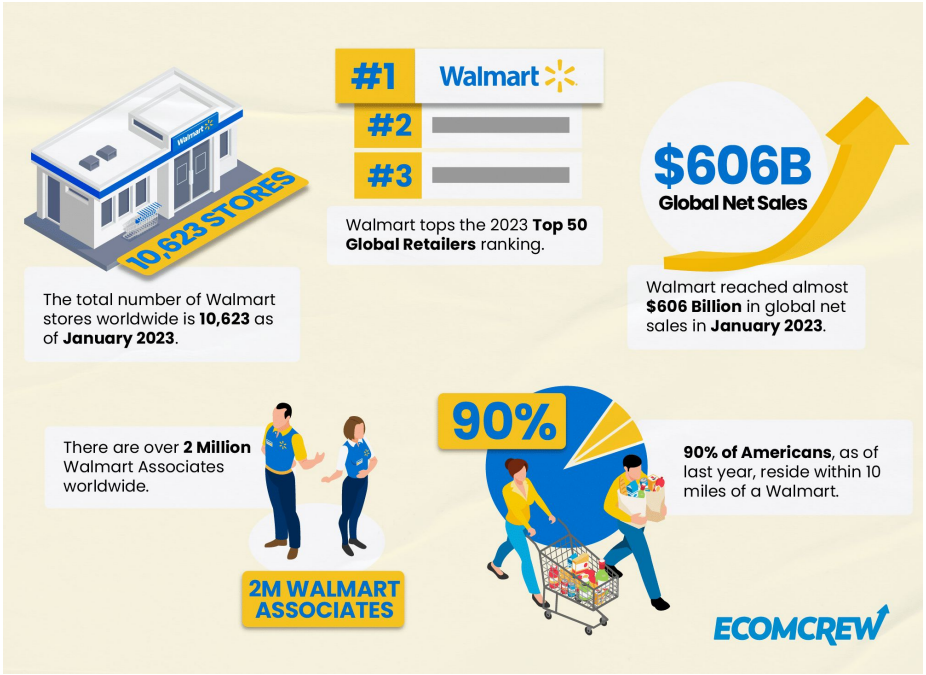
DBMS Project: Walmart

Walmart's Background

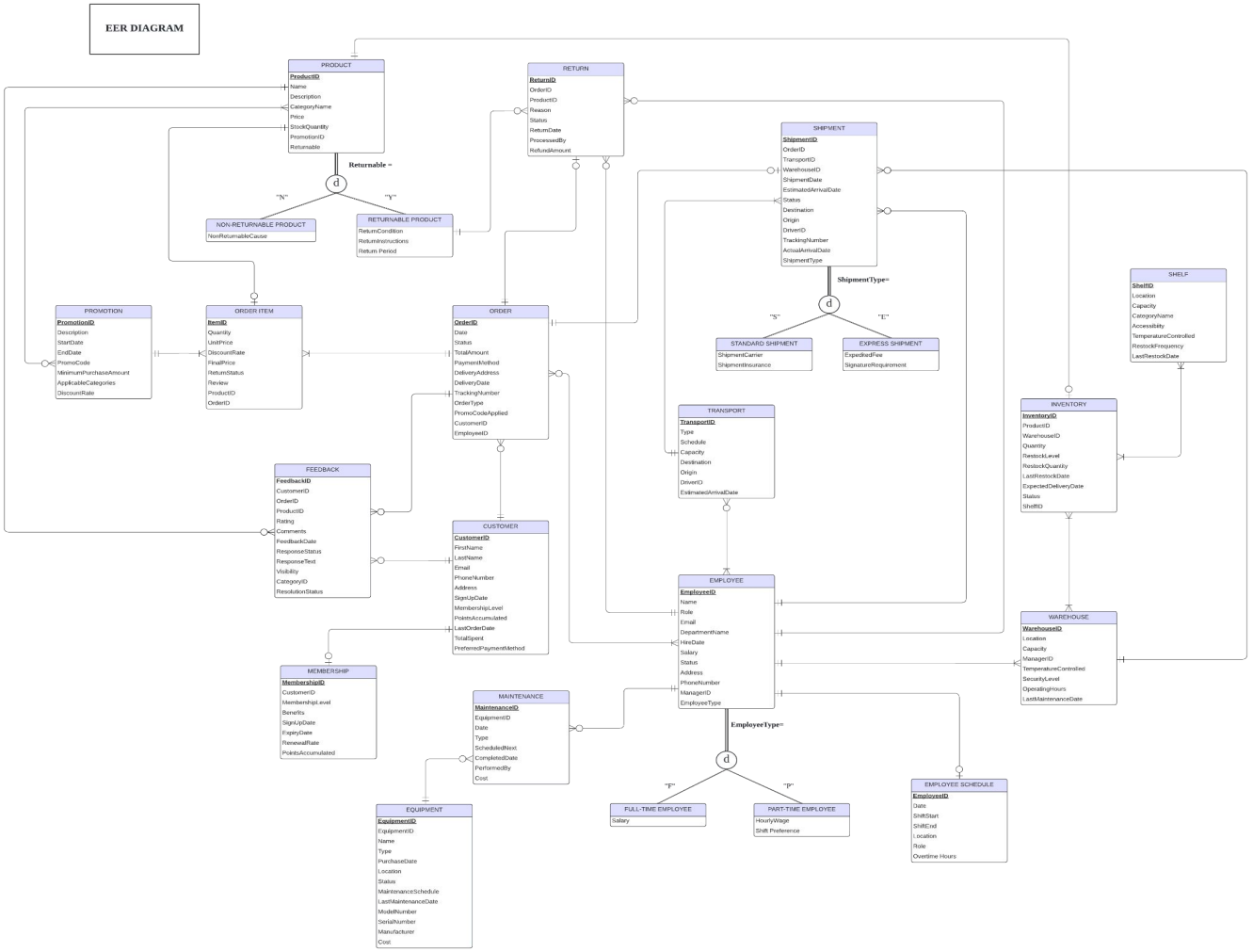
What is Walmart?	Foundations	Growth	Expansion
• Walmart is a retail corporation that hosts supercenters, department stores, and grocery stores • Extensive product offerings, competitive pricing, and convenient shopping experiences	• Founded in 1962 by Sam Walton • Emphasis on low prices and customer satisfaction • Headquarters in Bentonville, Arkansas • Locations in 52 US states	• Rapid expansion in rural areas throughout the 1970s-80s • Dependable distribution and inventory management methods that optimizes supply chain operations	• Global expansion starting in Mexico CIty in 1991 • UK, Argentina, Canada, Africa, China, Chile, India, Mexico • Diversification of products beyond groceries and household items

