

银行业务管理系统数据库设计

学号：PB19071535 姓名：徐昊天

1. 概念模型设计

1.1 实体设计

实体的设计结果及理由如下：

- 客户

属性：

- 客户姓名。
- 客户联系电话。
- 客户家庭住址。
- 客户身份证号(主键)。

理由：作为客户实体，包含需求中涉及的所有客户属性。由于银行的客户通过其身份证号来标识，故身份证号作为主键。

- 联系人

属性：

- 联系人姓名(主键)。
- 联系人手机号。
- 联系人Email。
- 联系人住址。
- 联系人关系。

理由：作为联系人实体，包含需求中涉及的所有联系人属性。并将联系人姓名作为唯一标识，即为主键。

- 账户

属性：

- 账户号(主键)。
- 余额。
- 开户日期。

理由：作为账户实体，包含需求中涉及的所有账户属性。并将账户号作为唯一标识，即为主键。

- 支票账户

属性：

- 透支额。

理由：作为账户的子类，具有储蓄账户没有的透支额属性。

- 储蓄账户

属性：

- 利率。
- 货币类型。

理由：作为账户的子类，具有支票账户没有的利率和货币类型属性。

- 员工

属性：

- 员工姓名。
- 员工电话号码。
- 员工家庭地址。
- 员工身份证号(主键)。
- 开始工作日期。

理由：作为员工实体，包含需求中涉及的所有员工属性。并将员工身份证号作为唯一标识，即为主键。

- 部门

属性：

- 部门号(主键)。
- 部门名称。
- 部门类型。

理由：作为部门实体，包含需求中涉及的所有部门属性。并将部门号作为唯一标识，即为主键。

- 支行

属性：

- 城市。

- 支行名字(主键)。
- 资产。

理由：作为支行实体，包含需求中涉及的所有支行属性。并将支行名字作为唯一标识，即为主键。

- 贷款

属性：

- 贷款号(主键)。
- 贷款金额。

理由：作为贷款实体，包含需求中涉及的所有贷款属性。并将贷款号作为唯一标识，即为主键。

- 支付

属性：

- 支付号(主键)。
- 支付金额。
- 支付日期。

理由：作为支付实体，包含需求中涉及的所有支付属性。并将支付号作为唯一标识，即为主键。

1.2 联系设计

联系的设计结果及理由如下：

- 客户：联系人 (1: 1)

一位客户只需提供一为联系人的信息。

- 部门：员工 (1: n)

一个部门有多个员工。

- 员工：部门 (1: n)

部门中多个员工只有一个为部门经理。

- 支行：员工 (1: n)

一个支行中有多位员工。

- 支行：贷款 (1: n)

一个支行可发放多笔贷款。

- 贷款：支付 (1: n)

一笔贷款存在逐次支付的情况。

- 客户：员工 (m: n)

多位员工可负责多位客户。

- 客户：账户 (m: n)

账户可以由多个客户所共有，一个客户也可开设多个账户。

- 客户：支行 (m: n)

