Collaborators: Jiayu Gu, Denghao Sun

Date: October 29, 2023

NetID: jg7956, ds6963

Section: 001

**Database Systems Project Part I - Modeling**

**Affirmation of our independent efforts: \_\_\_\_Denghao Sun, Jiayu Gu\_\_\_\_\_\_\_\_\_\_\_**

**ER Diagram Documentation**

1. entities

(1) **customers**

* SSN(primary key)
* CusLastName
* CusFirstName
* CusMiddName
* CustSuffix
* CusDOB
* CustomerAddress
  + CustAddress
  + CustCity
  + CustState
  + CustZip
  + AnnunalStartDate
  + AnnualEndDate
* CustSalutation
* CusteMailAddress
* Gender
* CustomerLegacyID
* WithholdingCode
* StartDate
* EndDate
* CustomerAliAliasLastName
  + AliasFirstName
  + AliasMiddleInitial
  + AliasSuffix
  + AliasDOB
  + AliasSalutation
  + AliasMailAddressas
* Customerlmage
  + DucumentID
  + ImageType
  + ImageFileLocation
  + DateReceived

(2) **Invoice**(weak entity:customer,Billing account)

* InvoiceNumber
* PaidDate
* DueDate
* RunDate
* InvoceLineNumber
* DataillD
* ConversionPendingFlag
* PainAheadFlag
* PremiumSubTotal

(3) **Remittance**(weak entity:Customer,Claim,Contract,Billing Address)

* RemittanceFreq
* RemittanceID
* RemittanceDate
* PaymentMethod
* PaymentDate

(4) **Account**

* AccountName
* AccountID(primary key)
* LocationAddress
* Location City
* Location State
* LocationZip
* TaxIDNumber
* NumberOfEmployees
* NumberOfEmployessDate
* ActivityStatus
* GroupNumber
* **AccountEligiblity(composite attribute)**
  + AccountFlag
  + BillingAccountFlag
  + StartDate
  + Description
  + EndDate
* **CompanyCode(composite attribute)**
  + CompanyName
  + LegacyCompanyNo
* LegacyFlexID
* AccountEstablishedDate
* PlanYearStartDate
* PlanYearEndDate
* SubsequentYearStartDate
* IndustryDescription
* DualCompanyFlag
* ComplexAccountFlag
* StandardIndustryCode
* AnnualizedPremuim
* NoOuststandingInvoices
* NoMonthsInactive
* LastInvoicePaidDate
* LastInvoicePaidDueDate
* LastInvoiceGenDate
* NextInvoiceGenDate
* LastServiceCallDate
* LastBillCount
* DisabilityOfferingStartDate
* LocationPhone
* AddressInformationSource
* WebAddress
* SpecialHandingCode
* MultiLocationAccountFlag
* PEOFlag
* DisabilityOfferingTaxStatus
* TransitOneFlag
* HSAFlag
* HRAFlag
* DataConfidenceLevel
* TotalPolicyCount
* PendingAnnualizedPremiun
* PercentByLineOfBusiness
* SchedledLapseDate
* PenetrationPercentage
* NoFSAParticiants
* 55 FLEXAgreement**(composite attribute)**
  + TransitOneFlag
* **AccountLEgacyAlias**
  + AliasSource
  + AliasID
  + AliasName
  + AliasAddress
  + AliasCity
  + AliasState
  + AliasZip
  + AliasPhone
  + AliasEmailAddress
  + AliasFax

(5)**AccAdmin**

* + **AdminRole**
    - AccountFlag
    - BAcctFlag
  + AdminDescription
  + AdminID
  + AdminLastName
  + AdminFirstName
  + AdminMiddleInitial
  + Adminsuffix
  + AdminAddress
  + AdminCity
  + AdminState
  + AdminZip
  + Phone
  + FaxNumber
  + Gender
  + EmailAddress

(6) **BillingAccount**(belong to account)

* BAcctID
* BAcctName
* BAcctName2
* BillingAddress1
* BillingAddress2
* BillingCity
* BillingState
* BillingZip
* OnlineBillingFlag
* ActivityStatus
* ActivityStatusDate
* PayrollProcessorFlag
* BillingPhone
* BillingAcctTypeDate
* SpecialHandingCode
* ChangeFilesFlag
* EnrollmentFileFlag
* DebitCardFlag
* BillingFileFlag
* FTPSite
* NextVisitDate

(7) **Product**

* LineOfBusiness
* Description
* **ProductSeries**
  + SeriesName
  + PlanName
  + RatebookLocationCode
  + AnnualizedPremium
  + PSGName

