CS 338 Assignment 4

University of Waterloo November 30, 2015

1 ER Model

1.1 Drawing

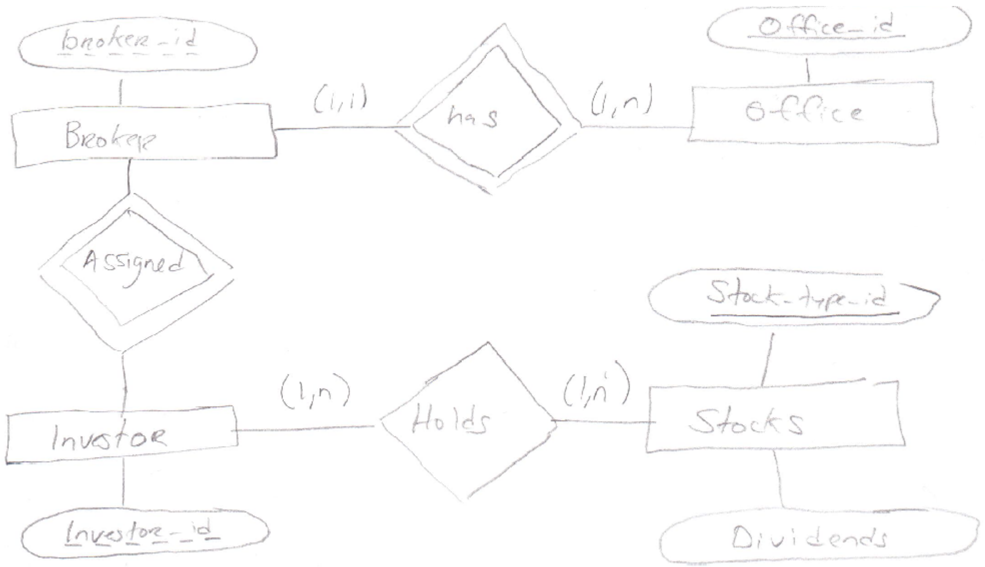

Figure 1: A sample ERM drawing, solutions may vary.

1.2 Schema

```
CREATE TABLE Broker(broker_id INT, office_id INT, PRIMARY KEY(broker_id));

CREATE TABLE Investor(investor_id INT, broker_id INT, PRIMARY KEY(investor_id), FOREIGN
KEY(broker_id) REFERENCES Broker(broker_id);

CREATE TABLE Stocks(stock_type_id INT, dividends DECIMAL(19,4), PRIMARY KEY(stock_type_id));

CREATE TABLE InvestorHoldings(investor_id INT, stock_type_id INT, CONSTRAINT is_pk
PRIMARY KEY(investor_id,stock_type_id), FOREIGN KEY(investor_id)
REFERENCES Investor(investor_id), FOREIGN KEY(stock_type_id) REFERENCES
Stock(stock_type_id));
```

2 Functional Dependencies

a) Show that if $X \to YZ$, then $X \to Y$.

We say that $A \to B$ if:
$\forall v_1, v_2, \ldots v_k, w_1, w_2, \ldots, w_k : R(v_1, \ldots, v_k) \wedge R(w_1, w_2, \ldots, w_k) \wedge (\wedge_{i \in A} v_i = w_i) \to (\wedge_{j \in B} v_j = w_j)$. By definition of $\to$, if $(\wedge_{i \in A} v_i = w_i) \to (\wedge_{j \in B} v_j = w_j)$, then $(\wedge_{i \in A} v_i = w_i) \to (\wedge_{j \in C} v_j = w_j)$, where $C \subseteq B$.

Therefore, if $X \to YZ$, then $X \to Y$, since $Y \subseteq YZ$.

b) Show that each relation has at least one superkey and one candidate key.

$K \subseteq R$ is a superkey if $K \to R$. Choose $K = R$ (the trivial superkey), then $K \to R$ so K is a superkey of R. A candidate key is a *minimal* superkey. Since we will always have at least one superkey, we can choose always choose a minimal key from them. Therefore, we have at least one candidate key.

c) Given a relation $R(A_1, A_2, \ldots, A_k, B_1, B_2, \ldots, B_k)$ with $2k$ attributes where $A_i \to B_i$ and $B_i \to A_i$ $\forall i : 1 \leq i \leq k$, how many *candidate keys* does the relation have?

