Dr. Stefan Brass
School of Information Sciences
University of Pittsburgh

February 18, 2000

INFSCI 2710 "Database Management" — Solution to Example Final Exam II —

Exercise 1 (Translation ER to Relational Model) 10 Points

- Topic(<u>No</u>, Name, child_of^o → Topic)
- Staff(<u>Name</u>, Email)
- Webpage(<u>URL</u>, Label)
- Evaluated(<u>URL</u> → Webpage, Grade, Last_Checked, Name → Staff)
- refers_to(<u>URL</u> → Webpage, <u>No</u> → Topic, Priority)

An alternative would be to merge "Webpage" and "Evaluated". Then the columns "Grade", "Last_Checked", and "Name" can be null (for web pages not yet evaluated), but you must specify the constraint that they can only be null together.

A third possibility is to store evaluated webpages in one relation, and non evaluated webpages in another relation. This needs the constraint that the URLs in both relations are disjoint. Since this constraint cannot be specified declaratively in Oracle, DB2 or SQL Server, this solution is worse than the other two.

The attribute "Name" in "Evaluated" can also be called "by".

Exercise 2 (SQL CREATE TABLE) 4 Points

```
CREATE TABLE Ordered_Comp(
    OrdNo NUMERIC(5) NOT NULL,
    CNo NUMERIC(4) NOT NULL,
    CONSTRAINT Ordered_Comp_Key PRIMARY KEY(OrdNo, CNo),
    CONSTRAINT OrdNo_Exists FOREIGN KEY(OrdNo) REFERENCES Orders,
    CONSTRAINT CNo_Exists FOREIGN KEY(CNo) REFERENCES Components)
```

`DELETE CASCADES` would make sense for "`OrdNo`", but not for "`CNo`": If an order is deleted, all its order details should also be deleted. But if a component is deleted, equivalent components must be manually replaced. It is not possible to deliver the computer e.g. without graphics card because the ordered graphics card became unavailable.

Exercise 3 (SQL) 18 Points

a)
```
SELECT COUNT(*)
FROM   Orders O, Models M
WHERE  O.MNo = M.MNo
AND    M.CPU IN ('Celeron', 'Pentium II')
```

b)
```
SELECT DISTINCT O1.OrdNo
FROM   Ordered_Comp O1, Ordered_Comp O2, Components C1, Components C2
WHERE  O1.OrdNo = O2.OrdNo AND O1.CNo = C1.CNo AND O2.CNo = C2.CNo
AND    C1.CATEGORY = 'CD-ROM' AND C2.CATEGORY = 'DVD'
```

c)
```
SELECT   M.MNo, CPU, MHz, COUNT(*)
FROM     Models M, Orders O
WHERE    M.MNo = O.MNo
GROUP BY M.MNo, CPU, MHz
```

d)
```
SELECT OrdNo, Name
FROM   Orders
WHERE  OrdNo NOT IN(SELECT X.OrdNo
                    FROM   Ordered_Comp X, Components C
                    WHERE  X.CNo = C.No AND C.Category = 'Disk')
```

e)
```
SELECT   DISTINCT Name
FROM     Orders O, Ordered_Comp C
WHERE    O.OrdNo = C.OrdNo
GROUP BY O.OrdNo, Name
HAVING   COUNT(*) >= 5
```

f)
```
SELECT   O.OrdNo, SUM(C.Price) + M.Price "Total_Price"
FROM     Orders O, Models M, Ordered_Comp X, Components C
WHERE    M.MNo = O.MNo AND X.OrdNo = O.OrdNo AND C.CNo = X.CNo
GROUP BY O.OrdNo, M.Price
UNION
SELECT   O.OrdNo, M.Price "Total_Price"
FROM     Orders O, Models M
WHERE    M.MNo = O.MNo
AND      O.OrdNo NOT IN (SELECT OrdNo FROM Ordered_Comp)
```

Exercise 4 (Views, Security) 3+2+2=7 Points

a)
```
CREATE VIEW Pgh_Orders AS
            SELECT * FROM Orders WHERE OrdNo >= 90000
            WITH CHECK OPTION
```

b) `GRANT SELECT, INSERT ON Pgh_Orders TO pghshop`
Because of the check option defined in a) only orders with a number $\geq$ 90000 can be inserted.

c) Yes, the view is updatable (it is a selection from a single table). The result of translating the `DELETE` statement to the base table is:
```
DELETE FROM Orders
WHERE       Name = 'S. Brass'
AND         OrdNo >= 90000
```

Exercise 5 (BCNF) 3+2=5 Points

a) This relation has two minimal keys:

- `Room_No`, `From_Date`
- `Room_No`, `To_Date`

b) The relation is not in BCNF. The last three FDs all do not have a complete key on the left hand side:

- `From_Date, To_Date` $\longrightarrow$ `Num_of_Nights`
- `From_Date, Num_of_Nights` $\longrightarrow$ `To_Date`
- `To_Date, Num_of_Nights` $\longrightarrow$ `From_Date`

This corresponds to the fact that one of the three attributes is redundant (it can be computed from the other two).