(8) **Contract**

* ContractNumber
* ActivityStatus
* ActivityStatusDate
* CoverageType
* BillingMethod
* SuspendCode
* ExceptionCode
* ModalPremium
* AutoPremium
* CreditCardNo
* ExpirationDate
* CardType
* BankingTransitNumber
* BankingAccountType
* BankingAccountNumber
* PremiumPaymentLimit
* SubstandardRate
* ValuationInterestCode
* SupplementalBenefitPlan
* SpecialAssemblyCode
* InForceFlag
* PayUpDate
* Language
* **AHPolicy**
  + ContractNumber
* **ContractUnderwriting**
  + Underwriter
  + DocumentID
  + ImageFileLocation
  + Karz
  + DateReceived
* **LifePolicy**(belong to Product and Contract)
  + InvestmentCode
  + DeclaredSourceCode
  + InterestType
  + Principal
  + InterestRate
  + AccruedInterest
  + BillingCode
  + PaidToDate
  + LastActivityDate
  + LegalLastName
  + LegalFirstName
  + LegalMiddleInitial
  + LegamNameChangeReason
  + CashValue
  + Mortality

(9) **Claim**

* ClaimNumber
* ClaimDate
* SettlementDate
* WellnessEligibilityDate
* **ClaimImage**
  + DocumentID
  + DocumentClass
  + ImageType
  + DateReceived
  + ProcessedDate

(10) **Associate**

* AssocLastName
* AssocFirstName
* AssocMiddleInitial
* AssocSuffix
* AssocDOB
* TenureDate
* **WritingNumber**
  + writingNumber
  + IssueDate
  + Status
  + VestedFlag
  + TerminationDate
  + ReinstatemntDate
  + NoPayRateNumberator
  + NoPayRateDenominator
  + PayAdvance
  + PayAdvanceName
  + Amount
* Assoc\_DBAs
  + DBA\_TIN
  + AssocPhone
  + ConsolAssocID
  + CorpEMailAddress
  + AssocAddress1
  + AssocAddress2
  + AssocCity
  + AssocState
  + AssocZip
  + Description
  + StartDate
  + NationalLunsuranceID

(11) **Territory**

* TerritoryName
* StartDate
* EndDate
* StateOperation
  + StateOperationName
  + TerritoryName(FK)
  + StateOperationDivision
    - StateOperationDivisionName
    - StateCode
    - EndDate
    - **Region**
      * RegionName
      * EndDate
      * **District**
        + DistrictName
        + EndDate
* CoordPositionProdChain
  + EndDate
  + DistrictName(FK)

(12) **Employee**

* EmployeeID
* EmpLastName
* EmpFirstName
* EmpMiddleInital
* EmpAddress1
* EmpAddress2
* EmpCity
* EmpState
* EmpZip

(13) **ManagerContract(weak entity, contract,territory,associate)**

* SitCode
* IssueDate
* ContractType
* ContractSignDate
* ContractProcessDate
* CommissionCode
* EndDate

(14)**AccMember**

* StartDate
* FSAContricutionAmount
* CustBAcctDepartmentName
* EndDate

(15)**ContractPremium**

* + AnnualizedPremium
  + ProcessDate
  + AppSignDate
  + **Premium\_MgmtContract**
    - Amount
    - CommissionRate

2. Relationship

(1) Account\_Member(Customer and Account)

* Attributes:
  + AccountId(FK)
  + CustomerSSN(FK)
  + StartDate
  + EndDate
* Cardinality Ratio:
  + One-to-Many
* Structural Constraints:
  + CUSTOMER: (1, N) can have one or more accounts.
  + ACCOUNT: (1, 1) is related to exactly one customer.
* Correctness: The relationship ensures that an account is associated with one and only one customer, while a customer can have one or multiple accounts, maintaining the integrity of customer-account associations.

(2) Account\_BillingAccount( Account and BillingAccount )

* Attributes:
  + AccountId(FK)
  + BAcctID(FK)
  + relationshipType
  + StartDate
  + BillingFrequency
  + NonBillableMonths
  + EnrollmentPEriodLength
  + FSAClaimsReimbursementMethod
  + FSAPlanType
  + FSA\_URMCap
  + SpecificationCode
  + AccountType
    - RCodeAccountFlag
    - RCodeAssocFlag
    - RcodeCustomerFlag
  + PaymentCardFlag
  + DepartmentCode
  + FICAExemptionFlag
  + ContributionPercentage
  + HighDebuctibleMedicalPaymentPlanFlag
  + MedicalHealthInsuranceFlag
  + SinglePointBillingFlag
  + ExpressRecnciliationFlag
  + FSAService Fee
  + GracePeriodLength
* Cardinality Ratio:
  + Many-to-Many
* Structural Constraints:
  + ACCOUNT: (1, N) Each Account can be associated with one or more billing accounts.
  + BILLINGACCOUNT: (1, N) Each BillingAccount can be associated with one or more Accounts.
* Correctness: The relationship guarantees that each Account can have multiple billing accounts and vice versa, ensuring comprehensive billing possibilities for each account.

