Airline Company

Data warehouse

Mohamed AlGhaly

DWH Modeling project for an Airline Company

Project Documentations

This word file will be a walkthrough of the complete cycle of the modeling and will hold the complete data generated throughout the process illustrated and well documented, Enjoy.

Project Objective:

The objective of this project is to create a robust and structured Data Warehouse that consolidates data from various sources, facilitating analytical reporting and business intelligence to enhance the airline's decision-making processes and performance analysis.

STEPS:

1- Defining Business Processes
2- Defining Business KPIs (Data Warehouse Objectives)
3- Defining Granularity for the Analysis Scope
4- defining sparsity, summarizability problems, and size Estimation
5- Defining Dimensions, Facts, and measurements
6- Data Integration
7- Data Warehouse Modeling (Defining the Schema)
8- Defining the Physical Model
9- Data Warehouse Indexing and Partitioning
10- Generating & Populating the data
11- Extracting insights to support the Management Decisions

Let’s dive into the modeling process!

STEP1

Step one will be defining:

Business Processes:

NOTE:

We are not going to talk about(model) all of those, we are not about to have a PHD in DWH Modeling, we will just choose the most critical ones.

1- Flights:

- This process involves the flights organized by the company, which includes determining routes, departure timing, arrival timing, aircraft, crew, and more for each flight.
- But we won't talk about pricing here, just go on with me.

2- Marketing Analysis:

- This process will also include the customers' upgrade, flyer miles, and promotions.
- This process is mainly concerned with the marketing team of the airline company.

3- Reservations:

- This process mainly involves the sale of tickets for customers, which is one of the main sources of revenue.
- It includes collecting customer info, booking of seats, issuance of tickets, management of ticket cancellation, and changes of bookings.

4- Check-In:

- This process involves the check-in of passengers at the airport, which involves the verification of travel documents, baggage screening, and seat allocation.
- In one word it is the efficient and organized process of getting passengers onto the aircraft.

5- In-flight Services:

- This process includes the provision of services to passengers during the flight, such as food and beverage service, entertainment, and any other special requests.

6- Baggage Handling:

- This process involves handling of passenger baggage, which includes the secure transfer of baggage from check-in to the aircraft and from the aircraft to baggage claim to passengers.

7- Aircraft Maintenance:

- This process involves the regular maintenance and repair of aircraft to ensure they are safe and operable.
- This process may also involve the replacement of aircraft parts, inspections, and testing.

8- Revenue Management:

- This process involves the management of pricing and inventory, which includes the determination of ticket prices, the allocation of seats, and the optimization of revenue.

9- Customer Services:

- This process involves three main relative branches:
  i. Customer Inquiries.
  ii. Customer Complaints.
  iii. Customer Feedback.
- Mainly concerned with customer satisfaction.

We are going to tackle just four business processes at most (Flights, Marketing Analysis, Reservations, and Customer Services).

STEP 2

Step two will be defining which questions we want our model to tackle:

Questions (Will be answered along with a specific time period)

1- Which customers use our services more frequently.
2- What are the most popular flight routes/Timings/airports/aircrafts.
3- What are the most profitable flight routes/Timings/airports/aircrafts.
4- Which customers contribute the most to our revenue.
5- What are the main sources of revenue for the company.
6- What is the impact of marketing, promotions, and flyer miles system on the revenue.

7- Which customer rank (gold, platinum, titanium) are most profitable to us.
8- Which booking channel is most rewarding.
9- What are the aspects that we need to improve to achieve better customer satisfaction.
10- Which flights receive the best/worst customer feedback.
11- Which crew members are most successful/lovable.
12- How to improve our business.
13- What are the booking patterns of frequent flyers, and what types of fare classes do they typically book.
14- How do customer demographics, such as age or income level, impact travel behavior and preferences.
15- Way, way more, but just keep up with this.

STEP 3

Step three will be defining granularity for each business process:

1- Flights:

- The most detailed grain is the combination of individual aircraft, route, source airport, destination airport, departure time, arrival time, captain, and flight attendant.