Walmart's Business Objectives



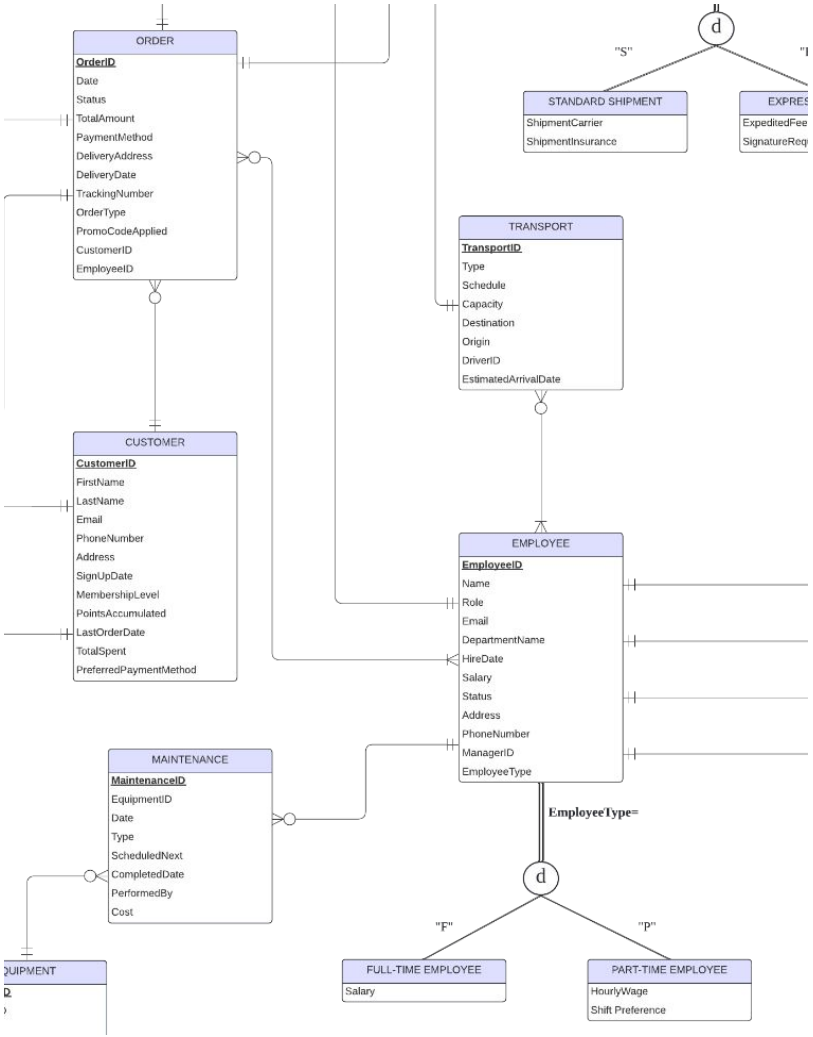
Provide competitive and affordable pricing, extensive product choices, and operational efficiency for customers.