(3) Account\_Product(account and product)

* Attributes:
  + ProductLineOfBusiness(FK)
  + AccountId(FK)
  + StartDate
  + EndDate
* Cardinality Ratio:
  + Many-to-Many
* Structural Constraints:
  + ACCOUNT: (1, N) An account can have many associated products.
  + PRODUCT: (1, N) A product can be linked to many accounts.
* Correctness: The relationship ensures that an account can have multiple products and vice versa.

(4) ContractingPartyRole(customer, contract)

* Attributes:
  + ContractNumber(FK)
  + CustomerSSN(FK)
  + Role
  + Description
* Cardinality Ratio:
  + Many-to-Many
* Structural Constraints:
  + CUSTOMER: (1, N) A customer can play roles in one or more contracts.
  + CONTRACT: ​​(1, N) Each contract can have multiple customers playing various roles.
* Correctness: A customer can participate in multiple contracts with different roles, and each contract can involve multiple customers, each having specific roles

(5) CoordPosition\_ProductionAsset(Contract and Territory)

* Attributes:
  + ProductionCreditSplitPercentage
  + ContractNumber(FK)
  + DistricName(FK)
* Cardinality Ratio:
  + Many-to-Many
* Structural Constraints
  + Associate:(1,1): Each ProductionAsset has one ContractNumber

(6) Account\_Associate(Account and Associate)

* Attributes:
  + CompanyCode(FK)
  + Level(FK)
  + StopDate
  + ServiceType
* Cardinality Ratio:
  + Many-to-Many

(7) CustomerInvoice(customer, invoice)

* Attributes:
  + InvoiceNumber(FK)
  + CustomerSSN(FK)
* Cardinality Ratio:
  + Many-to-Many
* Structural Constraints:
  + CUSTOMER: (1, N) A customer can have one or more invoices.
  + INVOICE: (1, N) Each invoice can belong to one or more customers.
* Correctness: A customer can have multiple invoices, and each invoice can correspond to multiple customers

(8) ClaimRemittance(claim, remittance)

* Attributes:
  + ClaimNumber(FK)
  + RemittanceFreq(FK)
* Cardinality Ratio:
  + Many-to-Many
* Structural Constraints:
  + CLAIM: (1, N) A claim can have one or more remittances.
  + REMITTANCE: (1, N) Each remittance can correspond to one or more claims.
* Correctness: A claim can have multiple remittances (payments or settlements), and each remittance can correspond to multiple claims (though this scenario may be less common unless there's a consolidated payment for multiple claims).

(9) ContractBenefit(product, contract)

* Attributes:
  + ContractNumber(FK)
  + LineOfBusiness(FK)
  + PolicyCountContribution
* Cardinality Ratio:
  + 1-to-Many
* Structural Constraints:
  + PRODUCT: (1:N) A product has one or more contract
  + CONTRACT: (1:1) Each contract number just belongs to one product.
* Correctness: Each product has multiple contracts, and each contract just belongs to one product.

(10) TerritoryCoordinator(Employee and Territory)

* Attributes:
  + TerritoryName(FK)
  + EmployeeID(FK)
  + EndDate
* Cardinality Ratio:
  + 1-to-1
* Structural Constraints:
  + Employee: one employee mange one territory
  + Territory: one territory is managed by one employee
* Correctness: Each territory only managed by one employee. One employee only can manage one territory.

(11) AdminAccountLink(accAdmin, account)

* Attributes:
  + AdminID(FK)
  + AccountID(FK)
* Cardinality Ratio:
  + Many-to-Many
* Structural Constraints:
  + ACCADMIN: (1, N) An AccAdmin can have roles in one or more accounts.
  + ACCOUNT: (1, N) Each account can have one or more AccAdmins assigned.
* Correctness: An AccAdmin can be associated with multiple accounts, and each account can have multiple AccAdmins, each possibly having different roles or permissions.

InvoiceBillingLink(invoice, billingAccount)

* Attributes:
  + InvoiceNumber(FK)
  + BAcctID(FK)
* Cardinality Ratio:
  + Many-to-Many
* Structural Constraints:
  + INVOICE: (1, N) An invoice can be associated with one or more billing accounts (though typically, an invoice is linked to just one billing account).
  + BILLINGACCOUNT: (1, N) Each billing account can have one or more invoices.
* Correctness: An invoice can be linked to multiple billing accounts. Similarly, each billing account can have multiple invoices.

