PLIABLE PLASTIC PRODUCTION GROUP B

Table of Contents

ERD FOR EACH FIGURE

Figure 1

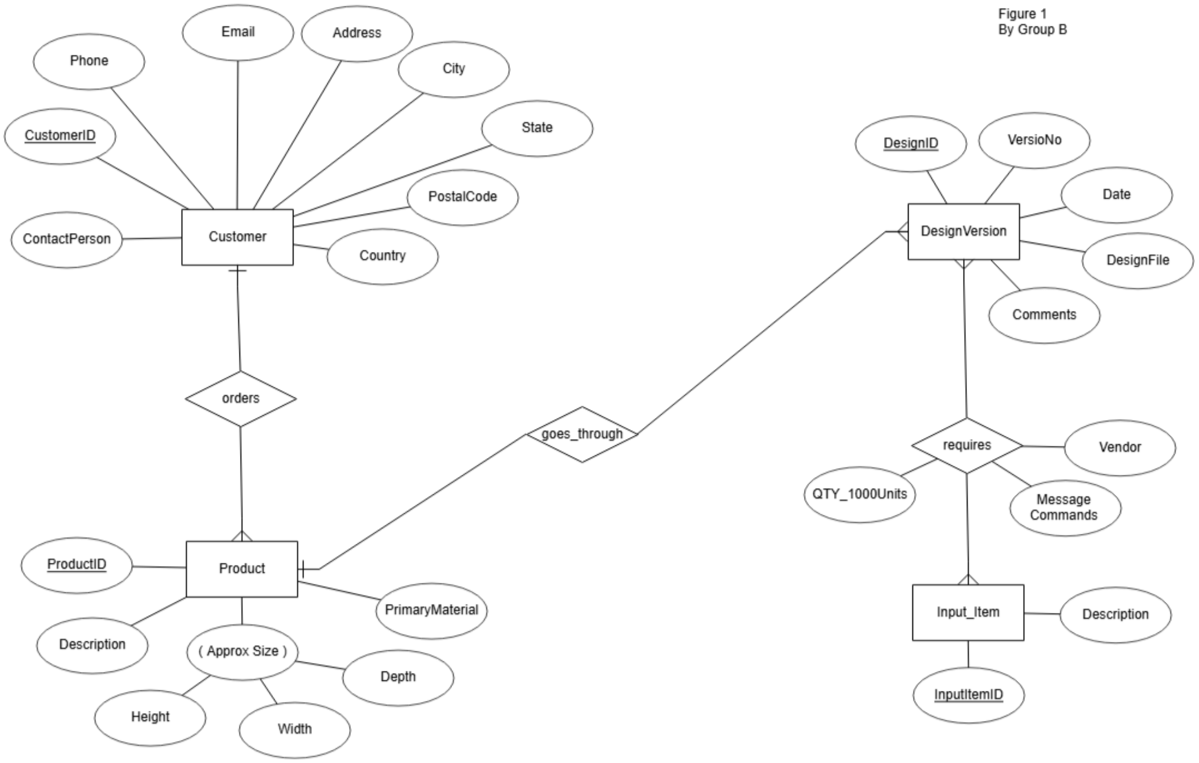


Figure 2