EER Model

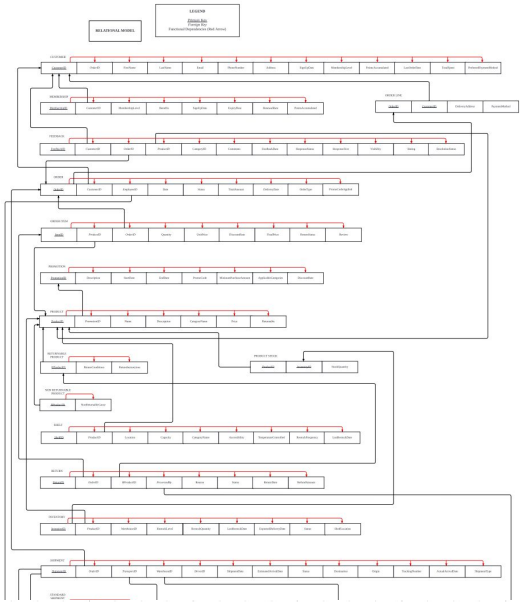

EER DIAGRAM
PRODUCT
ProductID
Name
Description
CategoryName
Price
StockQuantity
PromotionID
Returnable
Returnable =
d
"N"
"Y"
NON-RETURNABLE PRODUCT
NonReturnableCause
RETURNABLE PRODUCT
ReturnCondition
ReturnInstructions
Return Period
RETURN
ReturnID
OrderID
ProductID
Reason
Status
ReturnDate
ProcessedBy
RefundAmount
SHIPMENT
ShipmentID
OrderID
TransportID
WarehouseID
ShipmentDate
EstimatedArrivalDate
Status
Destination
Origin
DriverID
TrackingNumber
ActualArrivalDate
ShipmentType
ShipmentType=
d
"S"
"E"
STANDARD SHIPMENT
ShipmentCarrier
ShipmentInsurance
EXPRESS SHIPMENT
ExpeditedFee
SignatureRequirement
SHELF
ShelfID
Location
Capacity
CategoryName
Accessibility
TemperatureControlled
RestockFrequency
LastRestockDate
PROMOTION
PromotionID
Description
StartDate
EndDate
PromoCode
MinimumPurchaseAmount
ApplicableCategories
DiscountRate
ORDER ITEM
ItemID
Quantity
UnitPrice
DiscountRate
FinalPrice
ReturnStatus
Review
ProductID
OrderID
ORDER
OrderID
Date
Status
TotalAmount
PaymentMethod
DeliveryAddress
DeliveryDate
TrackingNumber
OrderType
PromoCodeApplied
CustomerID
EmployeeID
TRANSPORT
TransportID
Type
Schedule
Capacity
Destination
Origin
DriverID
EstimatedArrivalDate
INVENTORY
InventoryID
ProductID
WarehouseID
Quantity
RestockLevel
RestockQuantity
LastRestockDate
ExpectedDeliveryDate
Status
ShelfID
FEEDBACK
FeedbackID
CustomerID
OrderID
ProductID
Rating
Comments
FeedbackDate
ResponseStatus
ResponseText
Visibility
CategoryID
ResolutionStatus
CUSTOMER
CustomerID
FirstName
LastName
Email
PhoneNumber
Address
SignUpDate
MembershipLevel
PointsAccumulated
LastOrderDate
TotalSpent
PreferredPaymentMethod
EMPLOYEE
EmployeeID
Name
Role
Email
DepartmentName
HireDate
Salary
Status
Address
PhoneNumber
ManagerID
EmployeeType
WAREHOUSE
WarehouseID
Location
Capacity
ManagerID
TemperatureControlled
SecurityLevel
OperatingHours
LastMaintenanceDate
MEMBERSHIP
MembershipID
CustomerID
MembershipLevel
Benefits
SignUpDate
ExpiryDate
RenewalRate
PointsAccumulated
MAINTENANCE
MaintenanceID
EquipmentID
Date
Type
ScheduledNext
CompletedDate
PerformedBy
Cost
EmployeeType=
d
"F"
"P"
FULL-TIME EMPLOYEE
Salary
PART-TIME EMPLOYEE
HourlyWage
Shift Preference
EMPLOYEE SCHEDULE
EmployeeID
Date
ShiftStart
ShiftEnd
Location
Role
Overtime Hours
EQUIPMENT
EquipmentID
EquipmentID
Name
Type
PurchaseDate
Location
Status
MaintenanceSchedule
LastMaintenanceDate
ModelNumber
SerialNumber
Manufacturer
Cost

EER Model Details

- Key Entities and Relationships
  - Order and Customer
  - Order and Employee
  - Employee subtypes



Relational Model



LEGEND

Primary Key
Foreign Key
Supertype and Subtypes (Blue Arrow)
Linking Via Attributes (Dashed Line)
Functional Dependencies (Red Arrow)