多位客户可与多个支行合作。

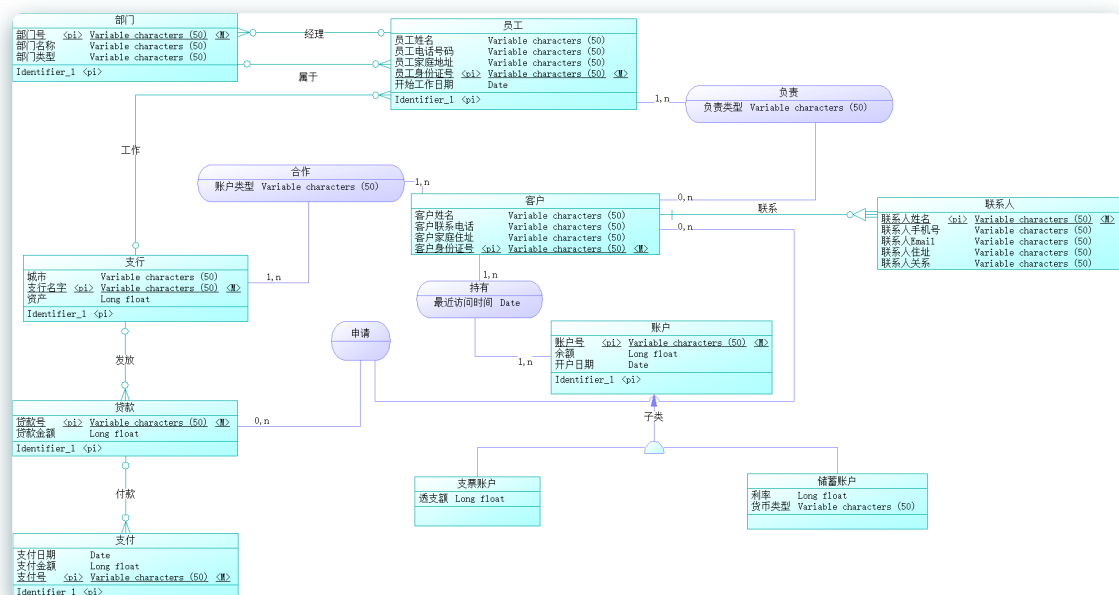
- 客户：贷款 (m: n)

贷款可以由一个或多个客户所共有，一个客户也可多次贷款。

- 支票账户与账户存在继承关系，支票账户是账户的子类。

- 储蓄账户与账户存在继承关系，储蓄账户是账户的子类。

1.3 Power Designer 的 ER 图



2. 概念模型到逻辑模型的转换

2.1 实体转换

实体转换后得到关系模式：

- 客户(客户姓名,客户联系电话,客户家庭住址,客户身份证号)。
- 联系人(联系人姓名,联系人手机号,联系人Email,联系人住址,联系人关系)
- 账户(账户号,余额,开户日期)
- 支票账户(账户号,透支额)
- 储蓄账户(账户号,利率,货币类型)
- 员工(员工姓名,员工电话号码,员工家庭地址,员工身份证号,开始工作日期)
- 部门(部门号,部门名称,部门类型)
- 支行(城市,支行名字,资产)
- 贷款(贷款号,贷款金额)
- 支付(支付号,支付金额,支付日期)

2.2 联系转换

考虑每个联系：

- 客户：联系人 (1：1)

客户(客户姓名,客户联系电话,客户家庭住址,客户身份证号)

联系人(联系人姓名,联系人手机号,联系人Email,联系人住址,联系人关系,客户身份证号)

- 部门：员工 (1：n)

部门(部门号,部门名称,部门类型)

员工(员工姓名,员工电话号码,员工家庭地址,员工身份证号,开始工作日期,部门号)

- 员工：部门 (1：n)

员工(员工姓名,员工电话号码,员工家庭地址,员工身份证号,开始工作日期,部门号)

部门(部门号,部门名称,部门类型,员工身份证号)

- 支行：员工 (1：n)

支行(城市,支行名字,资产)

员工(员工姓名,员工电话号码,员工家庭地址,员工身份证号,开始工作日期,部门号,支行名字)

- 支行：贷款 (1：n)

支行(城市,支行名字,资产)

贷款(贷款号,贷款金额,支行名字)

- 贷款：支付 (1: n)

贷款(贷款号,贷款金额,支行名字)

支付(支付号,支付金额,支付日期,贷款号)

- 客户：员工 (m: n)

增加模式：

负责(客户身份证号,员工身份证号,负责类型)

- 客户：账户 (m: n)

增加模式：

持有(客户身份证号,账户号,最近访问时间)

- 客户：支行 (m: n)

增加模式：

合作(客户身份证号,支行名字, 账户类型)

- 客户：贷款 (m: n)

增加模式：

申请(客户身份证号,贷款号)

