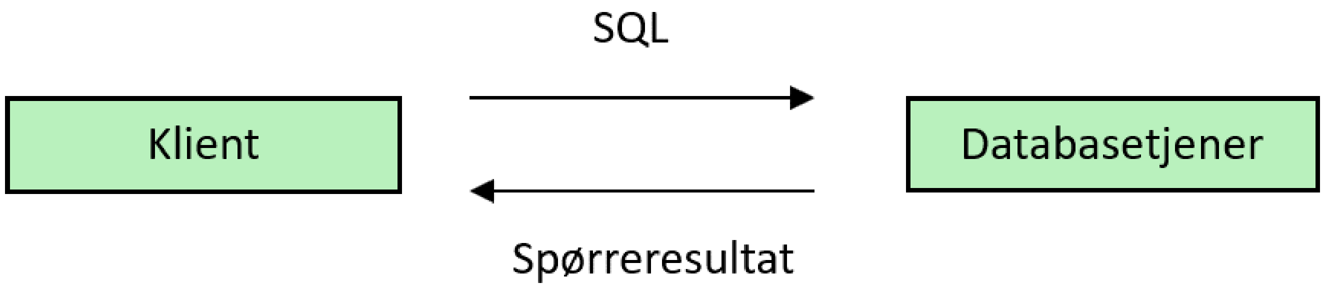
➢ https://mitt.uib.no/courses/27455/quizzes

Client/server architecture

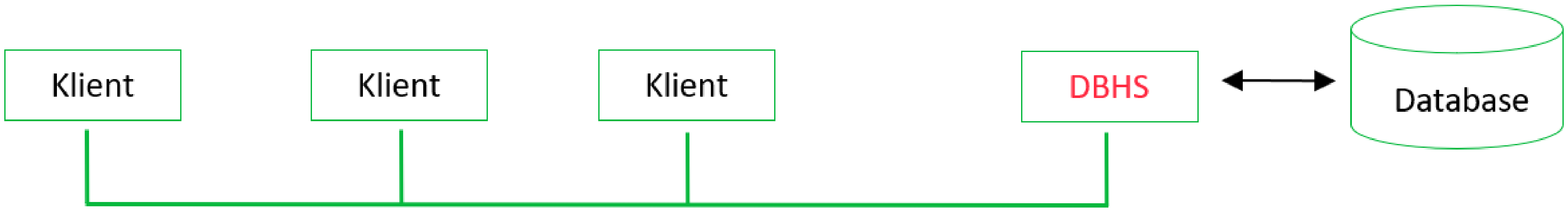
In general:



For Databases:



In Networks:



Cloud solutions

- **IaaS** (*Infrastructure as a Service*) is a ***basic service*** where you **get access to** one or more **machines**, but must install and manage the operating system and software yourself. Example: AmazonWeb Services.
- **PaaS** (*Platform as a Service*) is a more ***comprehensive service*** than IaaS, where in addition to machines and operating system you also get a finished development environment **in the *cloud***, e.g. in the form of a Linux server set up with a MySQL database.
- **SaaS** (*Software as a Service*) is a further **extended service** in relation to PaaS and can be considered as a way to offer software. Example: Google Apps and Office365.

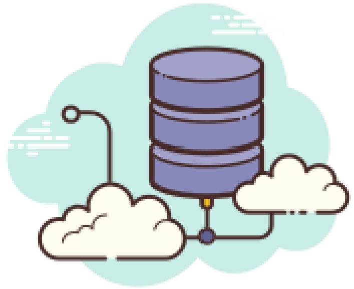
Databases in the *Cloud*

- A **cloud database** (Skydatabase) is a database that is stored in the cloud, and can basically be based on IaaS, PaaS or SaaS.
- **DBaaS** (DataBase as a Service) is a SaaS solution for databases, where you typically create, manage and use the database via a web application.
- **Azure SQL Database** is simply "SQL Server in the cloud" and is available as a DBaaS solution.

Query Optimization

- SQL queries are represented by expressions in **relational algebra** .
- Example of **rewriting rules** :
  - Slide selections and projections "inwards" to the tables to which they apply.
  - Swap equations to reduce the size of intermediate results.
  - Break down complex selections into several simple ones.
- **Cost-based optimization**
  - Uses rewrite rules to generate a number of plans - assigns each plan a cost based on statistics about e.g. the size of tables - and choose the best.

Summary: *Database Administration*



- Know the basic **structure** of a database management system (DBMS) and **which tasks it performs**.
- Perform simple database **administration**:
  - **user** administration, **backup** and **recovery**.
- Important concepts in **system architectures** for database solutions
  - **client / server**, **parallel**, **distributed** and **memory** databases.
- **Cloud** databases.
- **Query optimization**.