A candidate key is a minimal superkey. In order to fully determine the relation, a key K *must* have one of A_i, B_i $\forall i : 1 \leq i \leq k$. Therefore, since there are two choices for each i, we have a total of 2^k candidate keys.

3 Normal Forms

Given $R(B, O, I, S, Q, D)$, with $S \to D$, $I \to B$, $IS \to Q$, $B \to O$.

a) Find all candidate keys for R.

We want to find a key K such that $K \to BIOSQD$. Since by definition $X \to X$, for any X, we can choose $K = IS$ to have $K \to IS$. Since $I \to B$, $S \to D$, and $IS \to Q$, $K \to ISBDQ$. Finally, by the transitivity property, since $B \to O$, $K \to ISBDQO$, so K is a superkey of R. Also, K is a candidate key for R because it is minimal, if we remove I or S from K, we no longer determine R. There are no other candidate keys, because $IS \to Q$, and nothing determines IS, other than itself. As such, IS must be a component of a key. Therefore all other keys would be a superset of K, so K is the only candidate key.

b) Find a lossless BCNF decomposition of R.

First we need to compute F^+. Then we find non-trivial functional dependencies in F^+ which violate the BCNF conditions and create a new relation for the functional dependency, repeating this process until the BCNF conditions are satisfied.

First, we see that $S \to D \in F^+$. We create a new relation $R_2(S, D)$, and remove D from R. Now, we have $(R(B, O, I, S, Q), R_2(S, D))$.
Now, $I \to B \in F^+$. We create a new relation $R_3(I, B)$, and remove B from R. Now we have $(R(O, I, S, Q), R_2(S, D), R_3(I, B))$.
$I \to O \in F^+$ (transitive property), We create a new relation $R_4(I, O)$, and remove O from R. Now we have $(R(I, S, Q), R_2(S, D), R_3(I, B), R_4(I, O))$.

All functional dependencies in F^+ can longer be applied to the individual relations to violate the BCNF conditions. As such, we $(R(I, S, Q), R_2(S, D), R_3(I, B), R_4(I, O))$ is a lossless BCNF decomposition.

c) Find a lossless, dependency preserving 3NF decomposition of R.

$F = (S \to D, I \to B, IS \to Q, B \to O)$. We note that F is a minimal set of dependencies (i.e. a cover for itself), because removing any one of the dependencies in F would change the closure F^+.

Now we apply the `Compute3NF` function from class. We take the first dependency $S \to D$, and make a new relation out of it - $R_2(S, D)$. We now proceed to the second dependency $I \to B$, and make a relation $R_3(I, B)$. We repeat this process until the algorithm completes, giving us relations: $(R_2(S, D), R_3(I, B), R_4(I, S, Q), R_5(B, O))$. Note that because R_4 has a candidate key for R, we don't need to add a special table for that.

d) In the above sample, we had a 3NF decomposition. Students should talk about whether their ER diagram was BCNF, 3NF, or otherwise, and explain the advantages and disadvantages their ER model had over the above decompositions.

4 Relational Algebra

a) $\pi_{\#2,\#7}(\sigma_{\#5=\#6}(\sigma_{\#1=\#4}(Author \times Wrote) \times Publication))$.
Note that duplicates are not allowed in relational algebra. To speed this up, we will want indexes to be created on the author attribute for wrote, the aid attribute for author, and the pubid attribute for publication.

b) $Publication - \pi_{\#1,\#2}(\sigma_{\#1=\#3}(Publication \times Article)))$
To speed this up, we will want an index to be created on the pubid attribute in article and the pubid attribute in article

c) $\sigma_{\#1=\#3}(Publication \times Journal) \cup \sigma_{\#1=\#3}(Publication \times Book)$
To speed this up, we will want indexes to be created on the pubid attributes in book, journal, and publication.