

# COSC 265 – ASSIGNMENT 1

---

Zachary Sanson  
58520526

---

CREATED ON: 22/8/18

# University of Canterbury

## COSC265 assignment 2018

**Goal:** After completing this assignment you should be able to design EER schemas.

**Assessment:** worth 25%

**Due Date:** This assignment is due on 24 August, 2018, no later than 5:00 pm

**Late assignments:** Will only be accepted for one week after the due date, with a penalty for late submission of 15%. At one week after the due date (31 August, 2018, at 5:00 PM), no submissions will be accepted and 0 marks will be given to any student not having submitted the assignment by that date/time.

**Assignment Submission:** Submit your solution (EER diagram and the report) as well as the completed Plagiarism declaration form via Learn.

- **Both your report and diagram must include your first name, last name and student number.**
- **The report must be submitted in .PDF format, and the diagram must be submitted in .PNG format. Other file formats will not be marked.**
  - o To print your diagram to a .PNG file in EER-Tutor, click the Print Diagram button to create a print page, then Right-Click on that print page, and choose Save Image As, then give your image file a name but make sure the file type is .PNG, which is the default.

**Assignment Guidelines:** This assignment must be done individually using EER tutor. You should be aware of the University of Canterbury Academic Integrity Policy:

<http://www.canterbury.ac.nz/about/governance/ucpolicy/student/academic-integrity-guidance-for-staff-and-students/>

### The task

You are chosen to design a database for a bank. Below is a description of the data that needs to be recorded, maintained and accessed. Your submission must include the EER schema created with EER-Tutor, and a short written report. The report should clearly state any assumptions you make. You may also explain any complex components of your solution. You may also discuss any difficulties you faced in developing your EER schema. The report should not be longer than 500 words.

### The requirements

You need to design a database that will support the operation of a bank. The bank is organized into branches. In a city, there might be several branches. Each branch has a unique name, unique number and address. The bank monitors the assets of each branch. There is only one central branch.

Bank customers are identified by their customer ids, which are unique. The bank stores each customer's name, address (street name, street number and city), birth date, gender, IRD number, home phone number, and business phone number (if available). Customers may have accounts and can take out loans. A customer may be associated with a personal banker. A customer can have at most one account of each type with the bank. An account may be shared by customers, and a customer can have several accounts. There are several types of accounts: savings, checking, credit and loan accounts. Each account has a unique number. The bank maintains a record of each account's opening day, type, balance and also information about all transactions. For each customer associated with an account, the database should also store the user name for internet access and the chosen password (if appropriate). For each transaction (such as withdrawal or deposit), the database should store the date and time, the amount and the type of transaction. Withdrawals are available from checking and savings accounts, but not from credit cards or loans. For each withdrawal, it is necessary to store the manner in which the withdrawal was made (e.g. teller, ATM machine, or electronic transaction). Each ATM machine has a unique id and location. A deposit can be made via a teller (by cash or cheque) or electronically. If a customer has deposited a cheque, the database should store information about the cheque (unique cheque number, the bank that issued the cheque, the name on the cheque, date and amount).

A customer is issued a bank card, which covers all savings/checking/loan accounts for that customer. Each bank card has a unique number, the date of issue, expiry date and a PIN. There is a daily maximum for each bank card, which is determined by the bank based on the customer's history. For each account, a statement is issued every month, showing all transactions processed for that account since the previous report. Each report has a report number (within the current year), and shows the names of all customers associated with the account.

There are several types of savings accounts, with different interest rates. Each type of savings account has a unique name, the maximal age of the customer (when appropriate), monthly fee, the minimum amount of funds that the account must contain (optionally), the number of free staff-assisted transactions per month, the number of free internet transactions per month, service fee (for using ATM by other banks), overdraft debit interest rate, and a transaction fee.

At any time, there will be a number of blank cheques (i.e. a cheque book) issued for each checking account, and the database should store the numbers of issued cheques. A cheque number consists of the account number, plus a cheque number which is unique for that account. When processing a cheque, the database should store the date, the amount and the recipient.

A loan account has a unique loan number. The bank keeps track of the loan amount and the loan repayments. The frequency of repayments is known (monthly, weekly or bi-weekly), as well as the minimum amount for repayments. The customer may specify another account (savings or checking) for the loan to be repaid from. Each loan may have a fixed interest, or a floating (variable) interest. The date and amount are recorded for each payment. Interest rates for loans are published weekly by the bank.