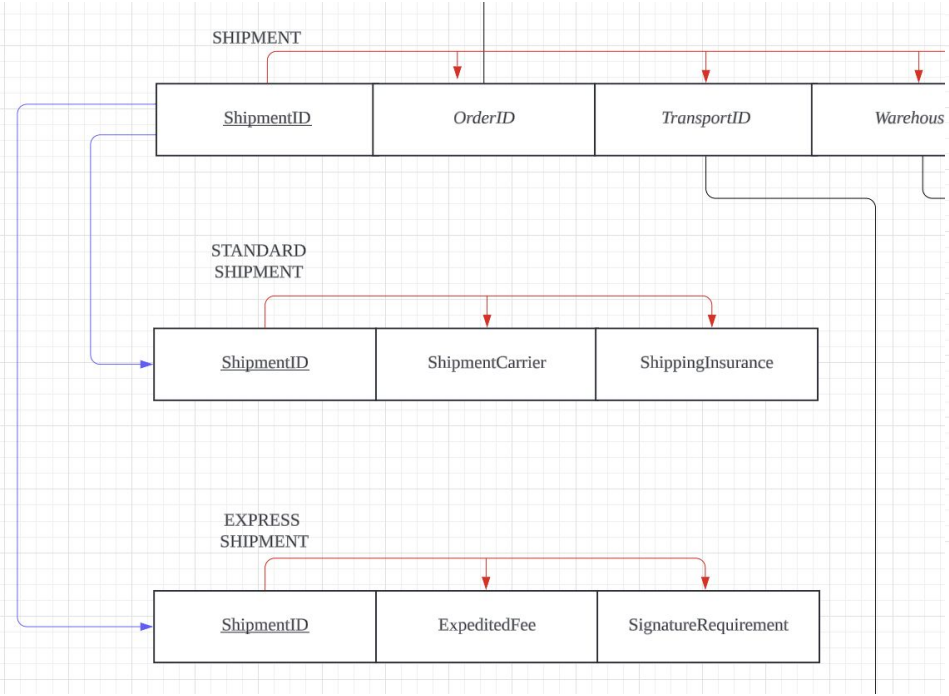
ORDER ITEM

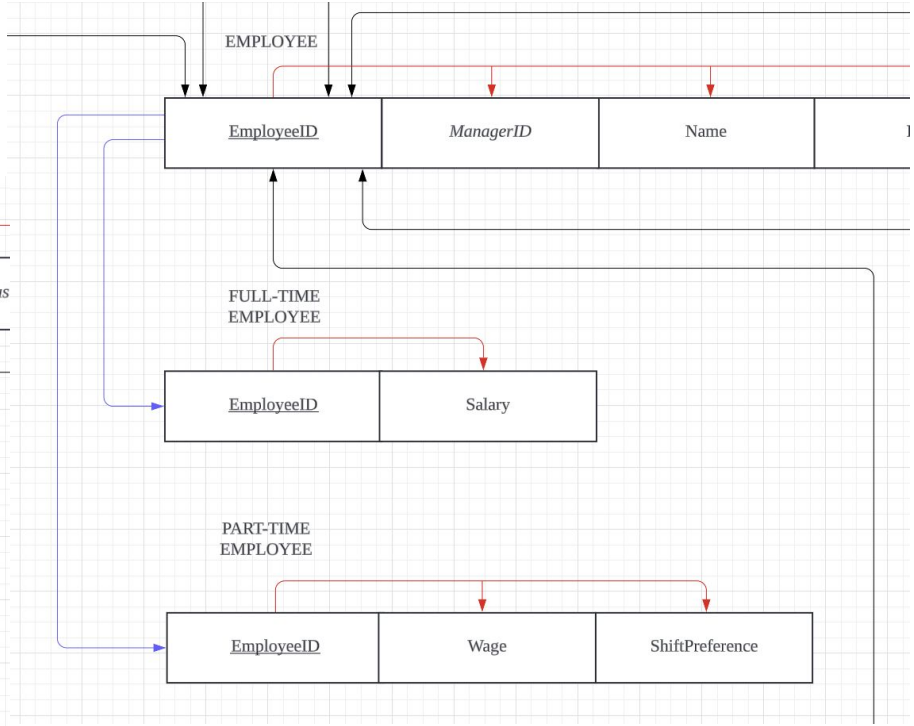
ItemID	*ProductID*	*OrderID*	Quantity	UnitPrice	DiscountRate	FinalPrice	ReturnStatus	Review

PROMOTION

PromotionID	Description	StartDate	EndDate	PromoCode	MininumPurchaseAmount	ApplicableCategories	DiscountRate