2- Marketing Analysis:

- The most detailed grain is the combination of individual customer, upgrade, flyer miles received, flyer miles redeemed, respond to promotions, over a specific flight at a given date.
- I preferred just for the simplicity of the modeling to separate all marketing related processes into a single business process, in which will analyze the marketing performance, by

analyzing each individual action taken in response to a marketing campaign.

3- Reservations:

- The most detailed grain is the ticket issued for an individual customer, for a specific flight, on a specific aircraft, having a certain route, from a certain airport at a given departure time, to an arrival airport at a specific arrival time, having a specific fare basis, for a specific seat, channel, and a payment method.

4- Customer Services:

- The most detailed grain is the combination of an individual customer care action (Inquiry, Feedback, Complaint) for a specific customer, on a specific flight, reservation channel, with payment method, on a specific crew member\s, and a flight captain at a given date.

STEP 4

Step four will be defining sparsity and summarizability problems:

NOTE:

Defining both will NEED to know the structure of each data source used to integrate the data warehouse so this section will be just imaginary estimation.

Sparsity:

For the sparsity any percentage given will be a joke without having access to the data sources, and estimating tables' sizes also will be impossible so we will just skip those two steps.

Here are some summarizability problems:

- For the marketing analysis process the respond to promotion may be incomplete as it is impossible to collect each flyer response to each promotion, but further data collection may be useful.
- For the marketing analysis process the overnight stand data may be incomplete as it is hard to collect such pieces of information so that column will contain tons of missing values.
- For the customer service process, it is not possible to ensure that each customer will include the needed data for each action so some complaint may lack which crew member, flight, …, etc he is complaining about, and so on.
- Those were just some of lots of summarizability problems, but as we have no access to any of the data source and we will just use imaginary data we will assume everything is magically perfect.

STEP 5

Step five will be the capstone for the project which involves determining both facts and dimensions.

FACTS & MEASUREMENTS:

1- Flights Fact (Transactional Fact Table)

a. What?

i. This is a fact table to keep track of each flight and collect some useful data about it, which will help us identify which flights are most popular, which aircraft/route/time/airport/captain/flight attendant is most popular, and much more.

b. Measurements:

i. Number of passengers(additive).
ii. Number of empty seats(additive).
iii. Number of the crew (non-additive as it is pointless to sum up the crew members over different flights).
iv. Number of booked tickets(additive), to keep track of any absent passengers.

c. Dimensions:
i. AIRCRAFT
ii. ROUTE
iii. DATE
iv. AIRPORT
v. CREW MEMBER
- For simplicity we will assume a flight has a single flight attendant.

2- Marketing Analysis Fact (Factless Fact Table)

a. What:
i. Just a factless fact table to help us track the marketing team performance, which customers show the best response to an upgrade/promotion, or uses the flyer miles system, and much more.

b. Measurements:
- No measurements will be used.

c. Dimensions:
i. CUSTOMER
ii. DATE
iii. UPGRADE
iv. FLYER MILES
v. PROMOTION
vi. FLIGHT

3- Reservations Fact (Transactional Fact Table)

a. What:
i. This is the core of our DWH which will help the financial team decides which customer brings the most value, which flight/airplane/airport/route delivers the best profit, which

reservation channel is most rewarding, which payment method delivers the most money, and much more.

b. Measurements:
    i. BASE PRICE (additive).
    ii. OVERNIGHT STAND (semi-additive).
    iii. DISCOUNT/PROMOTION (non-additive).
    iv. NET PRICE (additive).
    v. DISTANCE IN MILES (semi-additive).
c. Dimensions:
    i. CUSTOMER
    ii. FLIGHT
    iii. DATE
    iv. FARE BASIS CLASS
    v. PAYMENT METHOD
    vi. CHANNEL

4- Customer Services Fact (Transactional Fact Table)

a. What:
    i. This fact table helps us keep track of customer satisfaction and provides ways to improve the company's performance by responding to clients' needs.
b. Measurements:
    i. Severity (non-additive).
c. Dimensions:
    i. CUSTOMER
    ii. ACTION
    iii. FLIGHT
    iv. PAYMENT METHOD
    v. CHANNEL
    vi. CREW MEMBER
    vii. DATE

DIMENSIONS:

NOTE:

The attributes of each dimension will be shown in the schema.

1- Aircraft

- Conformed Dimension.
- A dimension holding data about all the airplanes the company owns.

