Inventory Database Management

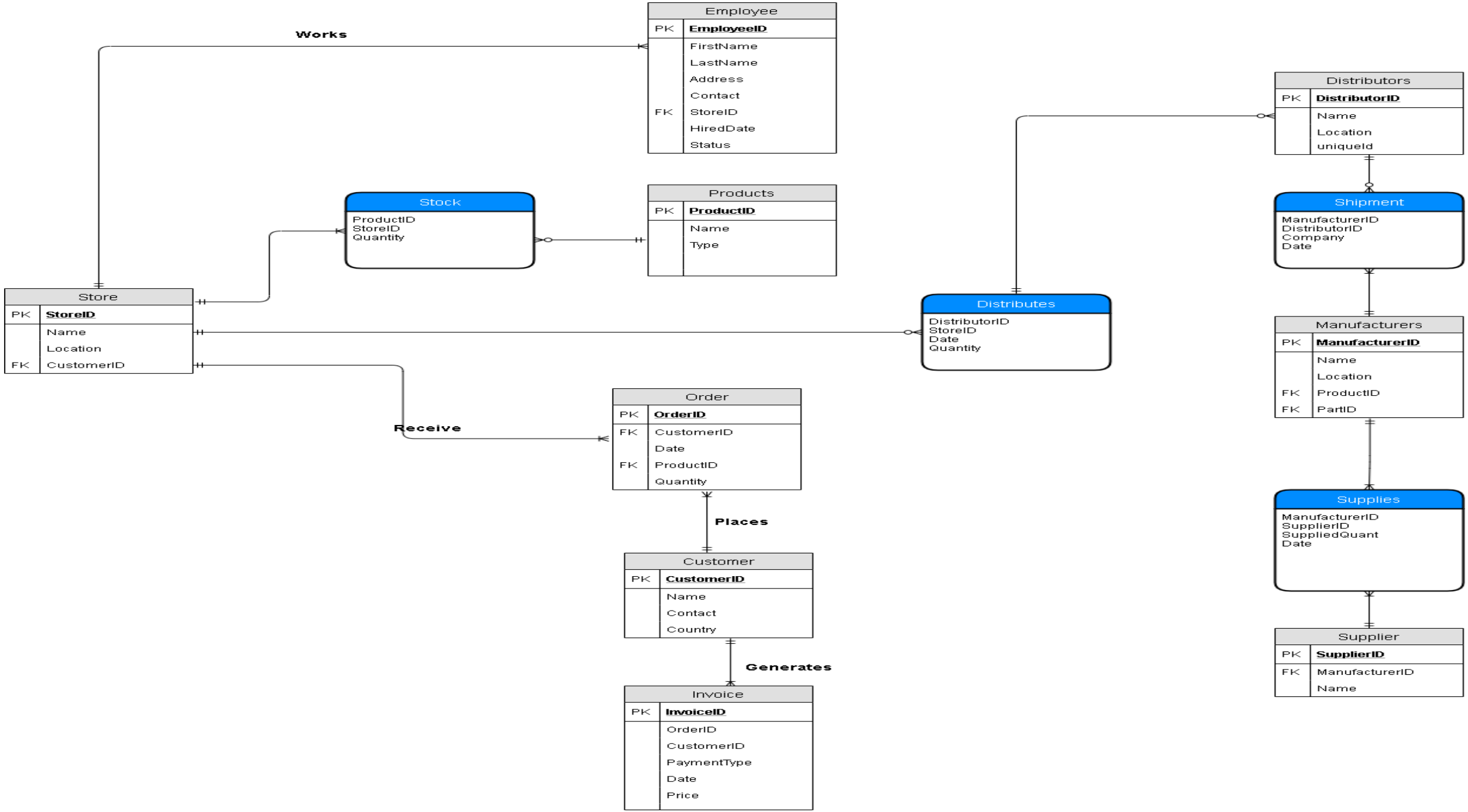
Team Blr:
Ashwin Alagappan Singaram
Sriram Bharadwaj Tirumakudal Ananthapadmanabh
Nikheel Susheel Navanale
Shravan Bhat Kulalu Shankar

Database Design Summary

- Inventory data management can be used to retrieve, store and manage information about a variety of shoes stored in global warehouses.
- This data can be used to analyze the demand, track the supply, order quantity and optimize the inventory operations.
- Using an integrated database, we can analyze the data for better planning and organization of the inventory.