Bank employees are identified by their staff numbers, which are also unique. The bank administration stores the name (first and last), home and office phone numbers of each employee, as well as the IRD number, gender, birth date, full address, position held and the current salary. The start date of the employment is also stored. Every employee works for a single branch. Each branch has a manager, who has overall responsibility for the management of the branch. Each bank employee is also an account holder, as his/her salary would be deposited into an account. The starting date for the manager is known.

Customers may also have credit cards with the bank. The bank issues VISA or MASTER cards. Each card has a unique number, the date of issue and the expiry date. For each credit account, there is a limit (i.e. the maximum amount the customer is allowed to charge to the card), and a maximum amount per transaction. Customers having credit cards get rewards for the total amount spent, which is a certain percentage of the sum of all payments made by the card. Each customer sharing the credit card is charged a yearly fee.

Each customer authorized for internet banking can perform the following operations on their accounts: check account balances, move funds between accounts, view and print transactions for a specified time period, and pay bills online. For paying bills online, it is necessary to specify the bank and account number for the receiver, the name of the person or the name of the company the payment is made to, the amount, the date when the payment is to be made (which might be in the future) and a comment to be sent to the receiver (such as invoice number). The database also contains information about registered companies (such as Spark NZ) which payments can be made to. In that case, the customer can simply select one of the registered companies from the list, and only specify the date, amount and the comment to be sent.

## **A few notes on marking**

Your solution will be marked on several aspects:

- Appropriateness of constructs you use. In EER diagrams, often there are several ways of representing the same collection of data. However, some of these ways would be more appropriate than others, due to various factors, such as the semantics of the domain, the space required to store the data, the typical kinds of operations that will be performed on the database etc. Make sure you analyze the requirements properly.
- Make sure you use the correct EER notation and naming guidelines.
- You are allowed to introduce additional assumptions. However, not every assumption is reasonable. Think carefully about any assumption you would like to add. You are welcome to discuss your assumptions with the lecturer/tutors.

# Plagiarism Declaration

This form needs to accompany your COSC 265 assignment submission.

I understand that plagiarism means taking someone else's work (text, program code, ideas, concepts) and presenting them as my own, without proper attribution. Taking someone else's work can include verbatim copying of text, figures/images, or program code, or it can refer to the extensive use of someone else's original ideas, algorithms or concepts.

I hereby declare that:

- My assignment is my own original work. I have not reproduced or modified figures/images, or writings of others without proper attribution. I have not used original ideas and concepts of others and presented them as my own.
- I have not allowed others to copy or modify my own figures/images, or writings. I have not allowed others to use original ideas and concepts of mine and present them as their own.
- I accept that plagiarism can lead to consequences, which can include partial or total loss of marks, no grade being awarded and other serious consequences, including notification of the University Proctor.

Name: Zachary Sanson

Student ID: 58520526

Signature: Euchinson

Date: 22/8/18

# Assignment 1

Wednesday, 15 August 2018

11:14 AM

## Assumptions

- Personal banker which is assigned to a customer is a bank employee.
- A customer cannot be an employee of the bank.
- Each banking statement can only contain information about one account at a time.
- A branch is assumed to only have one manager.
- Anyone who has an account under the bank must have a username/password for internet banking.  
(Even employees since they must own accounts themselves thus making them similar to customers.)  
-- *Otherwise we would need to make INTERNET\_BANKING\_SESSION a **weak entity**.*
- You cannot withdrawal cash electronically since you cannot possibly do so without interacting with a bank, although you can deposit electronically since this transfers directly to an account. A teller or authorised ATM machine is required to withdrawal cash.
- An account holder does not need to have a bank/credit card (e.g. If you are underaged then you should not have access to your accounts without your guardian being there to confirm the transfer).
- Because bank/credit cards change according to an account holders history and each have unique id's (e.g. The calculation of daily\_maximum and rewards), there can only be one card of each type associated with an account holder; although there can be many cards linked to the same account since people can share accounts.
- An employee can only work in one branch.
- An account must have an account owner.
- Transactions are created by the account holder (via a teller or ATM) or an electronic banking session.
- A branch must have at least, one employee to function and a branch must look after at least one account otherwise it is not a functioning branch.
- A cashier must be an employee of the branch.
- An account holder can only have one account of each type under their name.
- Each check account may have a cheque book associated with it; which is made up of many cheques of varying types (deposit/withdrawal). A cheque book must have cheques stored in the cheque book otherwise it is an empty cheque book (which doesn't make much sense for our case).