2- Route

- Conformed Dimension.
- Holding data about all routes for the company's flights.

3- Date

- Role-Playing Dimension.
- Typical calendar dimension for any DWH.

4- Airport

- Conformed Dimension.
- A dimension holding data about the airports that our aircrafts use.

5- Crew Member

- Conformed Dimension.
- A dimension holding data about all the employees in the company.

6- Customer

- Conformed Dimension.
- A typical dimension holding data about the passengers.

7- Upgrade

- Conformed Dimension.
- Holds data about all types of upgrades the company offers for its customers.

8- Flyer Miles

- Conformed Dimension.
- Holds data about the flyer miles classes that the company offers.

9- Promotion

- Conformed Dimension.
- Holds data about the discounts that the company offers for its customers.

10- Fare Bases Class

- Conformed Dimension.

- The tickets classes (Economy, VIP, and so on).

11- Payment Method

- Conformed Dimension.
- The different methods of payment the company offers.

12- Channel

- Conformed Dimension.
- The channels the company supports for tickets' reservation.

13- Action

- Conformed Dimension.
- Holds data from the application filled by customer (feedback, inquiry, or complaint).

14- Bridge Flight

- A bridge (junction or intermediary Dimension) that is used to connect fact tables with the flight fact table, we could have used a cancellation schema and connected fact tables together directly, but I prefer this approach.
- We will need to have a flight dimension as a context provider for every business process except for the flights process, so a bridge table is the most smart and efficient way to achieve that.