(12)LinkAccAndAccMem(Account and AccMem)

* Attributes:
  + AccountName(FK)
  + AccountMemberName
* Cardinality Ratio:
  + 1-to-Many
* Structural Constraints:
  + Account: (1, N): A account has one or more account member
  + AccMem: (1, 1): A account member belong to one account.
* Correctness: An account has one or more account members. However, an account member only belongs to one account.

(13)ContractInvoiceLink(contract, invoice)

* Attributes:
  + ContractNumber(FK)
  + InvoiceNumber(FK)
* Cardinality Ratio:
  + Many-to-Many
* Structural Constraints:
  + CONTRACT: (1, N) A contract can be associated with one or more invoices.
  + INVOICE: (1, N) Each invoice can be linked to one or more contracts.
* Correctness: A contract can have multiple invoices. Similarly, an invoice might reference multiple contracts.

(14) CustomerBindsInvoice(customer, invoice)

* Attributes:
  + InvoiceNumber(FK)
  + CustomerSSN(FK)
* Cardinality Ratio:
  + Many-to-Many
* Structural Constraints:
  + CUSTOMER: (1, N) A customer can have one or more invoices.
  + INVOICE: (1, N) Each invoice can belong to one or more customers.
* Correctness: A customer can have multiple invoices, and each invoice can correspond to multiple customers (though the latter scenario is less common unless there's a shared billing scenario).

(15)LinkOfAccAndMangerContract(Account and ManagerContract)

* Attributes:
  + AccountID(FK)
  + SitCode(FK)
* Cardinality Ratio:
  + Many-to-Many
* Structural Constraints:
  + Account: (N, N) An account can have one or more manager contracts.
  + ManagerContract: (N, N) Each manager contract can belong to one or more accounts.
* Correctness: An account can have multiple manager contracts, and each manager contract can correspond to multiple accounts.

(16) ContractRemittanceLink(contract, remittance)

* Attributes:
  + ContractNumber(FK)
  + RemittanceID(FK)
* Cardinality Ratio:
  + Many-to-Many
* Structural Constraints:
  + CONTRACT: (1, N) A contract can have one or more remittances associated with it.
  + REMITTANCE: (1, N) Each remittance can be linked to one or more contracts.
* Correctness: A contract can have multiple remittances, and each remittance can correspond to multiple contracts.

(17) ContractManagerLink(Associate, managerContract)

* Attributes:
  + ContractNumber(FK)
  + RemittanceID(FK)
* Cardinality Ratio:
  + Many-to-Many
* Structural Constraints:
  + CONTRACT: (1, N) A contract can have one or more remittances associated with it.
  + REMITTANCE: (1, N) Each remittance can be linked to one or more contracts.

(18) ContractManagerLink(contract, managerContract)

* Attributes:
  + ContractNumber(FK)
  + SitCode(FK)
* Cardinality Ratio:
  + Many-to-Many
* Structural Constraints:
  + CONTRACT: (1, N) A contract can be associated with one or more manager contracts.
  + MANAGERCONTRACT: (1, N) Each manager contract can be linked to one or more contracts.
* Correctness: A contract can have multiple manager contracts. Conversely, a manager contract might be associated with multiple contracts.

(19)ContractTerritoryLink(contract, territory)

* Attributes:
  + ContractNumber(FK)
  + TerritoryName(FK)
* Cardinality Ratio:
  + Many-to-Many
* Structural Constraints:
  + CONTRACT: (1, N) A contract can be associated with one or more territories.
  + TERRITORY: (1, N) Each territory can be linked to one or more contracts.
* Correctness: A contract can be valid or applicable in multiple territories. Conversely, a territory can have multiple contracts operating within it.

(20) EmployeeTerritoryLink(employee, territory)

* Attributes:
  + EmployeeID(FK)
  + TerritoryName(FK)
* Cardinality Ratio:
  + One-to-One
* Structural Constraints:
  + EMPLOYEE: (1, 1) Each employee is assigned to one territory.
  + TERRITORY: (1, 1) Each territory is managed or covered by one employee.
* Correctness: Each employee manages or covers exactly one territory, and each territory is managed or covered by exactly one employee.

(21)ContractContractPremiumLink(Contract and ContractPremium)

* Attributes:
  + PremiumCodeFK)
  + ContractNumber(FK)
* Cardinality Ratio:
  + One-to-Many
* Structural Constraints:
  + Contract: (1, N) Each contract has one or more premium
  + ContractPremium: (1, 1) Each ContractPremium is belonged to one contract
* Correctness: Each Contract has multiple benefits and each benefits has multiple premiums. Therefore each contract has multiple premiums. Conversely, a premium belong to one contract