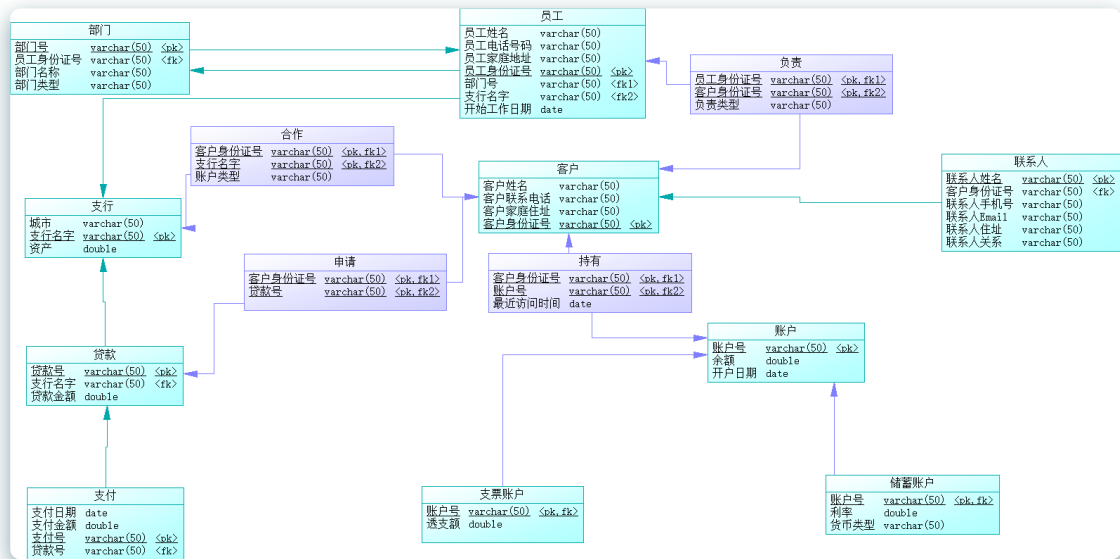
2.3 最终的关系模式

1. 客户(客户姓名,客户联系电话,客户家庭住址,客户身份证号)
2. 联系人(联系人姓名,联系人手机号,联系人Email,联系人住址,联系人关系,客户身份证号)
3. 账户(账户号,余额,开户日期)
4. 支票账户(账户号,透支额)
5. 储蓄账户(账户号,利率,货币类型)
6. 员工(员工姓名,员工电话号码,员工家庭地址,员工身份证号,开始工作日期,部门号,支行名字)
7. 部门(部门号,部门名称,部门类型,员工身份证号)
8. 支行(城市,支行名字,资产)
9. 贷款(贷款号,贷款金额,支行名字)
10. 支付(支付号,支付金额,支付日期,贷款号)
11. 负责(客户身份证号,员工身份证号,负责类型)

- 12. 持有(客户身份证号,账户号,最近访问时间)
- 13. 合作(客户身份证号,支行名字, 账户类型)
- 14. 申请(客户身份证号,贷款号)

3. Mysql 数据库结构实现

3.1 Power Designer 的 PDM 图



3.2 数据库表定义

表1.客户表(Client)

列名	中文含义	类型(长度)	允许为空 (NULL)	是否主键(Primary Key)	是否外键
C_name	客户姓名	varchar(50)	是	否	否
C_number	客户联系电话	varchar(50)	是	否	否
C_address	客户家庭住址	varchar(50)	是	否	否
C_id	客户身份证号	varchar(50)	否	是	否

表2.账户表(account)

列名	中文含义	类型(长度)	允许为空 (NULL)	是否主键(Primary Key)	是否外键
account_id	账户号	varchar(50)	否	是	否

列名	中文含义	类型(长度)	允许为空(NULL)	是否主键(Primary Key)	是否外键
balance	余额	double	是	否	否
open_date	开户日期	date	是	否	否

表3.联系人表(contact)

列名	中文含义	类型(长度)	允许为空(NULL)	是否主键(Primary Key)	是否外键
Con_name	联系人姓名	varchar(50)	否	是	否
Con_number	联系人手机号	varchar(50)	是	否	否
Con_address	联系人住址	varchar(50)	是	否	否
C_id	客户身份证号	varchar(50)	是	否	是
Con_email	联系人Email	varchar(50)	是	否	否
relation	联系人关系	varchar(50)	是	否	否

表4.支票账户表(check_account)