## Decisions

- The CHEQUE\_BOOK can be made up of several cheques so the Cheque\_Number is a **multi-valued attribute**.
- An ACCOUNT may have several transactions on it so there is a **multi-valued attribute** for the transactions called Transaction\_Ids which contains a list of the transactions for the account.
- A STATEMENT can be made up from many names/transactions on one statement, so these are both **multi-valued attributes** (Names, Transactions).
- Daily\_Maximum (under BANK\_CARD) is calculated from the account holders history and therefore is a **derived attribute**.
- Rewards (under CREDIT\_CARD) is calculated from the account holders history and therefore is a **derived attribute**.
- Intrest\_Rate (under ACCOUNT-SAVINGS) is calculated from the type of savings account it is and therefore is a **derived attribute**.
- CREDIT\_CARD:
  - o *Is a **weak entity** type as it relies on ACCOUNT-CREDIT being present.*
- BANK\_CARD:
  - o *Is a **weak entity** type as it relies on ACCOUNT-SAVINGS/CHECK/LOAN (BANK\_CARD\_ACCESSIBLE) being present.*
- BANK\_CARD\_ACCESSIBLE:
  - o *Is a set of accounts (**category entity**) that the bank card is allowed to modify.*
  - o *We use the **overlap-specialisation** with the ACCOUNT-(sub-accounts) so that BANK\_CARD\_ACCESSIBLE may be acquainted with any of those three sub-accounts.*

- WITHDRAWAL\_ACCESSIBLE:
  - *Is a set of accounts (**category entity**) that WITHDRAWAL is allowed to modify.*
  - *We use the **overlap-specialisation** with the ACCOUNT-(SAVINGS/CHECK) so that WITHDRAWAL\_ACCESSIBLE may be acquainted with any of those three sub-accounts.*
- ACCOUNT HOLDER:
  - *Is a **category entity** which has the inherited attributes of either CUSTOMER or BANK\_EMPLOYEE depending on the Is\_Staff value. This allows for the accounts to have account holders which are a mix of types (customers & employees) because a customers account could be shared with an employee of the bank. So all actions on accounts needs to be able to work for both customers and employees.*
- ACCOUNT-(sub-classes):
  - *An account may be of four types: savings, check, credit and loan. So these are set as **sub-classes** for ACCOUNT.*
  - *This allows the database to access each sub-account individually. There is a **disjoint-specialisation** for these sub-classes because an account can only be one type of account (savings/ check/ credit/ loan) at a time.*
- TRANSACTION:
  - *Contains three **sub-classes** as you can either have a WITHDRAWAL, TRANSACTION or ELECTRONIC\_TRANSFER.*
  - *The TRANSACTION entity does not have the ability to modify accounts, but the sub-classes can modify very specific accounts in accordance to their needs. Our sub-classes are joint with a **disjoint-specialisation** as the transaction should only be of one type at a time (deposit or withdrawal).*
- DEPOSIT:
  - *Is created from three **super-classes** (TRANSACTION, transaction type and TELLER).*
  - *The main super-class being TRANSACTION which inherits all the transaction details. The transaction type 'super-class' is the result of a **category-specialisation** so that a deposit has to either inherit the attributes from CHEQUE or CASH depending on the type of deposit we have. The TELLER super-class so we can inherit the details about the teller they processed the deposit with.*
- WITHDRAWAL:
  - *Is created from two **super-classes** (TRANSACTION and transaction type).*
  - *The main super-class being TRANSACTION which inherits all the transaction details to the withdrawal. The transaction type 'super-class' which is a result of a **category-specialisation** so that withdrawal must inherit attributes from the TELLER or ATM depending on the type of withdrawal they have created.*

## Problems

- Whilst trying to create the bank schema I came across the problem that both customers and employees could have accounts and if this was true then they should be able to share their account with a person of the other type, so to fix this issue I had to categorise both an employee and customer into an account holder.
- Trying to figure out how to create the database for the transaction structuring was a challenge, having to figure out specifics on what withdrawals/deposits can directly change or the specifics they need to function and having electronic banking on top of all that created some interesting interactions.

