

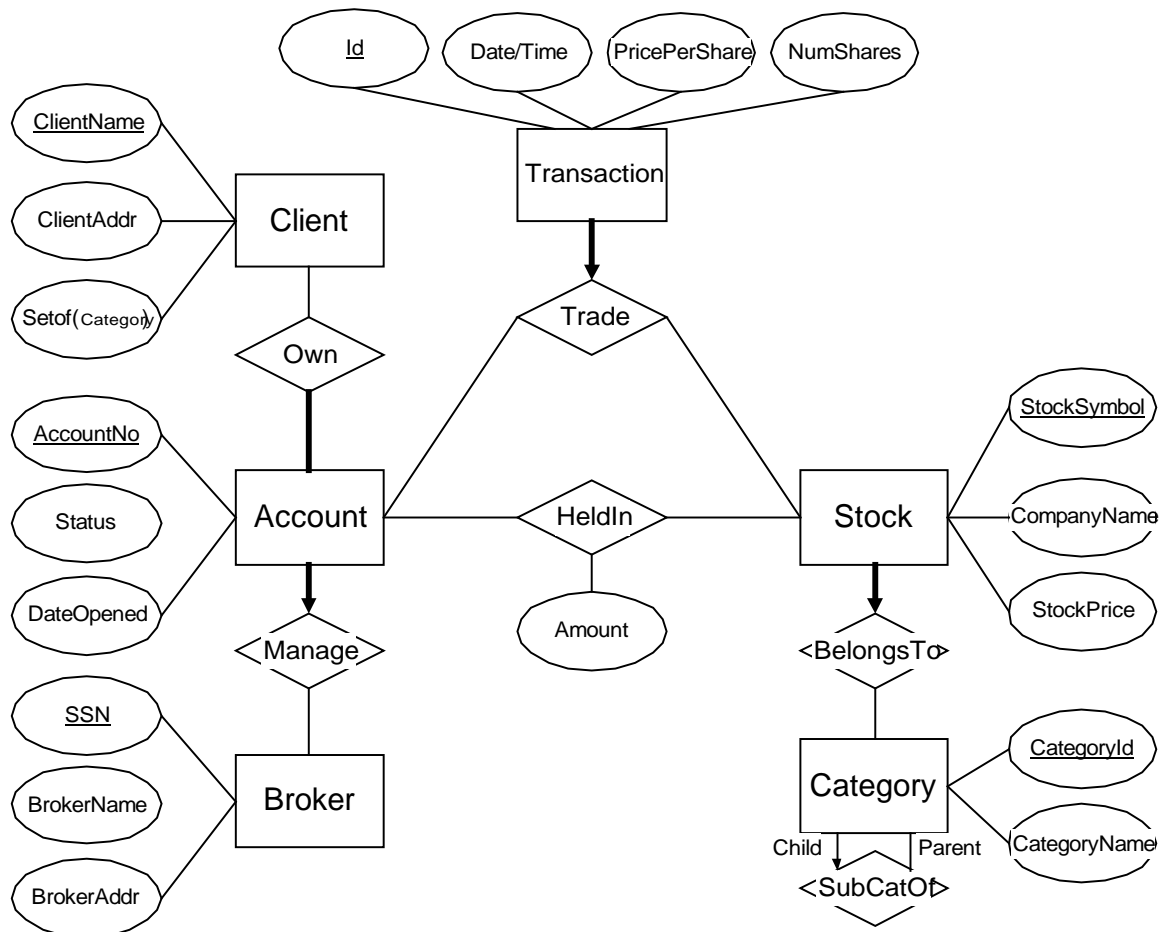
Solution to Midterm

Professor Smolka & Yan Rong

Nov, 2001

Any comment regarding the solution is welcome and you can send email to yanrong@cs.sunysb.edu

Problem 1.



Problem 2.

```
CREATE TABLE Client (  
    name VARCHAR(20),
```

```

        address VARCHAR(40),
        disallowdCat SETOF(INTEGER),
        PRIMARY KEY (name)
    )

CREATE TABLE Broker (
    ssn CHAR(9),
    name VARCHAR(20) NOT NULL,
    address VARCHAR(40),
    PRIMARY KEY (ssn)
)

CREATE TABLE Account (
    accountNo INTEGER,
    date DATETIME,
    status CHAR(1),
    ssn CHAR(9) NOT NULL,
    PRIMARY KEY (accountNo),
    FOREIGN KEY ssn REFERENCES Broker (ssn)
)

CREATE TABLE Own (
    name CHAR(9),
    accountNo INTEGER,
    PRIMARY KEY (name, accountNo),
    FOREIGN KEY name REFERENCES Client (name),
    FOREIGN KEY accountNo REFERENCES Account (accountNo)
)

CREATE ASSERTION hasOwner
CHECK ((SELECT COUNT (DISTINCT accountNo) FROM Account) =
      (SELECT COUNT (DISTINCT accountNo) FROM Own))

CREATE TABLE Category (
    id INTEGER,
    parentId INTEGER NOT NULL,      // root points to itself
    name VARCHAR(10),
    PRIMARY KEY (id),
    FOREIGN KEY parentId REFERENCES Category (id)
)

```

```
CREATE TABLE Stock (  
    symbol VARCHAR(10),  
    company VARCHAR(20),  
    price FLOAT,  
    categoryId INTEGER,  
    PRIMARY KEY (symbol),  
    FOREIGN KEY categoryId REFERENCES Category (id)  
)
```

```
CREATE TABLE HeldIn (  
    accountNo INTEGER,  
    symbol VARCHAR(10),  
    amount INTEGER NOT NULL,  
    PRIMARY KEY (accountNo, symbol),  
    FOREIGN KEY accountNo REFERENCES Account (accountNo),  
    FOREIGN KEY symbol REFERENCES Stock (symbol),  
)
```

```
CREATE TABLE Transaction (  
    id INTEGER,  
    accountNo INTEGER,  
    symbol VARCHAR(10),  
    dateTime DATETIME NOT NULL,  
    pricePerShare FLOAT NOT NULL,  
    numShares INTEGER NOT NULL,  
    PRIMARY KEY (id),  
    FOREIGN KEY accountNo REFERENCES Account (accountNo),  
    FOREIGN KEY symbol REFERENCES Stock (symbol)  
)
```

Problem 3.

```
1  
SELECT S.symbol  
FROM Stock S  
WHERE S.categoryId NOT IN (  
    SELECT C.disallowedCat  
    FROM Client C  
    WHERE C.name = ?)
```

```
SELECT S.symbol
FROM Stock S
WHERE S.price < 5.0
```

3

```
SELECT T1.symbol, SUM(T1.numShares) AS share1
FROM Transaction T1
GROUP BY T1.symbol
HAVING 5 > COUNT(
    SELECT T2.symbol, SUM(T2.numShares) AS share2
    FROM Transaction T2
    GROUP BY T2.symbol
    HAVING share1 < share2
)
```

4

```
SELECT *
FROM Client C
WHERE 3 < (
    SELECT count(*)
    FROM Category Cat
    WHERE Cat.id IN C.disallowedCat
)
```