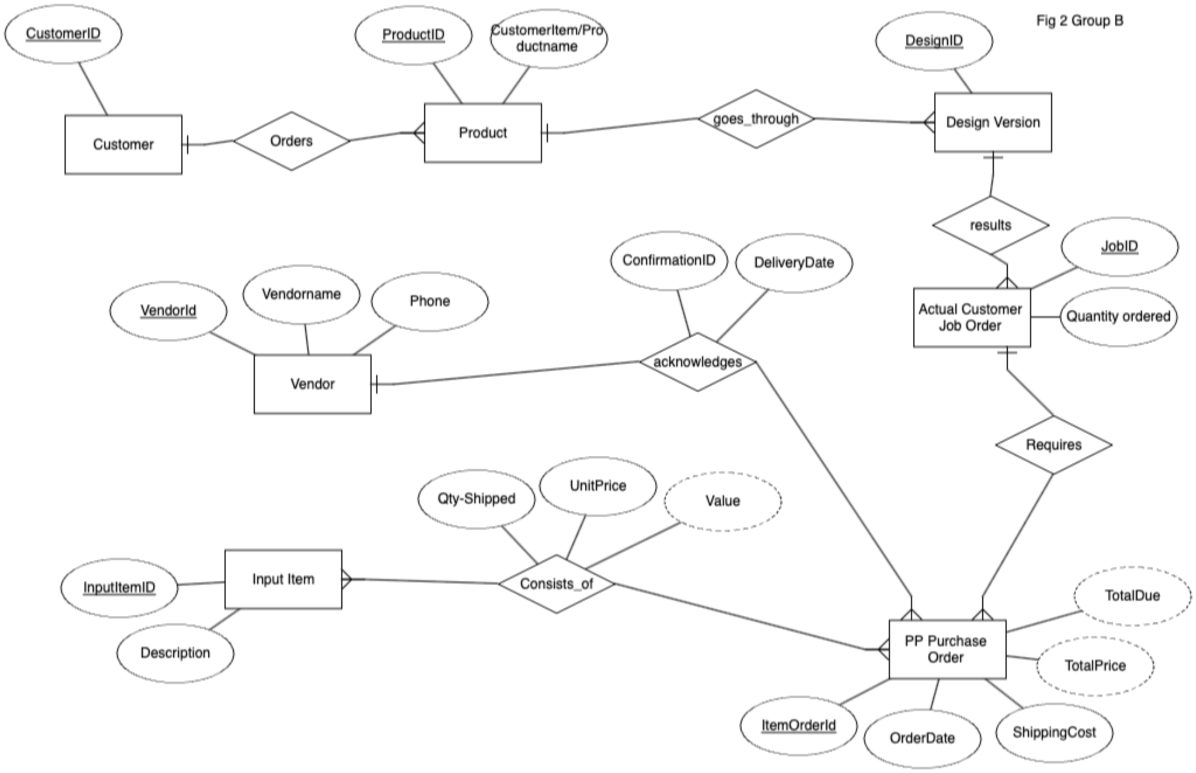


Figure 3

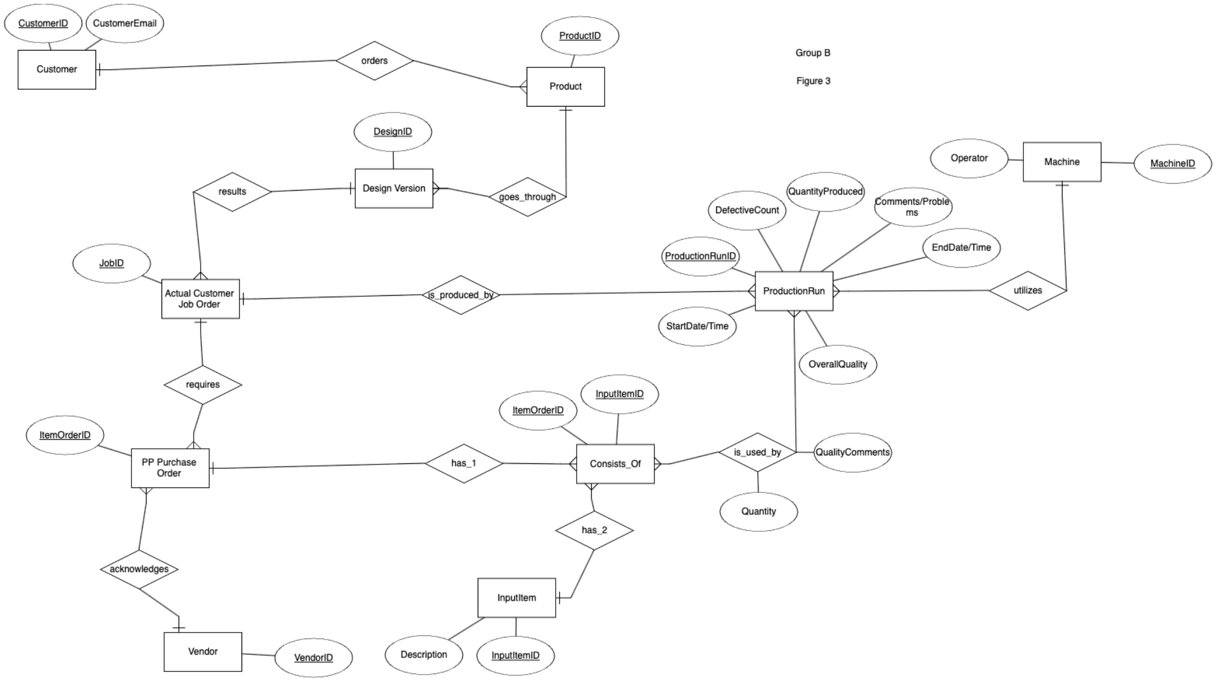
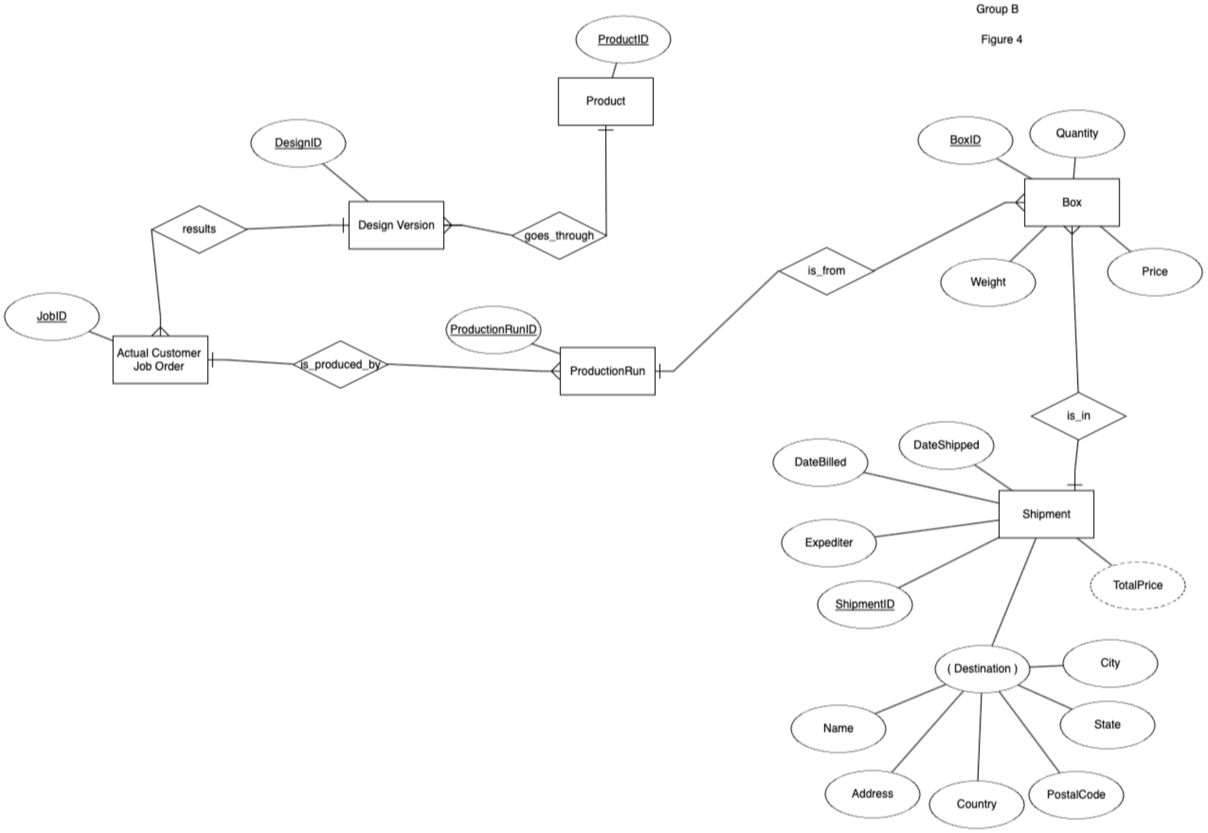