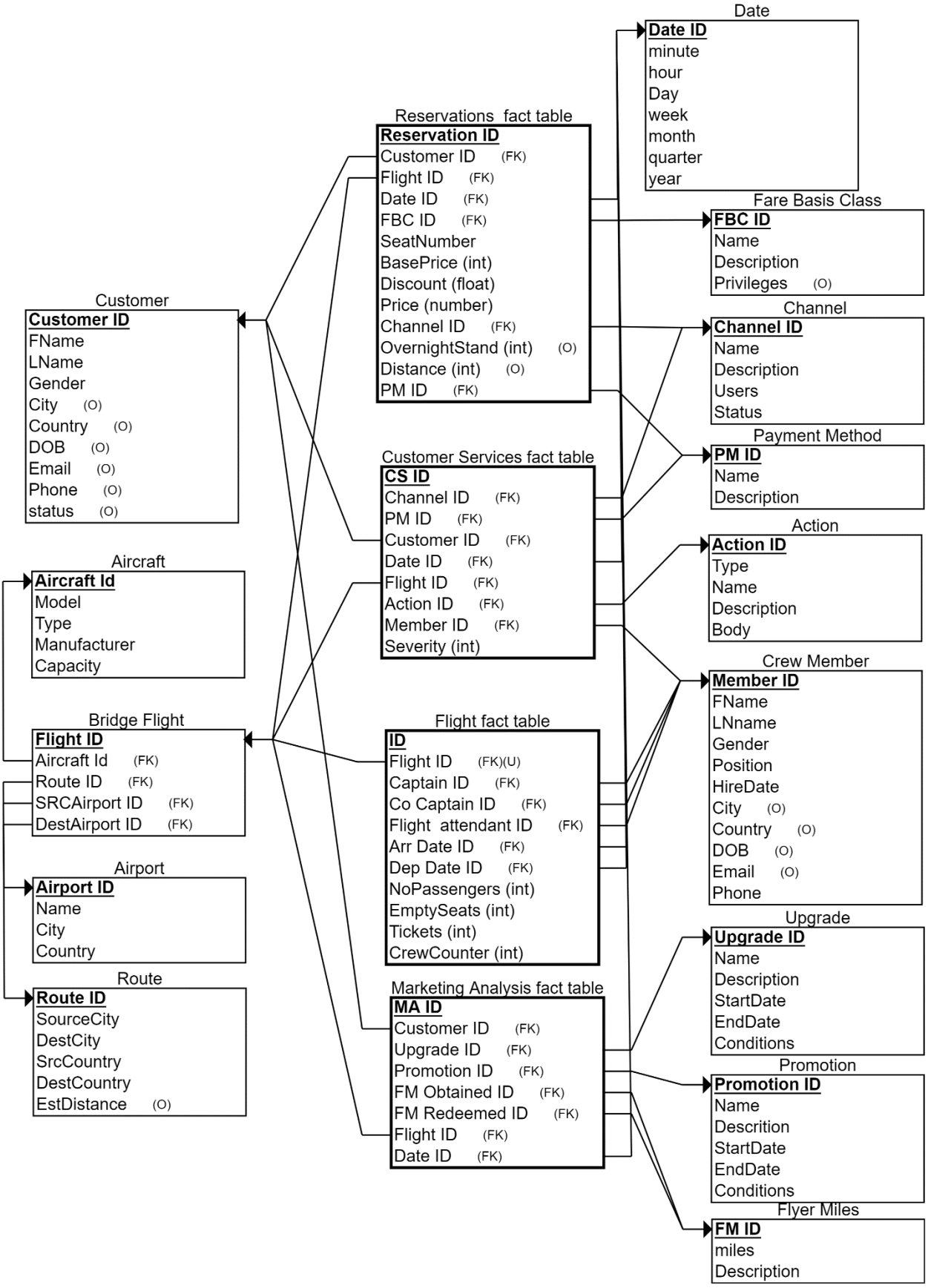
STEP 6

Step six typically will be integrating all data sources into a single point of truth, but having no access to data sources we will just assume we already have a single well-structured data source.

STEP 7 A

Step seven will be the schema modeling, and we will use a star schema.

You waited so long for this step to come but here we go.



STEP 7 B

Step seven B is a discussion about step 7.

WHY did we choose star schema modeling?

A star schema is perfect for our design for some reasons:

1- Simplified query performance: Star schema modeling allows us for simplified and optimized query performance. Since the fact table is at the center of the schema and connected to the dimension tables through foreign keys, queries can be executed efficiently and quickly, without the need for complex joins or subqueries.
2- Improved data analysis: Star schema modeling provides a simplified and intuitive way to analyze data from multiple dimensions. Analysts can easily drill down into data by navigating through the dimension tables, allowing for more complex analysis and insights.
3- Easier maintenance: Star schema modeling is easier to maintain than other modeling approaches. Since each dimension table is connected directly to the fact table, changes to one dimension table will not affect other tables in the schema. This makes it easier to modify and update the data warehouse over time.
4- Scalability: Star schema modeling is highly scalable and can handle large amounts of data. By separating the data into smaller, more manageable tables, the schema can accommodate large amounts of data without impacting query performance or data analysis.

Overall, star schema modeling is a popular and effective approach for designing data warehouses. It provides a simple, intuitive, and scalable way to organize and analyze data, making it easier for analysts and business users to get the insights they need from their data.

It mainly consists of 3 components:

- Dimensions: integral companions to a fact table. containing the textual context associated with a business process measurement event.
- Facts: stores the performance measurements resulting from an organizations' business process events.
- Measurements: The actual measurements stored in the fact tables.

For a simple data warehouse just like the one we are modeling a star schema is the perfect fit as it covers all our business needs, with high performance, data integrity, and powerful possibilities.

STEP 8

Step nine will be creating the table in both excel sheet for validation and in oracle database for analysis.

CUSTOMER TABLE (DIMENSION 1)		
Customer_ID (PK)	int	clustered
Fname	varchar	
Lname	varchar	
Gender	Char (1)	bitmap
City	varchar	
Country	varchar	non-clustered
DOB	date	
Email	varchar	unique
Phone	varchar	
Status	varchar	bitmap

CREW MEMBER TABLE (DIMENSION 8)		
Member_ID(PK)	int	clustered
Fname	varchar	
Lname	varchar	
Gender	char(1)	bitmap
City	varchar	
Country	varchar	non-clustered
DOB	date	
Email	varchar	unique
Phone	varchar	
Position	varchar	non-clustered
Hire_Date	date	

AIRCRAFT TABLE (DIMENSION 2)		
Aircraft_ID (PK)	in	clustered
Model	varchar	
Type	varchar	non-clustered
Manufacturer	varchar	
Capacity	int	

