MOBI-ADM-M Sample Exam

Daniela Nicklas and Golnaz Elmamooz

End of Summer Term 2022

- Exam: Advance Data Management Sample Exam (MOBI-ADM-M)

- Examiner: Prof. Dr. Daniela Nicklas

- Exam time: 60 minutes, 15 minutes for first reading time are granted

- Aids permitted: Authorized and pre-printed booklets with student's handwriting (details are documented in the VC)

Notes:

- The maximum of points is 60.

- The exam has 7 tasks. Only the best 6 tasks are counted.

- Bonus points are only counted if the exam is passed without bonus points.

- The exam is designed so that on average, each point takes roughly one minute to achieve. You can expect to need on average 5 minutes to solve a 5 point task.

- Write our answers in the dedicated areas with the exam. If needed, extra sheets can be provided. Any unreadable parts will be not be assessed.

- If any assumptions have to be made for a certain answer then they must be justified.

- The password for the solution file is `admsecret`

Student's name: ....

(Your complete name)

Reserved for the examiner.

Task:	1	2	3	4	5	6	7	Bonus
Points achievable:	10	10	10	10	10	10	10	10
Points achieved:								

The exam has 7 tasks.
The maximum achievable points are 60.
Bonus points are only counted if the exam is passed without bonus points.

Total sum:	

1. **Relational Model**[1]

   In the paper by E. F. Codd, *A relational model of data for large shared data banks*, Commun. ACM, Bd. 13, Nr. 6, S. 377–387, 1970, the motivation introduces 5 potential structures for data storage.

   Two of them are depicted here:

Structure 1. Parts, Projects, and Commitment Relationship as Peers			Structure 2. Parts and Projects as Peers Commitment Relationship Subordinate to Projects		
File	Segment	Fields	File	Segment	Fields
F	PART	part#	F	PART	part#
		part name			part name
		part description			part description
		quantity-on-hand			quantity-on-hand
		quantity-on-order			quantity-on-order
G	PROJECT	project#	G	PROJECT	project#
		project name			project name
		project description			project description
H	COMMIT	part#		PART	part#
		project#			quantity committed
		quantity committed			

   The author suggest to write sample programs that print the part number, part name, and quantity committed for every part used in the project called "alpha".

   In a Python like pseudo-code, the program for structure 1 could look as follows:

   ```
   F = open('F')
   for part in F.PART:
       if part.project_name=='alpha':
           print(part.#, part.name,
               part.quantity_committed)
   close(F)
   ```

   (a) (7 points) Write the pseudo-code for structure 2 using the same syntax as in the example.

   (b) (3 points) How does the relational model prevent access path dependence?

---

2. **Database Design**

   (a) (4 points) For a webshop, the client has decided to use MongoDB for managing the order history. However, their customers complain that the loading of the order history takes very long, even when nothing has changed. Which system could they use to speed up the response time for users?". Please specify below your choice of DBMS (that you have learned in ADM lecture) for this case and give a short explanation.

   (b) (3 points) Map the following concepts in relational database management system to other NoSQL databases and fill out the table below:

RDBMS	Extensible Record Stores	Graph DB	Document DB
Table			
Row			

   (c) (3 points) Please mention strengths and weaknesses of Document databases.

3. **Consistent Hashing**

Assume the following scenario: We plan to store documents that contain descriptions of costellation in the PP3 format. For this, we set up a distributed data management system that performs dynamic consistent hashing similar to Dynamo. The filename is used as a key; we found a hash function maps the keys to a value range of 0 to 9 as follows:

Document	Key	h(key)	Document	Key	h(key)
Andromeda.pp3	Andromeda	5	Lynx.pp3	Lynx	6
Columba.pp3	Columba	7	Musca.pp3	Musca	7
Equuleus.pp3	Equuleus	9	Octans.pp3	Octans	9
Hercules.pp3	Hercules	2	Sagitta.pp3	Sagitta	3
Hydrus.pp3	Hydrus	2			

We use three nodes to store data in our systems, named Node A (with token 4), Node B (token 5) and Node C (token 8). For fault tolerance, we configure the system to use 2 replicas, with a read quorum of 1 and a write quorum of 2.

(a) (5 points) We now start adding and querying documents from our system. Below you find the timeline of operations. Please mark in the table which nodes are affected by the described events or operations with an X.

		Affected or contacted nodes		
Time	Operation/Event	A	B	C
1	PUT("Andromeda", Andromeda.pp3)			
2	PUT("Lynx", Lynx.pp3)			
3	PUT("Hydrus", Hydrus.pp3)			
4	Node C gets unavailable			
5	GET("Lynx")			
6	PUT("Lynx", Lynx.pp3)			

(b) (5 points) Which documents are stored on Node B and Node C when the timeline ends? Complete the table below with timestamps and document names.