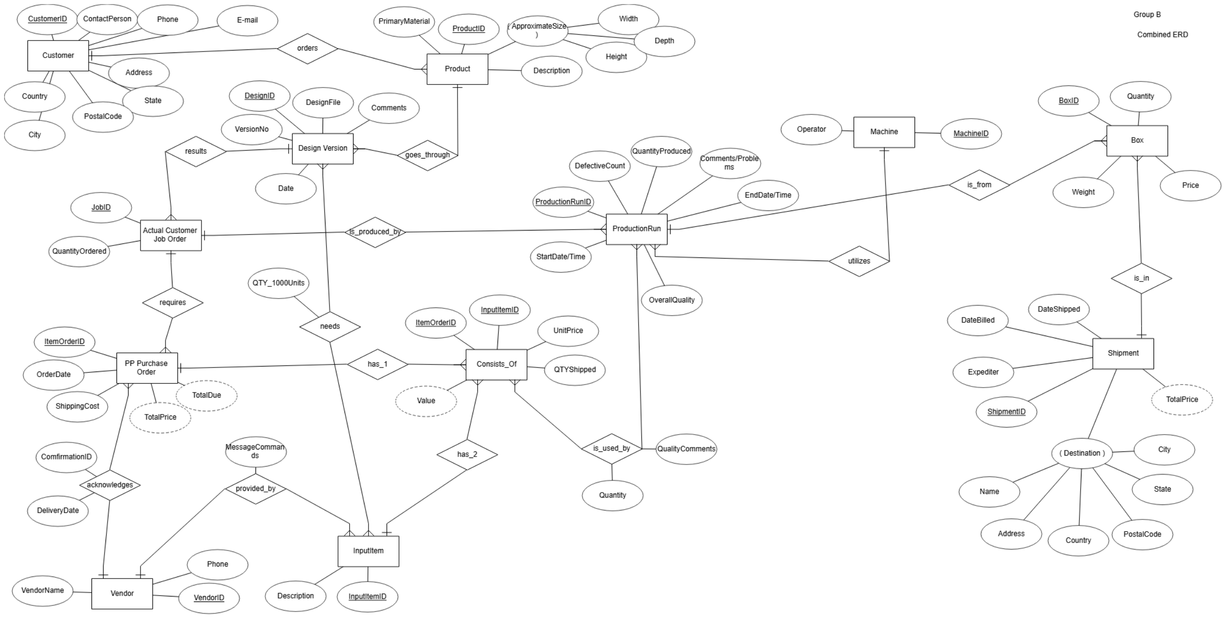
Figure 4



Group B

Figure 4

Combined ERD



Group B

Combined ERD

RELATIONAL SCHEMA DESIGN

Step 1: Each Entity becomes a table

1. Customer(CustomerID, ContactPerson, Phone, E-mail, Address, State, PostalCode, Country, City)
2. Product(ProductID, PrimaryMaterial, Width, Height, Depth, Description)
3. Machine(MachineID, Operator)
4. DesignVersion(DesignID, VersionNo, DesignFile, Comments, DesignDate)
5. ProductionRun(ProductionRunID, DefectiveCount, QuantityProduced, Comments/Problems, EndDate/Time, OverallQuality, StartDate/Time)
6. ActualCustomerJobOrder(JobID, QuantityOrdered)
7. PPPurchaseOrder(ItemOrderID, OrderDate, ShippingCost)
8. Shipment(ShipmentID, DateShipped, DateBilled, Expediter, DestinationAddress, Country, City, PostalCode, State, Name)
9. ConsistsOf(ItemOrderID, InputItemID, UnitPrice, QTYShipped)
10. Box(BoxID, Quantity, Price, BoxWeight)
11. Vendor(VendorID, VendorName, Phone)
12. InputItem(InputItemID, Description)