ACTION TABLE (DIMENSION 9)		
Action_ID (PK)	int	clustered
Name	varchar	
Type	varchar	bitmap
Description	varchar	
Body	varchar	

BRIDGE FLIGHT (DIMENSION 3)		
Flight_ID (PK)	int	clustered
Aircraft_ID (FK)	int	b_tree
Route_ID (FK)	int	b_tree
SRCAirport_ID (FK)	int	b_tree
DSTAirport_ID (FK)	int	b_tree

CHANNEL TABLE (DIMENSION 10)		
Channel_ID (PK)	int	clustered
users	int	
Name	varchar	
Description	varchar	
status	bool	bitmap

AIRPORT TABLE (DIMENSION 4)		
Airport_ID (PK)	int	clustered
Name	varchar	
City	varchar	

FARE BASIS CLASS TABLE (DIMENSION 11)		
FBC_ID (PK)	int	clustered
Name	varchar	bitmap
Discription	varchar	

Country	varchar	non-clustered

Privileges	varchar	

ROUTE TABLE (DIMENSION 5)		
Route_ID (PK)	int	clustered
Src_City	varchar	
Dest_City	varchar	
Src_Country	varchar	
Dst_Country	varchar	
Est_Distance	int	

DATE TABLE (DIMENSION 12)		
Date_ID (PK)	int	clustered
Minute	int	
Hour	int	
Day	int	
Week	int	
Month	int	
Quarter	int	
Year	int	

FLYER MILES TABLE(DIMENSION 6)		
FM_ID (PK)	int	clustered
Miles	int	
Description	varchar	

PAYMENT METHOD TABLE (DIMENSION 13)		
PM_ID (PK)	int	clustered
Name	varchar	
Description	varchar	

PROMOTION TABLE (DIMENSION 7)		
Prom_ID (PK)	int	clustered
Name	varchar	
Description	varchar	
Start_Date	date	
End_Date	date	
Conditions	varchar	

UPGRADE TABLE (DIMENSION 14)		
Upgrade_ID (PK)	int	clustered
Name	varchar	
Description	varchar	
Start_Date	date	
End_Date	date	
Conditions	varchar	

MARKETING ANALYSIS TABLE (FACT 3)		
MA_ID (PK)	int	clustered
Customer_ID (FK)	int	B_Tree
Upgrade_ID (FK)	int	B_Tree
Promotion_ID (FK)	int	B_Tree
Date_ID (FK)	int	B_Tree
Flight_ID (FK)	int	B_Tree
DM-Obt_ID (FK)	int	B_Tree
FM_Red_ID (FK)	int	B_Tree

CUSTOMER SERVICES TABLE (FACT 4)		
CS_ID (PK)	int	clustered
Customer_ID (FK)	int	B_Tree
Channel_ID (FK)	int	B_Tree
PM_ID (FK)	int	B_Tree
Date_ID (FK)	int	B_Tree
Flight_ID	int	B_Tree
Action_ID (FK)	int	B_Tree
Employee_ID (FK)	int	B_Tree
Serverity	int	

RESERVATION TABLE (FACT 1)		
Reservation _ID (PK)	int	clustered
Customer_ID (FK)	int	B_Tree

Flight_ID (FK)	int	B_Tree
Date_ID (FK)	int	B_Tree
FCB_ID (FK)	int	B_Tree
Channel_ID (FK)	int	B_Tree
PM_ID (FK)	int	B_Tree
Base_Price	int	
Discount	float	
Price	int	
Seat_Number	int	
Overnighr_stand	int	
Distance	int	

FLIGHT TABLE (FACT 2)		
ID (PK)	int	clustered
Flight_ID (FK)	int	B_Tree
Captain_ID (FK)	int	B_Tree
Co_Captain_ID (FK)	int	B_Tree
Fli_Att_ID (FK)	int	B_Tree
Arr_Date_ID (FK)	int	B_Tree
Dep_Date_ID (FK)	int	B_Tree
No_Passegers	int	
Empty_Seats	int	
Tickets	int	
Crew_Counter	int	

THE SCRIPT FOR GENERATING THE ORACLE TABLES AND POPULATING SAMPLE DATA (RANDOMLY GENERATED USING A PROGRAMMING LANGAUGE LIKE PYTHON OR WITH A THIRD-PARTY TOOL LIKE “mockaroo.com”) WILL BE ATTACHED.

STEP 9

In step 10 let’s talk for a little bit about each type of index we used and why did we choose such an index.

NOTE:

Taking about each index and the algorithm used to implement it would be a lot of fun, but it is way out of the scope of this project and will take a long time, so we will just give a hint on each one, Enjoy.

INDEX TYPE	WHEN	WHY
CLUSTERED	Primary Key Columns	physically order the data in a table based on the indexed column.
B_TREE	Foreign Keys Columns	highly efficient for range-based queries.
UNIQUE	unique columns	This index enforces uniqueness for the indexed column or columns.
HASH	Columns used in equality filtering	Uses hash-function to retrieve data in constant time so it is used for equality filtering and join conditions.

NON-CLUSTERD	Categorial Data (With wider value range)	stored separately from the data and contains a copy of the indexed column, along with a pointer to the corresponding data.
BITMAP	Categorial Data (With 2-5 values)	It uses a bitmap to represent the data, with each bit representing a possible value, so it is used on columns with a small number of distinct values.

For me, I prefer using a clustered index on the PK columns, non-clustered index on categorial data with wider range of values, b-tree for categorial data with 2-5 values, unique index for any unique column, and the last one is my personal favorite which can do magic, hash index can use a hash function to retrieve a piece of data from a table containing a zillion row in just O(1) time so I prefer using such index on any column used a lot in where statement with an equal sign maybe for filtering or joining tables, If I am about to implement a database engine, Hash Index will be forced on both primary and foreign keys.

NOTE:

Indexes can cause overhead or kill the performance of your database, so use it wisely knowing what you are doing, or just leave it to the DBMS and it will do a great job for you, unless you can do a greater job DO NOTHING.

The appropriate type of index for a particular situation will depend on a variety of factors, including the size and type of data, the frequency and type of queries, and the overall database design.

STEP 10

In step eleven will be generating and populating data into the data warehouse.

STEP 11

In step twelve will be answering questions with SQL queries.

1- Which customers use our services more frequently.

```
SELECT FNAME + ' ' + LNAME AS
CUST_NAME,
count(R.Reservation_ID) AS FREQUENCY
FROM RESERVATIONS_ R, CUSTOMER C
where R.Customer_ID = C.Customer_ID
GROUP BY FNAME + ' ' + LNAME
ORDER BY FREQUENCY DESC;
```

	CUST_NAME	FREQUENCY
1	Anet Ferrario	8
2	Luigi Catley	8
3	Jarrod Gress	8
4	Jarad Bruniges	8
5	Josey Dunston	8
6	Laughton Stitt	7
7	Shandeigh Hissie	7
8	Zabrina Starbeck	7
9	Nickolas Asals	7
10	Peirce Guilloux	7

2- What are the most popular flight routes/Timings/airports/aircrafts.

```
SELECT R.NAME AS ROUTE,
COUNT(*) AS FREQUENCY
FROM FLIGHT F,
Bridge_Flight_ B, ROUTE R
WHERE F.ID = B.Flight_ID
AND
R.Route_ID = B.Route_ID
GROUP BY R.NAME
ORDER BY FREQUENCY DESC;
```

	ROUTE	FREQUENCY
1	Cungapmimbo - Indonesia => Patitírion - Greece	23
2	Banjar Baleagung - Indonesia => Menzel Abderhama...	20
3	Ágios Matthaíos - Greece => New Shagunnu - Nigeria	20
4	Seres - Philippines => Ballinteer - Ireland	20
5	Zhoukou - China => Huang'ao - China	19
6	Qinshan - China => Kungsbacka - Sweden	18
7	Baklashi - Russia => Sangar Saray - Afghanistan	18
8	Tân Châu - Vietnam => Santa Ana - Venezuela	18
9	Conchal - Brazil => Shigutang - China	18
10	Champaign - United States => Benito Juarez - Mexico	18

```
SELECT A.Model AS PLANE, COUNT(*) AS
FREQUENCY FROM
FLIGHT F, Bridge_Flight_ B, Aircraft A
WHERE F.ID = B.Flight_ID
AND A.Aircraft_Id = B.Aircraft_Id
GROUP BY A.Model
ORDER BY FREQUENCY DESC;
```