Objectives

- To retrieve, store and easily access the information.
- The Database will help the store managers with efficient budgeting.
- Minimize the irregular inventory of stagnant shoe models and sizes.
- To integrate data related to finance, manufacturer, supplier and transportation for better results.
- Minimize the duplication of data and loss of data.
- The Databases helps to reflect upon the historical data to predict the future trend.


Employee
PK EmployeeID
FirstName
LastName
Address
Contact
FK StoreID
HiredDate
Status
Works
Distributors
PK DistributorID
Name
Location
uniqueId
Stock
ProductID
StoreID
Quantity
Products
PK ProductID
Name
Type
Shipment
ManufacturerID
DistributorID
Company
Date
Store
PK StoreID
Name
Location
FK CustomerID
Distributes
DistributorID
StoreID
Date
Quantity
Manufacturers
PK ManufacturerID
Name
Location
FK ProductID
FK PartID
Receive
Order
PK OrderID
FK CustomerID
Date
FK ProductID
Quantity
Supplies
ManufacturerID
SupplierID
SuppliedQuant
Date
Places
Customer
PK CustomerID
Name
Contact
Country
Supplier
PK SupplierID
FK ManufacturerID
Name
Generates
Invoice
PK InvoiceID
OrderID
CustomerID
PaymentType
Date
Price

DDL Statement – Create Table

```
CREATE TABLE [dbo].[Store](
    [StoreID] [float] NOT NULL,
    [Name] [nvarchar](255) NULL,
    [Location] [nvarchar](255) NULL,
    [CustomerID] [float] NOT NULL,
 CONSTRAINT [PK_StoreID] PRIMARY KEY CLUSTERED
(
    [StoreID]
    )
```

Stored Procedures

```sql
create proc [dbo].[MaxQuant]
AS Declare @MAXQ int
BEGIN
select @MAXQ = max(Quantity)
from [Order]
select C.CustomerID, C.[Name], O.Quantity
from Customer C Join [Order] O on C.CustomerID = O.CustomerID
where O.Quantity = @MAXQ
end
GO
exec MaxQuant
/****** Object:  StoredProcedure [dbo].[MaxQuantity]     ******/
```

100 %

Results | Messages

	CustomerID	Name	Quantity
1	373153	Linden, Mathew	5
2	682031	Jeannite, Tayana	5
3	408375	Wilber, Barry	5
4	481016	Medeiros, Jennifer	5
5	778636	Hendrickson, Trina	5
6	155262	Trang, Mei	5
7	494100	Lindsay, Leonara	5
8	719374	Tejeda, Lenora	5
9	420326	Fernandes, Nilson	5
10	553485	Barton, Nader	5

```sql
create proc [dbo].[productinfo] (@Producttype nvarchar(255))
 as begin
 select P.[Type],M.ManufacturerID, M.[Name], M.[Location]
 from Manufacturer M join Product P
 On M.ProductID = P.ProductID
 where P.[Type] = @Producttype
 end
GO
exec productinfo Boots
```

100 %

Results | Messages

	Type	ManufacturerID	Name	Location
1	Boots	483246	AMERICAN BOAT WORKS INC	PITTSBURGH
2	Boots	663408	ARES CUSTOM YACHTS	FORT LAUDERDALE
3	Boots	607844	ALPINE FIBERGLASS PRODUCTS IN	MT JULIET
4	Boots	299129	ACME ENGINEERING CO	NORTH FT MYERS
5	Boots	589342	ANCHOR GARAGE INC	EDGEWATER
6	Boots	270864	ANDREW GEMENY & SON	WALDORF
7	Boots	570454	AAD WELDING INC	Bartonville
8	Boots	726582	AQUA-CRAFT BOAT CO	CORONA
9	Boots	441190	SNEAKS UNLIMITED INC	PELION
10	Boots	694862	AMERICAN INTL INFLTBLE	LONG BEACH

```sql
CREATE proc [dbo].[info] (@Name nvarchar(255))
AS BEGIN
select I.CustomerID, C.[Name], I.Payment_type, I.Price
from Invoice I JOIN Customer C
ON I.CustomerID = C.CustomerID
where C.[Name] = @Name
END
GO
exec dbo.[info] Morgan
```

100 %

Results | Messages

	CustomerID	Name	Payment_type	Price
1	12348	Morgan	NULL	159

Function

```
CREATE FUNCTION [dbo].[Experience] (@EMpId int)
RETURNS int
as begin
Declare @g float
Declare @Num date
DECLARE @date float
Select @Num=HiredDate from Employee where EmpId=@EMpI
select @date=(convert(char(8), getdate(), 112)/10000)
Select @g=(Convert(Char(8),@Num,112)/10000)
return ((@date-@g))
END
GO
select [dbo].[Experience](463671) as Experience
```