Step 2: Each Relationship becomes a table

1. orders(ProductID, CustomerID)
2. results(JobID, DesignID)
3. goes_through(DesignID, ProductID)
4. is_produced_by(ProductionRunID, JobID)
5. utilizes(ProductionRunID, MachineID)
6. is_from(BoxID, ProductionRunID)
7. requires(ItemOrderID, JobID)
8. needs(DesignID, InputItemID, QTY_1000Units)
9. is_in(BoxID, ShipmentID)
10. has_1(ItemOrderID, InputItemID, PPPurchaseOrder.ItemOrderID)
11. has_2(ItemOrderID, InputItemID,InputItem.InputItemID)
12. provided_by(InputItemID, VendorID, MessageCommands)
13. is_used_by(ProductionRunID, ItemOrderID, InputItemID, QualityComments, Quantity)
14. acknowledges(ItemOrderID, VendorID, ConfirmationID, DeliveryDate)

Step 3: Combine tables with exactly the same primary key column(s)

1. Product(ProductID, PrimaryMaterial, Width, Height, Depth, Description, CustomerID)
2. ActualCustomerJobOrder(JobID, QuantityOrdered, DesignID)

3. DesignVersion(DesignID, VersionNo, DesignFile, Comments, DesignDate, ProductID)
4. ProductionRun(ProductionRunID, DefectiveCount, QuantityProduced, Comments/Problems, EndDate/Time, OverallQuality, StartDate/Time, JobID, MachineID)
5. Box(BoxID, Quantity, Price, BoxWeight, ProductionRunID, ShipmentID)
6. PPPurchaseOrder(ItemOrderID, OrderDate, ShippingCost, JobID, VendorID, ConfirmationID, DeliveryDate)
7. InputItem(InputItemID, Description, VendorID, MessageCommands)
8. ConsistsOf(ItemOrderID, InputItemID, UnitPrice, QTYShipped, PPPurchaseOrder.ItemOrderID, InputItem.InputItemID)

Step 4: Final equivalent Relational Database Schema

1. Customer(CustomerID, ContactPerson, Phone, E-mail, Address, State, PostalCode, Country, City)
2. Product(ProductID, PrimaryMaterial, Width, Height, Depth, Description, CustomerID)
3. Machine(MachineID, Operator)
4. DesignVersion(DesignID, VersionNo, DesignFile, Comments, DesignDate, ProductID)
5. ProductionRun(ProductionRunID, DefectiveCount, QuantityProduced, Comments/Problems, EndDate/Time, OverallQuality, StartDate/Time, JobID, MachineID)
6. ActualCustomerJobOrder(JobID, QuantityOrdered, DesignID)
7. PPPurchaseOrder(ItemOrderID, OrderDate, ShippingCost, JobID, VendorID, ConfirmationID, DeliveryDate)
8. Shipment(ShipmentID, DateShipped, DateBilled, Expediter, DestinationAddress, Country, City, PostalCode, State, Name)
9. Box(BoxID, Quantity, Price, BoxWeight, ProductionRunID, ShipmentID)
10. Vendor(VendorID, VendorName, Phone)
11. InputItem(InputItemID, Description, VendorID, MessageCommands)
12. ConsistsOf(ItemOrderID, InputItemID, UnitPrice, QTYShipped, PPPurchaseOrder.ItemOrderID, InputItem.InputItemID)
13. needs(DesignID, InputItemID, QTY_1000Units)
14. is_used_by(ProductionRunID, ItemOrderID, InputItemID, QualityComments, Quantity)

