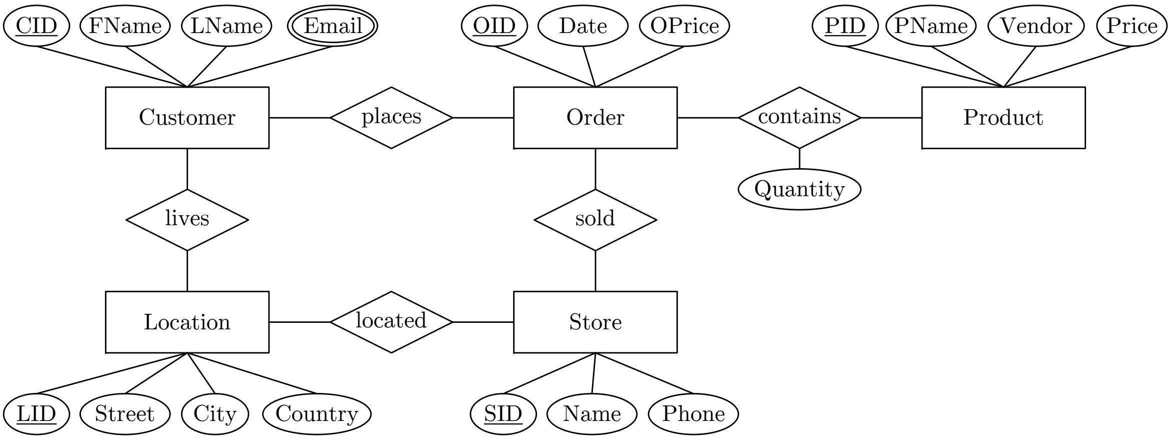
Univ.-Prof. Dr.-Ing. Matthias Boehm
Graz University of Technology
Computer Science and Biomedical Engineering
Institute of Interactive Systems and Data Science
BMK endowed chair for Data Management

February 04, 2022

Exam INF.01017UF Data Management (Winter 2021/22, V1a)

Important notes: The working time is 90min, and lecture materials or any kind of mobile devices are not allowed. Please, make sure to put your *name* and *matriculation number* on the top right of the first page of the task description, and each additional piece of paper. You may give the answers in English or German, written directly into the task description.

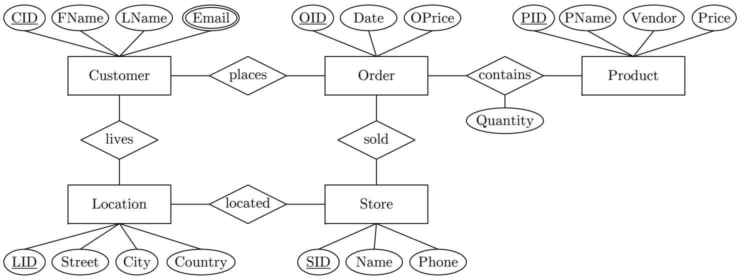
Task 1 Data Modeling (23 points)



(a) Given the Entity-Relationship (ER) diagram above, specify the cardinalities in Modified Chen notation based on the following information. (**10 points**)

- A customer may place between one and many orders, and every order is placed by exactly one customer. Every customer lives in exactly one location, but multiple customers may live in the same location. Customers have one or many emails (multi-valued attribute Email).
- An order contains at least one but potentially many ordered items (i.e., product of a specific quantity). A product may be ordered by many customer orders. The total order price OPrice is not directly derived from the item prices (additional tax and discount, not represented here).
- Every order is sold at a specific (exactly one) store, and a store may have sold zero, one, or many orders. A store has exactly one location, and there cannot be two stores in the same location.

(b) Map the given ER diagram into a relational schema in third normal form, including data types, primary keys, and foreign keys. Besides the given keys, there are functional dependencies City → Country, PName → Vendor, and PName → Price, but both PID and PName are candidate keys. Your schema should also ensure that exactly one relationships are enforced (e.g., each order has an associated customer). (**13 points**)


CID
FName
LName
Email
OID
Date
OPrice
PID
PName
Vendor
Price
Customer
places
Order
contains
Product
Quantity
lives
sold
Location
located
Store
LID
Street
City
Country
SID
Name
Phone

Task 2 Structured Query Language (30 points)

Movies [Mio $]

MID	Title	Year	Revenue
1	The Matrix	1999	455
2	Suicide Squad	2016	746
3	Mr & Mrs Smith	2005	488
4	Passengers	2016	300
5	Catwoman	2004	83
6	The Hunger Games	2012	700
7	Avengers	2012	1520

Ratings

MID	UID	Score
1	1	5
1	6	5
3	4	4
1	5	5
5	4	2
7	3	4
6	4	3
7	6	5

Users

UID	Name	Country
1	Red	DE
2	Orange	AT
3	Yellow	DE
4	Green	AT
5	Blue	DE
6	Violet	DE

(a) Given the Movies, Ratings, and Users tables above, compute the results for the following three queries: (**15 points**)

```
Q1: SELECT M.Title, U.Name, R.Score
      FROM Movies M, Ratings R, Users U
      WHERE M.MID = R.MID
        AND R.UID = U.UID
        AND U.Name = 'Green'
      ORDER BY R.Score ASC

Q2: SELECT Title, Year
      FROM Movies WHERE Year > 2010
    INTERSECT
    SELECT Title, Year
      FROM Movies WHERE Revenue > 500


Q3: SELECT Country, avg(R.Score), max(R.Score)
      FROM Ratings R JOIN Users U ON (R.UID=U.UID)
      GROUP BY U.Country
      ORDER BY avg(R.Score) DESC
```

(b) Given the Movies, Ratings, and Users tables above, write SQL queries to answer the following questions (in a way that is independent of the shown data): (**15 points**)

- Q4: Which users rated more than one movie (return the Name, and count)?