100 %

Results | Messages

	Experience
1	6

Views

```sql
create view [dbo].[product_view]
AS
SELECT P.ProductID, P.[Name] AS Product_Name, P.[Type], M.[Name] AS Manufacturer_Name, S.Comp
FROM Product P JOIN Manufacturer M
ON P.ProductID = M.ProductID
JOIN Shipment S
ON
M.ManufacturerID = S.ManufacturerID
GO
select * from product_view
```

100 %

Results | Messages

	ProductID	Product_Name	Type	Manufacturer_Name	Company	Date
1	534509	Northface	Loafers	FOUR WINNS INC	MSC Shipping Company	2018-06-16 00:00:00.000
2	556058	Northface	Boots	LA VIDA YACHT SALES	Maersk	2018-05-01 00:00:00.000
3	346133	Columbia	Flip-Flop	DER GROUP	China Ocean Shipping Company	2018-06-23 00:00:00.000
4	427400	Under Armor	Flip-Flop	ALL CRAFT MARINE LLC	CMA CGM	2018-06-24 00:00:00.000
5	282147	Under Armor	Sandals	AARON HINES DESIGNS	One Ocean	2018-04-04 00:00:00.000
6	215181	Crocs	Sandals	WEST COAST ALUMINUM OUTFITRS	MSC Shipping Company	2018-04-29 00:00:00.000
7	303776	Puma	Flip-Flop	ALA FIBERGLASS	Maersk	2018-04-09 00:00:00.000
8	205647	Crocs	Sneakers	ACADIA CANOES	China Ocean Shipping Company	2018-06-21 00:00:00.000
9	208934	Puma	Flip-Flop	RENEGADE MARINE INC	CMA CGM	2018-04-22 00:00:00.000
10	561286	Puma	Loafers	LIBERATOR BOATS	One Ocean	2018-06-17 00:00:00.000

```sql
/****** Object:  View [dbo].[store_view]    ******/

create view [dbo].[Store_view]
AS
SELECT C.CustomerID, C.[Name], I.Price, O.Quantity, O.OrderID
FROM Invoice I JOIN [Order] O
ON O.OrderID = I.OrderID
JOIN Customer C
ON C.CustomerID = O.CustomerID
GO
Select * from Store_view
```

100 %

Results | Messages

	CustomerID	Name	Price	Quantity	OrderID
1	13671	Huynh, Ming	179	1	196668
2	19670	Champaigne, Brian	139	2	788643
3	28540	Owad, Clinton	151.2	2	227068
4	28608	Pelech, Emil	199	2	277548
5	30256	LeBlanc, Brandon R	139.3	1	391834
6	31867	Pearson, Randall	199	1	288229
7	32737	Veera, Abdellah	90.3	2	383194
8	33804	Guilianno, Mike	139	5	604376
9	39649	Stanley, David	118.3	4	544424
10	41344	Brown, Mia	159	1	628786

Trigger

```
/****** Create trigger     ******/
Create TRIGGER History
   ON  dbo.Employee
  FOR INSERT
AS
BEGIN
insert into dbo.Empaudit
([EmpID],[First Name],[Last Name],[Address],[Contact],[StoreID],[Status],[HiredDate],[Action],[ActionDate])
Select I.[EmpID], I.[First Name], I.[Last Name], I.[Address], I.[Contact], I.[StoreID], I.[Status], I.[HiredDate], 'I', getdate()

from inserted I
where (I.[Contact] IS NULL OR I.[Contact] = 0)
End

Insert into Employee values (654321,'Don','Wu','PO BOX 52972',NULL,88,'Full-Time','7/7/2014')
Select * from Empaudit
```

100 %

Results | Messages

	EmpID	First Name	Last Name	Address	Contact	StoreID	Status	HiredDate	Action	ActionDate
1	490471	John	Shankar	137 Park Dr	NULL	71	Part-Time	0	I	2020-04-07 19:02:01.267

Conclusion

- With the help of the database, we can help the stores increase the chance of customers finding their favorite shoes at the store.
- Historical sales data from the stores will be directed to the database for better service. It helps to integrate the supply chain network from the manufacturer to the end customer.
- This can also be used to store information to adapt the supply chain strategies of the manufacturers, suppliers and the logistics personnel.