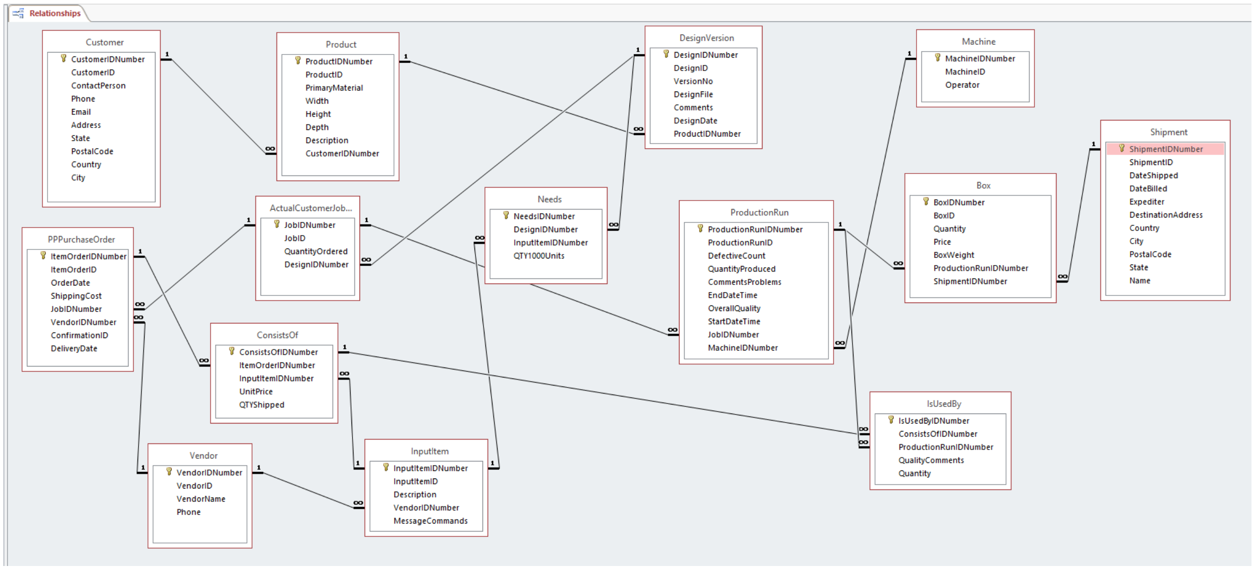
Step 5 - Access tables

1. Customer(CustomerIDNumber, CustomerID, ContactPerson, Phone, Email, Address, State, PostalCode, Country, City)
2. Product(ProductIDNumber, ProductID, PrimaryMaterial, Width, Height, Depth, Description, CustomerIDNumber)
3. Machine(MachineIDNumber, MachineID, Operator)
4. DesignVersion(DesignIDNumber, DesignID, VersionNo, DesignFile, Comments, DesignDate, ProductIDNumber)
5. ProductionRun(ProductionRunIDNumber, ProductionRunID, DefectiveCount, QuantityProduced, Comments/Problems, EndDateTime, OverallQuality, StartDateTime, JobIDNumber, MachineIDNumber)

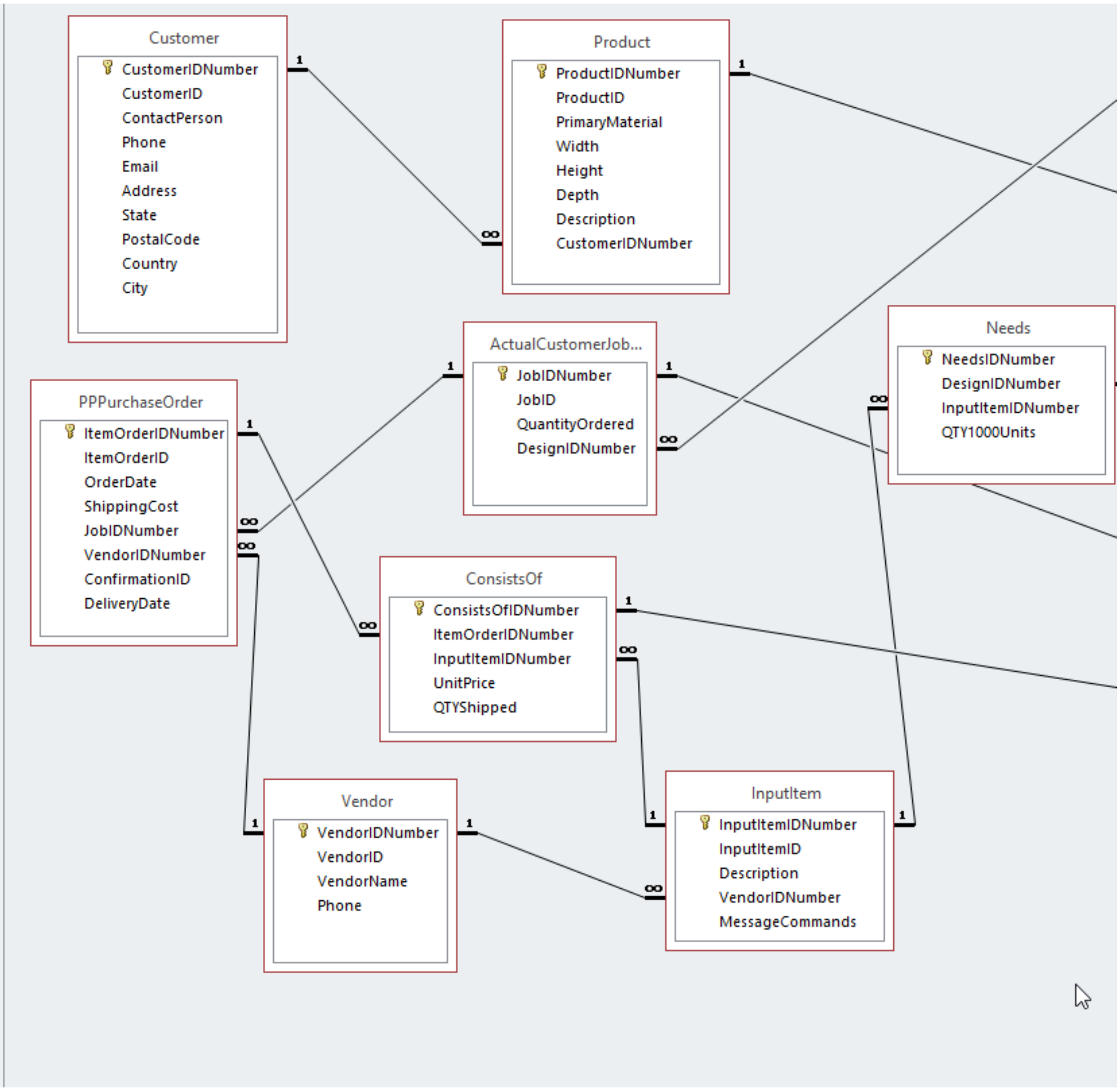
6. ActualCustomerJobOrder(JobIDNumber, JobID, QuantityOrdered, DesignIDNumber)
7. PPPurchaseOrder(ItemOrderIDNumber, ItemOrderID, OrderDate, ShippingCost, JobIDNumber, VendorIDNumber, ConfirmationID, DeliveryDate)
8. Shipment(ShipmentIDNumber, ShipmentID, DateShipped, DateBilled, Expediter, DestinationAddress, Country, City, PostalCode, State, Name)
9. Box(BoxIDNumber, BoxID, Quantity, Price, BoxWeight, ProductionRunIDNumber, ShipmentIDNumber)
10. Vendor(VendorIDNumber, VendorID, VendorName, Phone)
11. InputItem(InputItemIDNumber, InputItemID, Description, VendorIDNumber, MessageCommands)
12. Needs(NeedsIDNumber, DesignIDNumber, InputItemIDNumber, QTY1000Units)
13. ConsistsOf(ConsistsOfIDNumber, ItemOrderIDNumber, InputItemIDNumber, UnitPrice, QTYShipped)
14. IsUsedBy(IsUsedByIDNumber, ConsistsOfIDNumber, ProductionRunIDNumber, QualityComments, Quantity)