- Q5: Compute the average rating of all movies, sorted descending by average rating (return Title, and average rating with NULL for no rating).

- Q6: Which users did not rate any movies (return the UID, Name, and Country)?

Task 3 Query Processing (16 points)

(a) Assume relations $R(a, b, c)$ and $S(d, e)$, and indicate in the table below whether or not the two relational algebra expressions per row are equivalent in bag semantics (✓for equivalent, × for non-equivalent). For non-equivalent expressions, briefly explain why. (**4 points**)

Expression 1	Expression 2	Equivalent? Why Not?
$\sigma_{b=3 \wedge d<b}(R \bowtie_{a=e} S)$	$(\sigma_{b=3}(R)) \bowtie_{a=e} (\sigma_{d<3}(S))$	
$\sigma_{b>7}(\gamma_{b;\,\text{sum}(c)}(R))$	$\gamma_{b;\,\text{sum}(c)}(\sigma_{b>7}(R))$	
$\pi_{b,d}(R \bowtie_{a=e} S)$	$(\pi_{a,b}(R)) \bowtie_{a=e} (\pi_{d,e}(S))$	
$\sigma_{b>7}(R) \cap \sigma_{a<b \wedge b<a}(R)$	$\sigma_{b>7}(R)$	

(b) Draw two logical query trees for query Q2 from Task 2(a): once in unoptimized form (with intersection), and once in optimized form (without intersection). (**6 points**)

(c) Describe the conceptual ideas of a nested-loop join, and a hash join. Furthermore, assume $R \bowtie S$ with cardinalities $N = |R|$ and $M = |S|$, and enter the space and time complexity of these operators (in the open-next-close iterator model) in the table below. (**6 points**)

Operator	**Time Complexity**	**Space Complexity**
Nested Loop Join		
Hash Join		

Task 4 Transaction Processing (6 points)

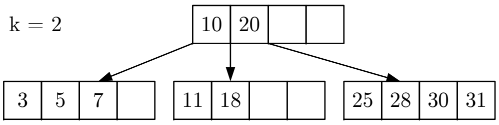
Explain the concept of a database transaction log (not locks), and how it helps to ensure Atomicity and Durability of changes made by uncommitted and committed transactions on failures.

Task 5 Distributed Data Analysis (5 points)

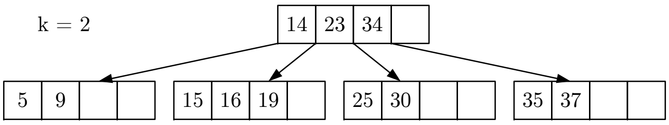
Explain Apache Spark's abstraction of Resilient Distributed Datasets (RDDs), and how it facilitates data-parallel computation in distributed environments.

Task 6 Physical Design (20 points)

(a) Given the B-tree (k=2) below, insert key 16, then insert 26, and draw the resulting B-tree. (**5 points**)

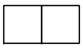


(b) Given the B-tree (k=2) below, delete key 14, then delete 37, and draw the resulting B-tree. (**5 points**)

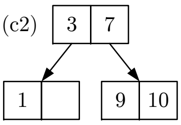


(c) Which of the following trees are valid—i.e., satisfy the constraints of—B-trees with k=1. Mark each tree as valid (✓), or invalid (×) and name the violations. (**5 points**)

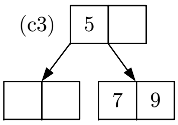
(c1)



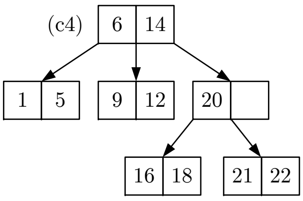
(c2)



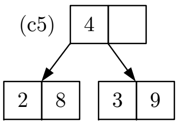
(c3)



(c4)



(c5)



(d) Given a relation $R(a, b, c)$ and a query workload Q09: $\sigma_{a<3}(R)$, Q10: $\sigma_{a<7}(R)$, Q11: $\sigma_{a\geq 7}(R)$, and Q12: $\sigma_{a<3 \wedge b=2}(R)$, find a disjoint and complete horizontal partitioning into three partitions R_1, R_2, and R_3 that improves the cost of all four queries. Provide the relational algebra expressions for partitioning and querying. (**5 points**)

R_1 :

R_2 :

R_3 :

Q09:

Q10:

Q11:

Q12: