

DBDD Project Phansekar, Ojas Section 06 NUID:001826636

Telecom Service Provider Organization Database

Project Definition:

Telecom Service Providers have lots and lots of customers to handle daily and also need to give service to the new customers. In order to have the records of the customers and also the employees who are working for them, there needs to be a database created for them. This is the reason why, I decided to choose this topic as a project for Database Management and Database Design.

Data Included in my database

I have maintained a database of all the customers and employees of a telecom service provider organization. Database also has the information regarding the sim number, phone number, call records, call duration. It also provides information regarding the plans on which the customer is currently active on, how much data he/she has used. Medium of information which the customer demands is through salesperson access. Salesperson could also be able to help the customer with the status of his current order and also with the tracking information. Another access is given to system admin who can access all the data in the organization starting from employee data to his work location, salary, department.

Relationships:

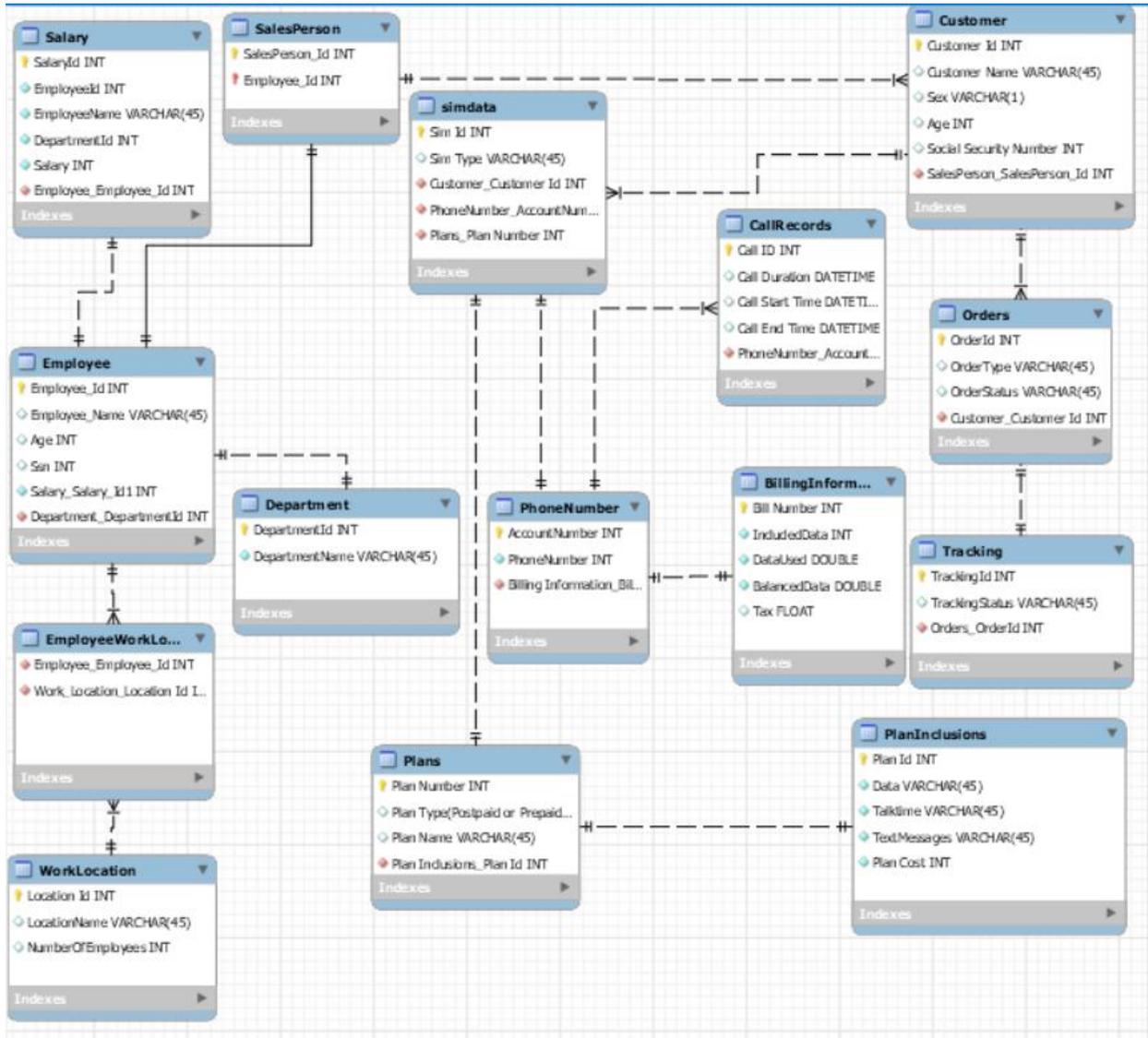
Customer can have one or more sims, but one sim can be owned by only one person. When a customer enters a shop, he approaches a salesperson, this means at a particular time a customer can approach a salesperson at a time. Salesperson has the access to the plans of the customer on his computer system. Salesperson has access to one or more than one postpaid and prepaid plans also he can help the customer with the current usage, balanced data and help him find the billing information. Customer has an SSN, age, address, name associated with his account. Customer can be on one or more plans depending on whether the customer is prepaid or postpaid. A particular plan can be used by one or more customers.

A particular customer can have a single bill and a bill is generated only for one person. Customer's usage is recorded showing the salesperson the current usage, remaining balance, usage.

An employee can work at multiple work location and a work location has one or more employees.

An employee can work in a department and a department has different slots of the salaries in each of them. Salesperson is able to view the orders made by the customer and help him find the order and tracking status.

ER Diagram:



Screenshots of the Trigger created:

```
Delimiter //
Create Trigger assignDepartments after insert on employee
for each row
begin
insert into salary values (NEW.employeeid,New.Employee_Name,NEW.departmentid,NEW.Salary);
end//
Delimiter ;
```

```
insert into employee values(27,'Devdip Sen','654787654',56,5,10000);
insert into employee values(28,'Alpana Sharan','987787654',45,3,2500);
insert into employee values(29,'Priyanka Singh','238787654',43,3,2500);
insert into employee values(30,'Ranjani Iyer','777787654',34,2,5000);
insert into employee values(31,'Amlan Bhuyan','555787654',23,4,7500);
insert into employee values(32,'Manoj Prabhakar','444787654',21,1,1000);
insert into employee values(33,'Raj Phadke','999787654',56,5,10000);
insert into employee values(34,'Priya Yadav','228787654',33,1,1000);
insert into employee values(35,'Sayali Joshi','158787654',22,4,7500);
insert into employee values(36,'Pranav Patil','658787654',25,5,10000);
insert into employee values(37,'Rohit Patil','222787654',45,3,2500);
insert into employee values(38,'Swanand Sapre','111787654',65,5,10000);
```

select * from salary;

EmployeeId	EmployeeName	DepartmentId	Salary
27	Devdip Sen	5	10000
28	Alpana Sharan	3	2500
29	Priyanka Singh	3	2500
30	Ranjani Iyer	2	5000
31	Amlan Bhuvan	4	7500
32	Manoj Prabhakar	1	1000
33	Raj Phadke	5	10000
34	Priya Yadav	1	1000
35	Sayali Joshi	4	7500
36	Pranav Patil	5	10000
37	Rohit Patil	3	2500
38	Swanand Sapre	5	10000

select * from employee;

EmployeeId	Employee_Name	SSN	Age	DepartmentId	Salary
27	Devdip Sen	458787654	56	5	10000
28	Albana Sharan	987787654	45	3	2500
29	Privanka Singh	238787654	43	3	2500
30	Raniani Iver	777787654	34	2	5000
31	Amlan Bhuvan	555787654	23	4	7500
32	Manoi Prabhakar	444787654	21	1	1000
33	Rai Phadke	999787654	56	5	10000
34	Priva Yadav	228787654	33	1	1000
35	Savali Joshi	158787654	22	4	7500
36	Pranav Patil	658787654	25	5	10000
37	Rohit Patil	222787654	45	3	2500
38	Swanand Sapre	111787654	65	5	10000

Screenshot of stored procedure:

```
delimiter //  
create procedure enterBillingInformation(IN includedData int, IN dataUsed int, IN tax double)  
begin  
insert into billingInformation (IncludedData,DataUsed,BalancedData,Tax)values(includedData, dataUsed,includedData-dataUsed,tax);  
end //
```

call enterBillingInformation(2000,1800,200);

select * from billingInformation;

BillNumber	IncludedData	DataUsed	BalancedData	Tax
1	50	40	10	15.1
2	100	60	40	20.3
3	200	100	100	30.3
4	500	200	300	40.3
5	500	100	400	50
6	1000	100	900	70
7	800	300	500	60
8	1200	400	800	100
9	1500	400	1100	100
10	1500	600	900	300
11	1500	800	700	180
12	2000	900	1100	180
13	2000	500	1500	170
14	2000	500	1500	200
15	2000	1800	200	200

Screenshot of Table created using timediff() for call records table

```
insert into callrecords values(1,'12:20:20','12:21:20',timediff('12:21:20','12:20:20'),10);
insert into callrecords values(2,'11:23:24','15:40:30',timediff('15:40:30','11:23:24'),12);
insert into callrecords values(3,'08:30:10','08:32:20',timediff('08:32:20','08:30:10'),11);
insert into callrecords values(4,'21:45:30','21:50:34',timediff('21:50:34','21:45:30'),14);
insert into callrecords values(5,'12:32:21','12:34:20',timediff('12:34:20','12:32:21'),10);
insert into callrecords values(6,'19:10:13','19:55:20',timediff('19:55:20','19:10:13'),10);
insert into callrecords values(7,'20:30:25','21:21:20',timediff('21:21:20','20:30:25'),10);
insert into callrecords values(8,'13:40:56','14:34:20',timediff('14:34:20','13:40:56'),10);
insert into callrecords values(9,'16:20:20','18:21:20',timediff('18:21:20','16:20:20'),10);
insert into callrecords values(10,'17:42:59','18:21:20',timediff('18:21:20','17:42:59'),11);
insert into callrecords values(11,'22:48:13','23:21:13',timediff('23:21:13','22:48:13'),11);
insert into callrecords values(12,'23:12:20','23:33:45',timediff('23:33:45','23:12:20'),11);
insert into callrecords values(13,'17:15:20','17:50:20',timediff('17:50:20','17:15:20'),11);
insert into callrecords values(14,'16:35:13','16:59:59',timediff('16:59:59','16:35:13'),10);
insert into callrecords values(15,'14:36:54','14:40:23',timediff('14:40:23','14:36:54'),12);
insert into callrecords values(16,'11:52:56','12:21:20',timediff('12:21:20','11:52:56'),12);
insert into callrecords values(17,'09:31:43','10:21:28',timediff('10:21:28','09:31:43'),13);
insert into callrecords values(18,'06:14:54','06:57:56',timediff('06:57:56','06:14:54'),13);
insert into callrecords values(19,'22:48:13','23:21:13',timediff('23:21:13','22:48:13'),19);
insert into callrecords values(20,'14:51:54','14:59:23',timediff('14:59:23','14:51:54'),17);
```

Select * from callrecords;

CallId	CallStartTime	CallEndTime	CallDuration	CallAccountNumber
1	12:20:20.000000	12:21:20.000000	00:01:00.000000	10
2	11:23:24.000000	15:40:30.000000	04:17:06.000000	10
3	08:30:10.000000	08:32:20.000000	00:02:10.000000	11
4	21:45:30.000000	21:50:34.000000	00:05:04.000000	14
5	12:32:21.000000	12:34:20.000000	00:01:59.000000	10
6	19:10:13.000000	19:55:20.000000	00:45:07.000000	10
7	20:30:25.000000	21:21:20.000000	00:50:55.000000	10
8	13:40:56.000000	14:34:20.000000	00:53:24.000000	10
9	16:20:20.000000	18:21:20.000000	02:01:00.000000	10
10	17:42:59.000000	18:21:20.000000	00:38:21.000000	11
12	23:12:20.000000	23:33:45.000000	00:21:25.000000	11
13	17:15:20.000000	17:50:20.000000	00:35:00.000000	11
14	16:35:13.000000	16:59:59.000000	00:24:46.000000	10
15	14:36:54.000000	14:40:23.000000	00:03:29.000000	12
16	11:52:56.000000	12:21:20.000000	00:28:24.000000	12
17	09:31:43.000000	10:21:28.000000	00:49:45.000000	13
18	06:14:54.000000	06:57:56.000000	00:43:02.000000	13
19	22:48:13.000000	23:21:13.000000	00:33:00.000000	19
20	14:51:54.000000	14:59:23.000000	00:07:29.000000	17

Screenshot of Views

- 1) View to display the call records of a particular customer, which would help the customer and salesperson both save time finding the records.

```
create view displayCallRecords as
select * from callrecords where callaccountnumber =
( select accountnumber from phonenumber where accountnumber =
(select simaccountnumber from simdata where simcustomerid =
(select customerid from customer where customerid = 1)));
```

```
select * from displayCallRecords;
```

CallId	CallStartTime	CallEndTime	CallDuration	CallAccountNumber
4	21:45:30.000000	21:50:34.000000	00:05:04.000000	14

- 2) View to display the phone number associated with a particular phone number, which helps the salesperson to help the customer in knowing the phone number if the customer forgets his phone number due to old age.

```
create view displayPhoneNumber as
select CustomerId, CustomerName, phonenumber.PhoneNumber from customer
inner join simdata
on customer.customerid = simdata.simcustomerid
inner join phonenumber
on simdata.simaccountnumber = phonenumber.accountnumber;
```

```
select * from displayPhoneNumber;
```

CustomerId	CustomerName	simnumber	PhoneNumber
2	Harsh Shah	1234567890123460	1745889839
16	Vitavshree Updili	1234567890123461	1235465768
14	Shruti Mehta	1234567890123462	1345267859
1	Jishnu Vasudevan	1234567890123463	1876509876
12	Kal Buarara	1234567890123464	1236564789
13	Neerai Raiout	1234567890123465	1675849305
15	Sameer Goel	1234567890123466	1578893409
10	Dharit Shah	1234567890123467	1758483934
7	Anubhav Gupta	1234567890123468	1657483948
5	Neha Verma	1234567890123469	1784967348

- 3) View to display the order status of all the customers who have ordered online plans along with gadgets.

```
create view displayOrderStatus as
select CustomerId, CustomerName, orders.orderstatus from customer inner join orders on customer.customerid = orders.ordercustomerid;
```

```
select * from displayOrderStatus;
```

CustomerId	CustomerName	orderstatus
1	Jishnu Vasudevan	Shipped
2	Harsh Shah	Partially Shipped
3	Rachana Rambhad	Payment Incomplete
4	Lagan Gupta	Order Cancelled
5	Neha Verma	Pending
6	Aniel Patel	Refund Initiated
7	Anubhav Gupta	Order Cancelled
8	Aditva Joshi	Pending
9	Parnal Diche	Partially Shipped
10	Dharit Shah	Shipped
11	Girish Sanai	Order Cancelled
12	Kal Buarara	Partially Shipped
13	Neeraj Raibut	Payment Incomplete
14	Shruti Mehta	Shipped
15	Sameer Goel	On The way
16	Vivashree Updli	Order Cancelled
17	Rohit Kamble	Order Declined
18	Privanka Desai	Refund Initiated
19	Komal Shirodkar	Pending
20	Simmah Kazi	On The way

- 4) View the department name of all the employees according to their department name.

```
create view displayDepartment as
select EmployeeId, Employee_Name, department.DepartmentName from Employee
inner join department on employee.employeeid = department.departmentdid;
```

```
Select * from displayDepartment;
```

EmployeeId	Employee_Name	DepartmentName
1	Oias Phansekar	Information Technology
2	Shrevas Kalavanaraman	Sales & Marketing
3	Saurabh Kulkarni	Finance
4	Vivek Shetve	Human Resource
5	Mihir Patil	Customer Care

5) View to display plans of a particular customer.

```
create view displayPlans as
select * from planinclusions where PlanId =
(select PlanInclusionId from plans where PlansId =
(select SimPlanNumber from simdata where SimCustomerId =
(select CustomerId from customer where CustomerName = 'Jishnu Vasudevan')));
```

Select * from displayPlans;

PlanId	Data	Talktime	TextMessages
5	500MB	300 Minutes	500

6) View to display work location of a particular customer.

```
create view displayWorkLocation as
select EmployeeId,Employee_Name,worklocation.locationname from Employee
inner join employeeworklocation
on employee.employeeid = employeeworklocation.workemployeeid
inner join worklocation
on employeeworklocation.workemployeeid = worklocation.locationid where EmployeeId = 3;
```

Select * from displayWorkLocation;

EmployeeId	Employee_Name	locationname
3	Saurabh Kulkarni	New York

7) View to display the salesperson allotted to a particular person.

```
create view displaySalespersonAlloted as
select CustomerName ,employee.employee_name as SalesPersonAlloted from customer
inner join salesperson
on customer.customersalespersonid = salesperson.salespersonid
inner join employee
on salesperson.idemployeesalesperson = employee.employeeid;
```

Select * from displaySalespersonAlloted;

CustomerName	SalesPersonAlloted
Jishnu Vasudevan	Mihir Patil
Neha Verma	Mihir Patil
Parnal Diche	Mihir Patil
Neerai Raiout	Mihir Patil
Rohit Kamble	Mihir Patil
Harsh Shah	Karan Thevar
Aniel Patel	Karan Thevar
Dharit Shah	Karan Thevar
Shruti Mehta	Karan Thevar
Privanka Desai	Karan Thevar
Rachana Rambhad	Chetan Mistrv
Anubhav Gupta	Chetan Mistrv
Girish Sanai	Chetan Mistrv
Sameer Goel	Chetan Mistrv
Komal Shirodkar	Chetan Mistrv
Laan Gupta	Shantanu Sawant
Aditva Joshi	Shantanu Sawant
Kal Burara	Shantanu Sawant
Viiavshree Uppili	Shantanu Sawant
Simmah Kazi	Shantanu Sawant

8) View to show the tracking status of all the customer's orders.

```
create view displayTrackingStatus as
select customername,tracking.trackingid,tracking.trackingstatus from customer
inner join orders
on customer.customerid = orders.ordercustomerid
inner join tracking
on orders.orderid = tracking.trackingorderid;
```

Select * from displayTrackingStatus;

customername	trackingid	trackingstatus
Dharit Shah	1	On the wav
Parnal Diche	2	Arrived to courier service
Shruti Mehta	3	Near bv closest disoatch location
Viiavshree Uppili	4	Arrived to courier service
Rohit Kamble	5	Arrived to courier service
Komal Shirodkar	6	On the wav
Anubhav Gupta	7	Near bv disoatch location
Harsh Shah	8	Arriving Tomorrow
Neerai Raiout	9	Arriving Today

User Privileges

1) HR Access

```
create user 'HRUser'@'localhost' identified by 'hruser';
```

```
grant all on dbms_project.* to 'HRUser'@'localhost';
```

2) Salesperson Access

```
create user 'salesperson'@'localhost' identified by 'salesperson';
```

```
grant select on dbms_project.salesperson to 'salesperson'@'localhost';
```

```
grant select on dbms_project.callrecords to 'salesperson'@'localhost';
```

```
grant select on dbms_project.customer to 'salesperson'@'localhost';
```

```
grant select on dbms_project.simdata to 'salesperson'@'localhost';
```

```
grant select on dbms_project.phoneNumber to 'salesperson'@'localhost';
```

```
grant select on dbms_project.plans to 'salesperson'@'localhost';
```

```
grant select on dbms_project.planInclusions to 'salesperson'@'localhost';
```

```
grant select on dbms_project.Billinginformation to 'salesperson'@'localhost';
```

```
grant select on dbms_project.orders to 'salesperson'@'localhost';
```

```
grant select on dbms_project.tracking to 'salesperson'@'localhost';
```