ACCESS RELATIONSHIP SCREEN

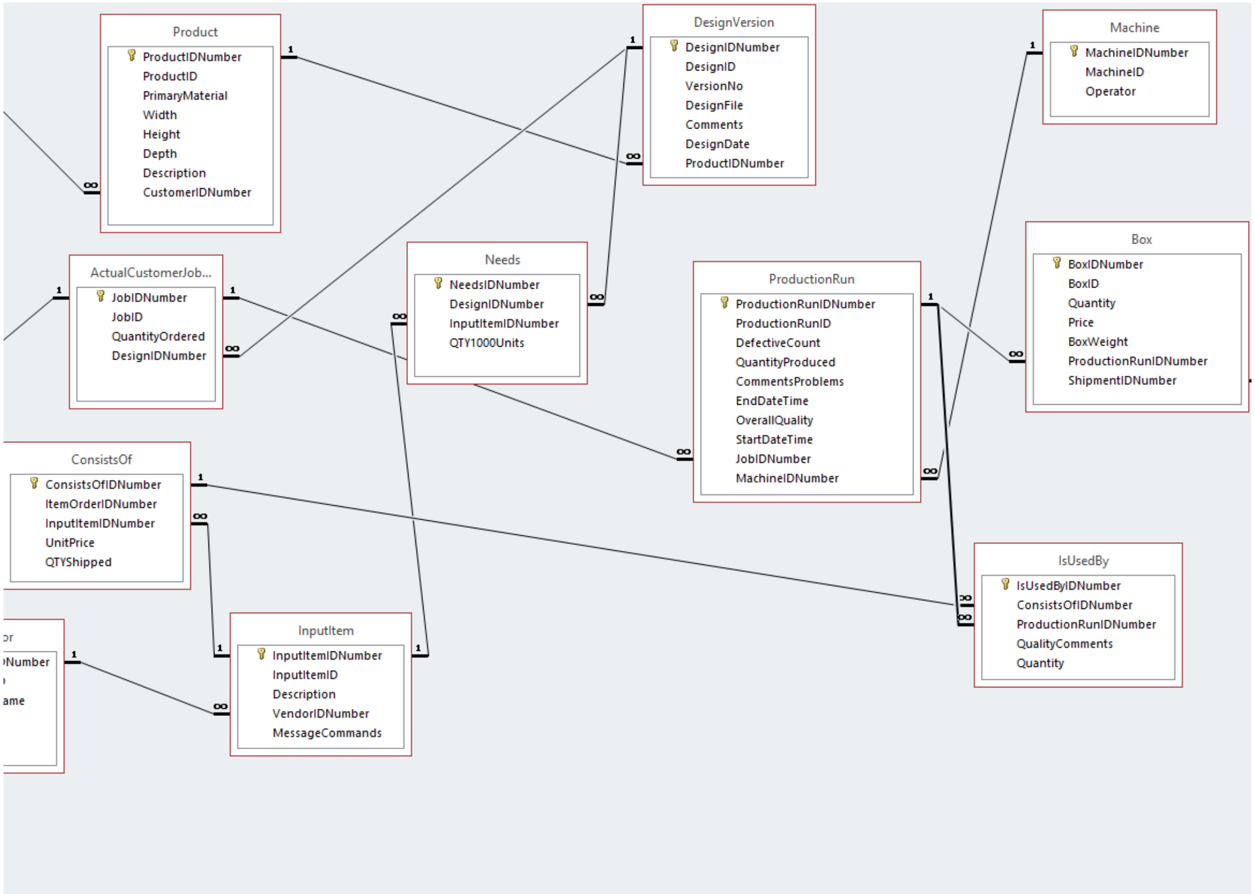
Combined Access Relationship Diagram



Left Most Part


Customer
CustomerIDNumber
CustomerID
ContactPerson
Phone
Email
Address
State
PostalCode
Country
City
Product
ProductIDNumber
ProductID
PrimaryMaterial
Width
Height
Depth
Description
CustomerIDNumber
ActualCustomerJob...
JobIDNumber
JobID
QuantityOrdered
DesignIDNumber
Needs
NeedsIDNumber
DesignIDNumber
InputItemIDNumber
QTY1000Units
PPPurchaseOrder
ItemOrderIDNumber
ItemOrderID
OrderDate
ShippingCost
JobIDNumber
VendorIDNumber
ConfirmationID
DeliveryDate
ConsistsOf
ConsistsOfIDNumber
ItemOrderIDNumber
InputItemIDNumber
UnitPrice
QTYShipped
Vendor
VendorIDNumber
VendorID
VendorName
Phone
InputItem
InputItemIDNumber
InputItemID
Description
VendorIDNumber
MessageCommands
1
∞

Middle Part



Product
- ProductIDNumber