列名	中文含义	类型(长度)	允许为空(NULL)	是否主键(Primary Key)	是否外键
account_id	账户号	varchar(50)	否	是	是
overdraft	透支额	double	是	否	否

表5.支票账户表(save_account)

列名	中文含义	类型(长度)	允许为空(NULL)	是否主键(Primary Key)	是否外键
account_id	账户号	varchar(50)	否	是	是

列名	中文含义	类型(长度)	允许为空(NULL)	是否主键(Primary Key)	是否外键
interest_rate	利率	double	是	否	否
currency_type	货币类型	varchar(50)	是	否	否

表6.员工表(employee)

列名	中文含义	类型(长度)	允许为空(NULL)	是否主键(Primary Key)	是否外键
E_name	员工姓名	varchar(50)	是	否	否
E_number	员工手机号	varchar(50)	是	否	否
E_address	员工家庭地址	varchar(50)	是	否	否
E_id	员工身份证号	varchar(50)	否	是	否
department_id	部门号	varchar(50)	是	否	是
B_name	支行名字	varchar(50)	是	否	是
work_date	开始工作日	date	是	否	否

表7.部门表(department)

列名	中文含义	类型(长度)	允许为空(NULL)	是否主键(Primary Key)	是否外键
department_id	部门号	varchar(50)	否	是	否
department_name	部门名称	varchar(50)	是	否	否
department_type	部门类型	varchar(50)	是	否	否
E_id	员工身份证号	varchar(50)	是	否	是

表8.支行表(bank)

列名	中文含义	类型(长度)	允许为空(NULL)	是否主键(Primary Key)	是否外键
city	城市	varchar(50)	是	否	否
B_name	支行名字	varchar(50)	否	是	否
assets	资产	double	是	否	否

表9.贷款表(loan)

列名	中文含义	类型(长度)	允许为空(NULL)	是否主键(Primary Key)	是否外键
L_id	贷款号	varchar(50)	否	是	否
B_name	支行名字	varchar(50)	是	否	是
L_ammount	贷款金额	double	是	否	否

表10.支付表(payment)

列名	中文含义	类型(长度)	允许为空(NULL)	是否主键(Primary Key)	是否外键
P_id	支付号	varchar(50)	否	是	否
P_date	支付日期	varchar(50)	是	否	否
P_amount	支付金额	double	是	否	否
L_id	贷款号	varchar(50)	是	否	是

表11.持有表(own)

列名	中文含义	类型(长度)	允许为空(NULL)	是否主键(Primary Key)	是否外键
C_id	客户身份证号	varchar(50)	否	是	是

列名	中文含义	类型(长度)	允许为空(NULL)	是否主键(Primary Key)	是否外键
visit_date	最近访问时间	date	是	否	否
account_id	账户号	varchar(50)	否	是	是

表12.负责表(response)

列名	中文含义	类型(长度)	允许为空(NULL)	是否主键(Primary Key)	是否外键
C_id	客户身份证号	varchar(50)	否	是	是
E_id	员工身份证号	varchar(50)	否	是	是
R_type	负责类型	varchar(50)	是	否	否

表13.申请表(apply)

列名	中文含义	类型(长度)	允许为空(NULL)	是否主键(Primary Key)	是否外键
C_id	客户身份证号	varchar(50)	否	是	是
L_id	贷款号	varchar(50)	否	是	是

表14.合作表(associate)

列名	中文含义	类型(长度)	允许为空(NULL)	是否主键(Primary Key)	是否外键
C_id	客户身份证号	varchar(50)	否	是	是
B_name	支行名字	varchar(50)	否	是	是
A_type	账户类型	varchar(50)	是	否	否

4. 总结与体会

设计工作：

- 首先将文档需求中出现的实体及其必需的属性总结出来并分别设计实体。
- 其次结合需求中提及到的联系关系，将实体之间联系起来。
- 最后综合所有需求内容，对实体及其联系关系进行修改和优化。

体会：

- 首次使用powerdesigner工具绘图，对数据库的概念模型和逻辑模型有了更加深刻的了解。
- 通过设计数据库，对课堂上学习的理论知识有了进一步的理解。