Supertypes and Subtypes

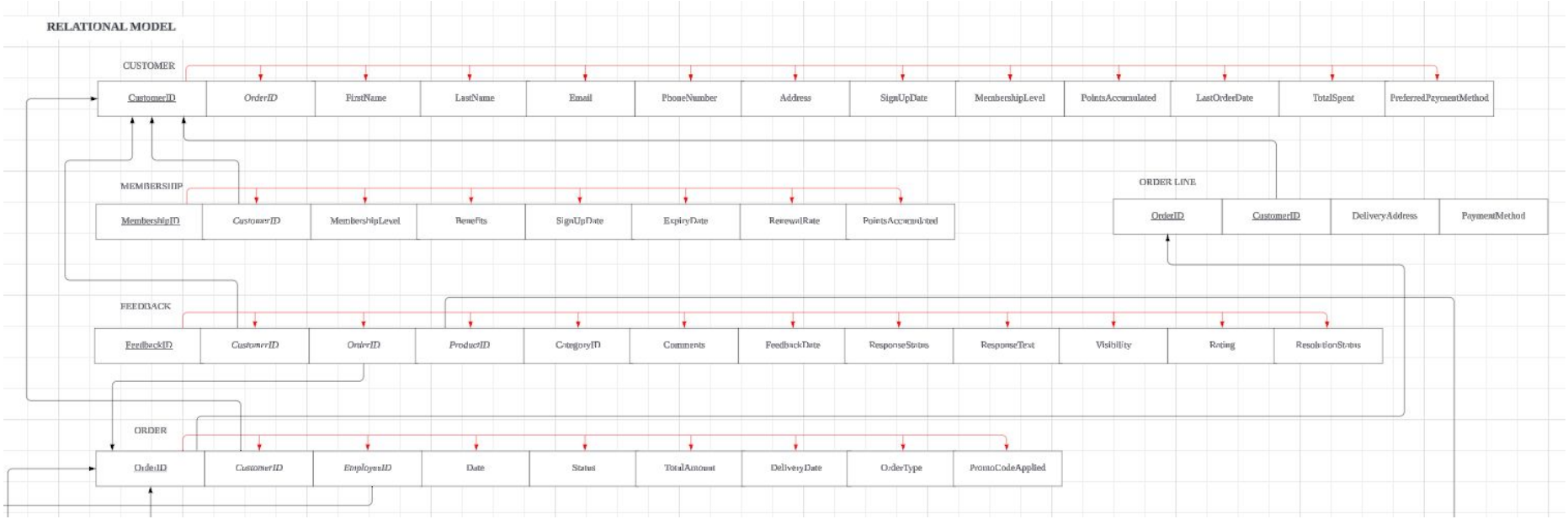




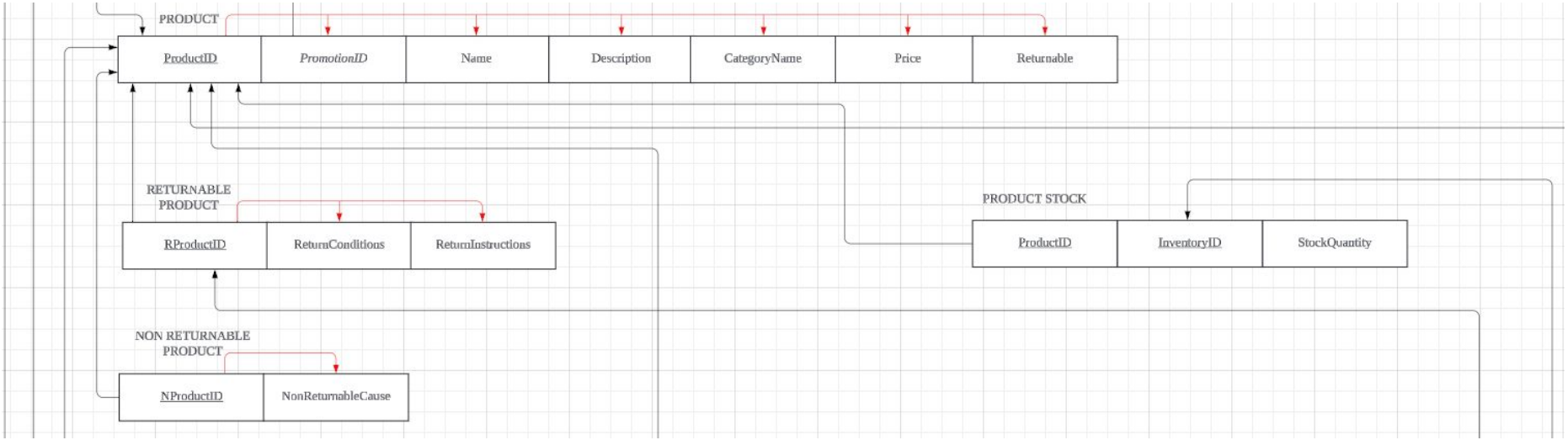
- Shipment and Employee
  - Supertypes/Subtypes