- ProductID
- PrimaryMaterial
- Width
- Height
- Depth
- Description
- CustomerIDNumber

DesignVersion
- DesignIDNumber
- DesignID
- VersionNo
- DesignFile
- Comments
- DesignDate
- ProductIDNumber

Machine
- MachineIDNumber
- MachineID
- Operator

ActualCustomerJob...
- JobIDNumber
- JobID
- QuantityOrdered
- DesignIDNumber

Needs
- NeedsIDNumber
- DesignIDNumber
- InputItemIDNumber
- QTY1000Units

ProductionRun
- ProductionRunIDNumber
- ProductionRunID
- DefectiveCount
- QuantityProduced
- CommentsProblems
- EndDateTime
- OverallQuality
- StartDateTime
- JobIDNumber
- MachineIDNumber

Box
- BoxIDNumber
- BoxID
- Quantity
- Price
- BoxWeight
- ProductionRunIDNumber
- ShipmentIDNumber

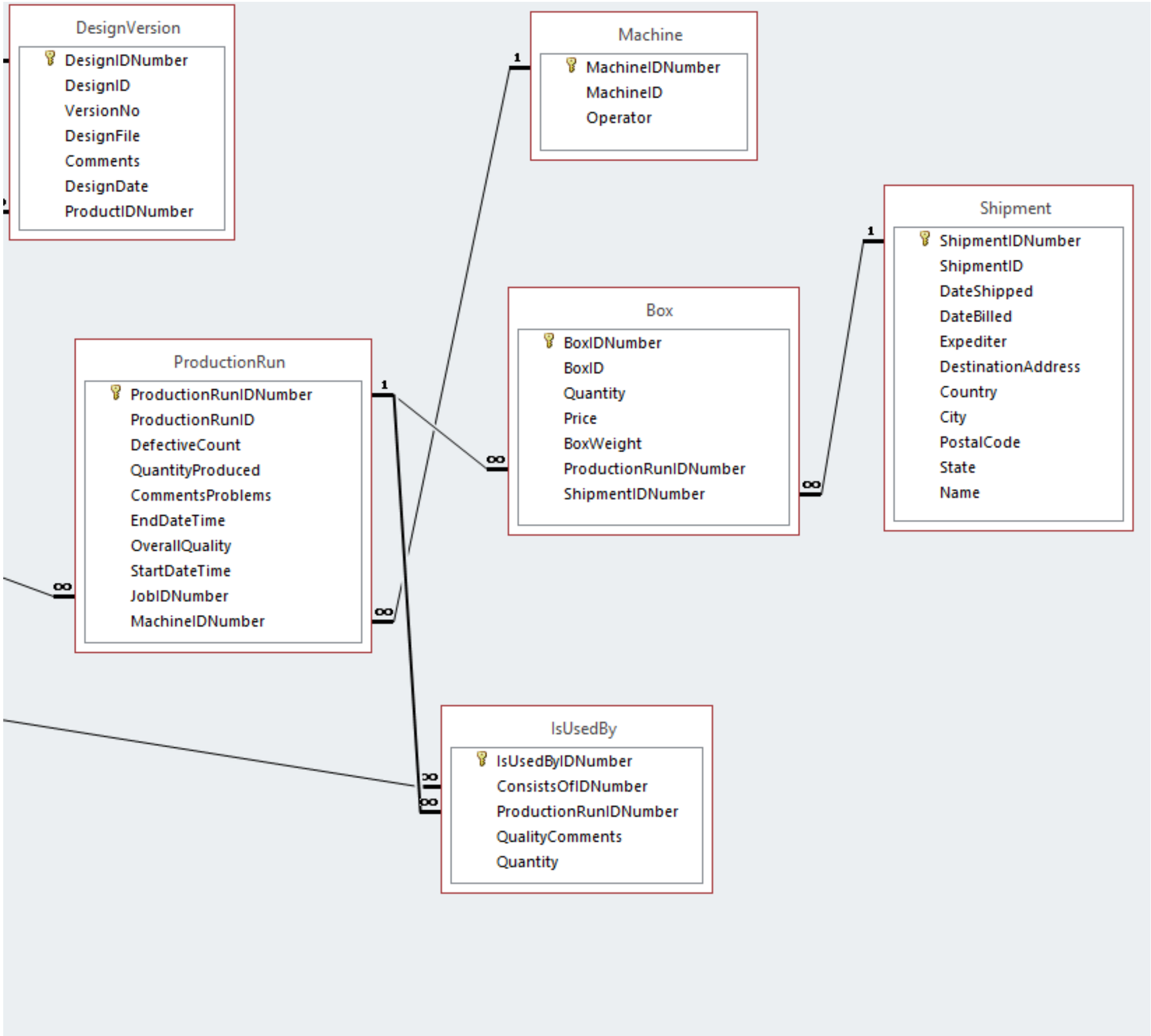
ConsistsOf
- ConsistsOfIDNumber
- ItemOrderIDNumber
- InputItemIDNumber
- UnitPrice
- QTYShipped