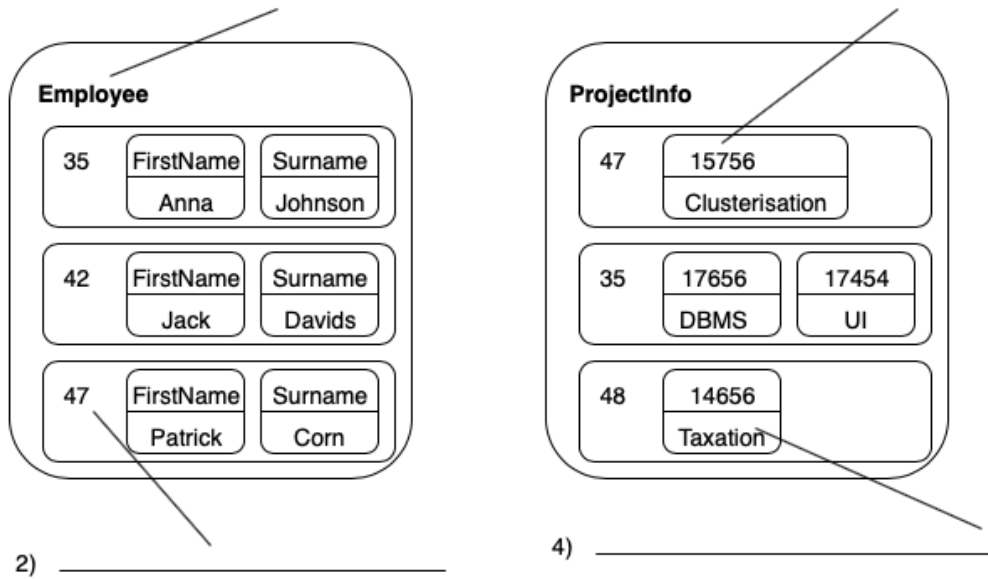| Node_B		Node_C	
t	Object	t	Object

Optional space for assumptions / remarks:

4. **Extensible Record Stores.**

(a) (4 points) In the figure below you see a database structure for an Extensible Record Stores. Please provide terms for the elements we point at.

1) _____   3) _____

Employee

35	FirstName	Surname
	Anna	Johnson

42	FirstName	Surname
	Jack	Davids

47	FirstName	Surname
	Patrick	Corn

ProjectInfo

47	15756
	Clusterisation

35	17656	17454
	DBMS	UI

48	14656
	Taxation

2) _____   4) _____

(b) (1 point) Is it correct that the key "47" exists both in "Employee" and in "ProjectInfo"? Specify in which scope "47" should be unique.

(c) (2 points) Specify the complete key information for "Taxation".(without timestamp)

(d) (3 points) How do Extensible Record Stores storage reduce physical disk space usage?

5. **General Knowledge.**

(a) (3 points) What does "eventually consistent" mean in the context of distributed database?

(b) (3 points) Define the CAP theorem.

(c) (4 points) Briefly sketch a proof for the CAP theorem.

6. **Graph DB (Design / Domain Modeling)**

   Consider the following domain design for a graph database. There are three types of roles in a software company: user, project manager and developer. User defines a project. Project manager manages the project and developer works on a project. Each project has an ID number, description of project and deadline. Project manager and employees have name, ID, name of the section they are working in and salary. For each employee who is working on a project the start date of working on a project is stored.

   (a) (5 points) List the *labels, relations and properties* in your domain.

   Labels:

   Relations:

   Properties:

(b) (5 points) Consider the following facts: Professor Elisabeth is head of the Microeconomics chair. She teaches bachelor course International Business(6 ECTS) and master course Taxation(6 ECTS). Robert is a student. His BA-number is 1987-1232. He is enrolled to both these courses and works for the chair. Data for these facts are inserted in database. How the resulting graph would look like? Use all the relations, properties and labels mentioned above to draw the graph.
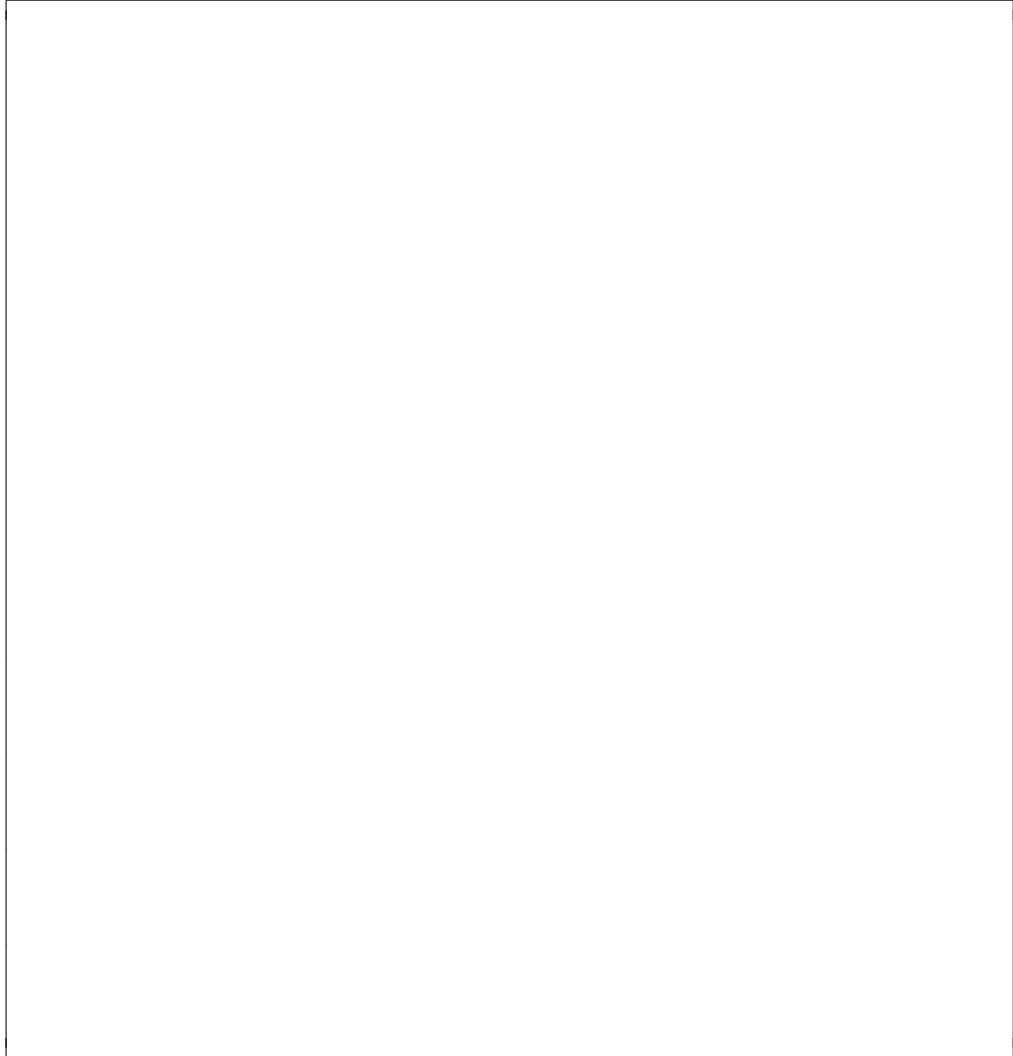
7. **Query Language.**

   Consider that we have designed an appropriate structure in different database management systems for the Email entity with the properties shown in figure below. To execute queries for that data structure we use different query languages.
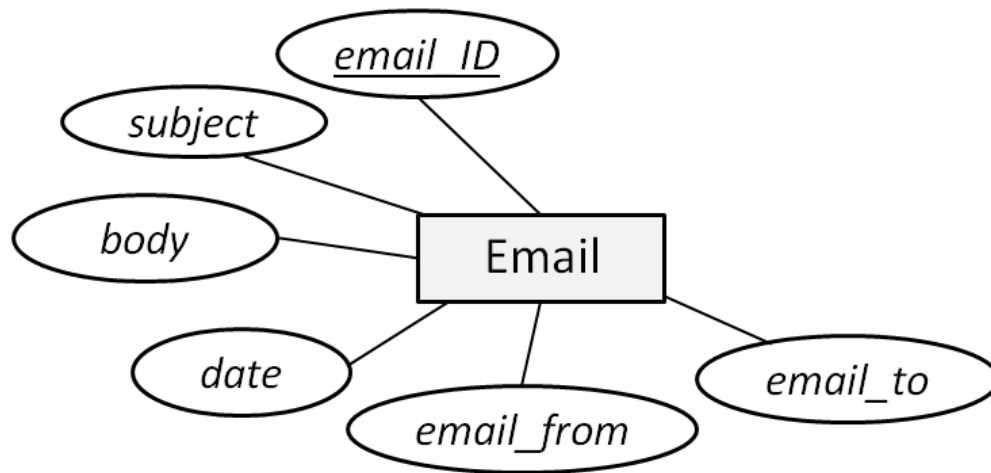
   (a) (6 points) Consider the query below and answer three questions:

   ```
   ALTER table enron.email ADD priority = 1;
   ```

   1. What is the name of the query language?

   2. In which database management system can it be executed?

3. Would this query add a new column with default value? If no, what should be changed to return desired result?

(b) (4 points) If a customer asks you to design a MongoDB database, what should you first ask your customer? Why?