3NF Normalization

3NF applies to OrderLine Table

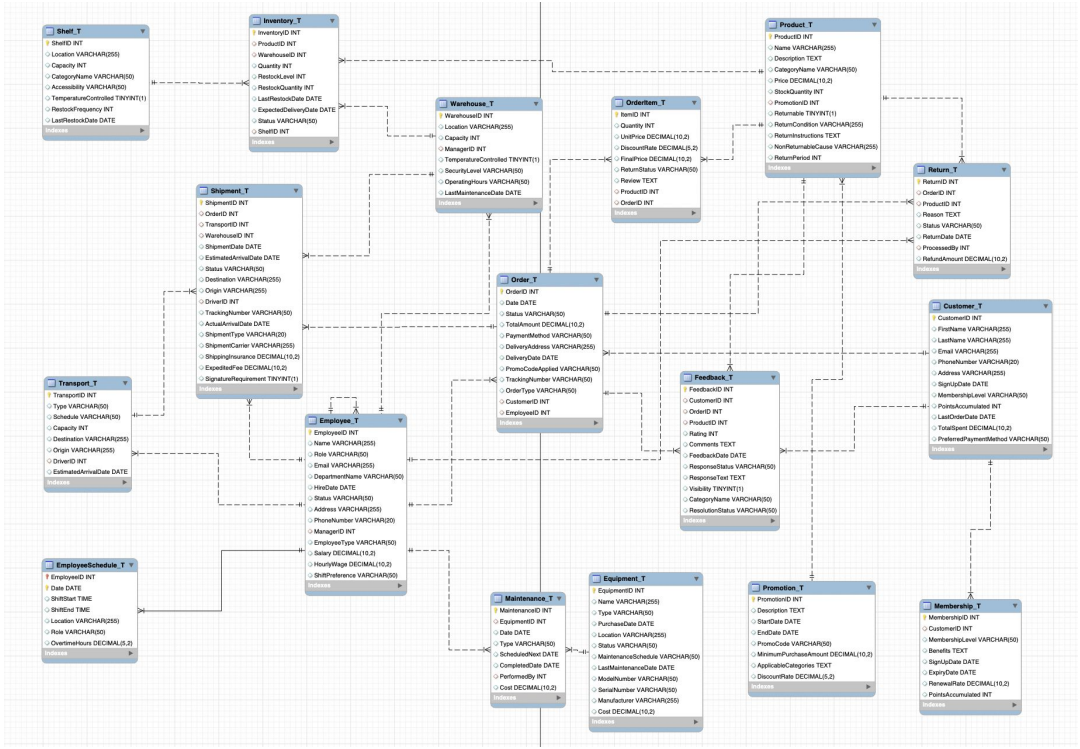


3NF Normalization Cont.