IsUsedBy
- IsUsedByIDNumber
- ConsistsOfIDNumber
- ProductionRunIDNumber
- QualityComments
- Quantity

or
- Number
-)
- ame

InputItem
- InputItemIDNumber
- InputItemID
- Description
- VendorIDNumber
- MessageCommands

Right Most Part



INTEGRITY CONSTRAINTS - Triet Tran

- Applied to all tables
  - Original primary keys are not null and unique
    - Test data: Insert into a Customer with no CustomerID
  - Foreign key integrity constraint
    - Test data: Insert into a Customer with no CustomerID equals CUST006
- Validation Rules added to each table
  - ProductionRun
    - EndDateTime > StartDateTime
    - Test data: Insert into a ProductionRun simply with EndDateTime > StartDateTime

- PPPurchaseOrder
    - DeliveryDate > OrderDate
- All DateTime values are non null
- Many Non Null constraints on columns we deem important to the business and for the purpose of our report demonstration
    - ActualCustomerJobOrder: QuantityOrdered
    - Box: Quantity, Price
    - ConsistsOf: UnitPrice, QTYShipped
    - Customer: Country, State
    - DesignVersion: VersionNo
    - IsUsedBy: Quantity
    - Needs: QTY 1000 Units, PrimaryMaterial
    - PPPurchaseOrder: ShippingCost, ConfirmationID (also unique)
    - Product: PrimaryMaterial
    - ProductionRun: DefectiveCount, QuantityProduced, OverallQuality
    - Shipment: Expediter

KNOWN ISSUES

- When data is not populated, the form TT-ProductionRunByCustomerBrowser and TT-TransactionSupportDesignVersion may have issues due to the dropdown box not having values. The reason is that the code always assumes available data and selects the first value of the query for the dropdown box.

SPECIAL EFFORTS

Enhancing Box Shipment Transactions - Vyshali:

- Considerable effort was dedicated to developing a transaction form that streamlines the process of recording shipment details.
- Key features include pre-filled fields for customer and order information, automated calculations for box weight and dimensions, and real-time connectivity with shipping carriers to generate tracking updates.
- Validation rules were carefully implemented to ensure accurate input of shipment dates, delivery addresses, and box contents, reducing errors and improving overall shipping efficiency.

Optimizing Pricing and Cost Calculations - Vyshali

- It is done to ensure accurate and streamlined calculations for pricing and production costs.

- The Unit Price for the Quantity Ordered in Actual Customer Job Order details is calculated from the Total Cost of the Production run.
- The Price per Unit for the production run was determined by dividing the total price of the box by the quantity of boxes produced.
- To calculate the Cost of the Production Run, the unit price was multiplied by the total quantity produced.
- Additionally, the Unit Price for the quantity ordered in the Actual Customer Job Order Details was derived from the total cost of the production run, ensuring precision in cost allocation and transparency in pricing.

Generating various and realistic data - Triet Tran:

- Various data are generated to represent different KPIs: because the KPIs are set at the beginning, the data is generated purposely to demonstrate them. This includes different quality comments for the quality KPI, different timestamps for the time to market KPI and different pricing for the revenue KPI.
- Realistic data:
    - Because one of the KPIs is time to market, it is important to generate dates for each process realistically - ex. Design Date of a design should be prior to its Production Run Start Date… All of the date information is generated in correct chronological order.
    - Items from the same Box come from the same Production Run. That means they use the same InputItems. As such, they are generated to have the same unit price.
    - The quantity ordered in Actual Customer Job Order equals the total number of Quantity Produced in its respective Production Runs and the quantity of items in each box from those Production Runs.
    - Quantity of Input Item to each Production Run in the same Actual Customer Job Order has the same ratio as the quantity produced of each Production Run (a Production Run producing more products uses proportionately more inputs).
    - Quantity of InputItem to a Production Run uses the specification in the Needs table.

Detailed Narration:

- We try to model our system as close to a real system as possible. We hope that by focusing on KPIs and adding narrations (what each form/report is used for and by whom), you can have a better understanding of how we imagine the system to be used.