3NF applies to Product Stock Table

Sample Database



- Database covers Walmart's daily operations
- Database contains
  - 17 Entities
  - 25 Foreign Keys

Sample Data - Creating Tables

```sql
-- Employees
CREATE TABLE Employee_T (
    EmployeeID INT PRIMARY KEY,
    Name VARCHAR(255),
    Role VARCHAR(50),
    Email VARCHAR(255),
    DepartmentName VARCHAR(50),
    HireDate DATE,
    Status VARCHAR(50),
    Address VARCHAR(255),
    PhoneNumber VARCHAR(20),
    ManagerID INT, -- This will reference EmployeeID within the same table
    EmployeeType VARCHAR(50),
    Salary DECIMAL(10, 2),
    HourlyWage DECIMAL(10, 2),
    ShiftPreference VARCHAR(50),
    FOREIGN KEY (ManagerID) REFERENCES Employee_T(EmployeeID) -- This establishes the hierarchical relationship
);
```

Sample Query 1

```
SELECT CustomerID, FirstName, LastName, TotalSpent,
       RANK() OVER (ORDER BY TotalSpent DESC) AS SpendingRank
FROM Customer_T;
```

Query: Rank customers by spending in descending order.

CustomerID	FirstName	LastName	TotalSpent	SpendingRank
9	Sophia	Taylor	2200.00	1
5	Emma	Wilson	2000.00	2
4	Sarah	Brown	1500.00	3
8	William	Miller	1250.00	4
1	John	Doe	1000.00	5
10	James	Anderson	900.00	6
6	Oliver	Jones	800.00	7
2	Jane	Smith	750.00	8
7	Ava	Davis	650.00	9
3	Michael	Johnson	500.00	10

Sample Query 2

```
SELECT DISTINCT c.CustomerID, c.FirstName, c.LastName, c.PointsAccumulated
FROM Customer_T c
LEFT JOIN Order_T o ON c.CustomerID = o.CustomerID
WHERE c.PointsAccumulated >= 500
AND (o.Date < DATE_SUB(CURDATE(), INTERVAL 3 MONTH) OR o.Date IS NULL)
GROUP BY c.CustomerID, c.FirstName, c.LastName, c.PointsAccumulated;
```

Query: Retrieve customers with at least 500 points but no orders in the last 3 months.

CustomerID	FirstName	LastName	PointsAccumulated
1	John	Doe	500
4	Sarah	Brown	800
5	Emma	Wilson	1000
8	William	Miller	700
9	Sophia	Taylor	1200

Sample Query 3

```
SELECT e.EmployeeID, e.Name, IFNULL(SUM(o.TotalAmount), 0) AS TotalSales
FROM Employee_T e
LEFT JOIN Order_T o ON e.EmployeeID = o.EmployeeID
GROUP BY e.EmployeeID, e.Name
ORDER BY TotalSales DESC
LIMIT 5;
```

Query: Sort by top five employees' total sales.

EmployeeID	Name	TotalSales
13	Eva Garcia	1414.50
11	Bob Smith	521.62
10	Alice Johnson	302.22
12	Diana Lee	0.00
14	Frank Williams	0.00

Data Importance

Why is it important to have sample data and queries?

- Test functionality and reliability of the database
- Eliminate data inconsistencies and null values
- Simulate real-world business scenarios

Why is it important for Walmart to develop its database?

- Analyze purchasing behaviors
- Maximize supply chain and operational efficiency
- Maintain safe working conditions for employees
- Get feedback on how to improve the customer experience

Data Security

How can Walmart protect its database and maintain data integrity?

- Security measures
  - Data encryption
  - Security audits
  - User authentication / authorization methods
  - Ensure proper data backup and recovery
- Importance of data security
  - Protect against ransomware attacks / data breaches
  - Maintain data integrity and customer confidentiality
  - Identify and mitigate key security issues

Summary

“Save Money. Live Better.”

By offering high quality products at a low price, Walmart helps customers save money so that they can get the best out of their shopping experience.