	PLANE	FREQUENCY
1	model2	783
2	model8	747
3	model4	670
4	model9	663
5	model5	635
6	model3	556
7	model10	544
8	model6	504
9	model7	458
10	model1	440

3- What are the most profitable flight routes/Timings/airports/aircrafts.

```
SELECT R.NAME AS ROUTE,
SUM(PRICE_) AS TOTAL
FROM Reservations_,
Bridge_Flight_ B , ROUTE R
WHERE
Reservations_.Flight_ID =
B.Flight_ID
AND B.Route_ID = R.Route_ID
GROUP BY R.NAME
ORDER BY TOTAL DESC;
```

	ROUTE	TOTAL
1	Ágios Matthaíos - Greece => New Shagunnu - Nigeria	198205
2	Angren - Uzbekistan => Taposan - Indonesia	193969
3	Baklashi - Russia => Sangar Saray - Afghanistan	181346
4	Mirzec - Poland => Klokot - Kosovo	180075
5	Eišiškes - Lithuania => Zhamog - China	176945
6	Zhoukou - China => Huang'ao - China	176760
7	Gamawa - Nigeria => El Guapinol - Honduras	175529
8	Gândara - Portugal => Shilipu - China	174744
9	Margaharja - Indonesia => Plan de Ayala - Mexico	170334
10	Belköl - Kazakhstan => Créteil - France	168083
11	Daogao - China => Sirajganj - Bangladesh	167649
12	Tampa - United States => Lomintsevskiy - Russia	167227
13	Banjar Baleagung - Indonesia => Menzel Abderhama...	165527
14	Kebonkai - Indonesia => Itapé - Paraguay	164768

```
SELECT A.NAME AS AIRPORT,
SUM(PRICE_) TOTAL
FROM Reservations_,
Bridge_Flight_ B , AIRPORT A
WHERE Reservations_.Flight_ID
= B.Flight_ID
AND (B.SRCAirport_ID =
A.Airport_ID OR
B.DestAirport_ID =
A.AIRPORT_ID)
GROUP BY A.NAME
ORDER BY TOTAL DESC;
```

	AIRPORT	TOTAL
1	Springdale Municipal Airport	1070289
2	Beale Air Force Base	942139
3	Wallal Airport	882802
4	Shahroud Airport	769844
5	Rouyn Noranda Airport	723332
6	Haugesund Airport	720491
7	William P Hobby Airport	670685
8	Dera Ghazi Khan Airport	666818
9	Ontario International Airport	663856
10	Mallacoota Airport	652426
11	Capitan FAP Carlos Martinez De Pinillos Internat...	634519
12	Zulu Inyala Airport	633705
13	Attu Heliport	627731
14	Mount Keith Airport	623219

4- Which customers contribute the most to our revenue.

```
SELECT FNAME + ' ' + LNAME AS CUST_NAME,
SUM(PRICE_) TOTAL
FROM
    Reservations_ R,
    Customer C
WHERE
    R.Customer_ID = C.Customer_ID
GROUP BY FNAME + ' ' + LNAME
ORDER BY TOTAL DESC;
```

Results Messages

	CUST_NAME	TOTAL
1	Jarad Bruniges	42170
2	Andrea Schimpke	34326
3	Aurilia Capron	33989
4	Anet Ferrario	33811
5	Indira Klimczak	33572
6	Josey Dunston	33507
7	Jarrod Gress	32832
8	Christye Furmston	30684
9	Yves Dosdell	30590
10	Gerry Salleir	30112
11	Keelia Coltan	30062
12	Titos Lyenyng	30053
13	Luigi Catley	29924
14	Prinz Izod	29657

5- What are the main sources of revenue for the company.

Collecting together all the precious insights helps us to get some knowledge about which airplane, airport, timing, routes, or even city are popular or gives us good money, so we can go support the decision system to make a wise-data-driven decisions.

6- What is the impact of marketing, promotions, and flyer miles system on the revenue.

```
select count(*) / (max(year) - min(year))
from Marketing_Analysis MA, date D
where MA.Date_ID =D.Date_ID;
```

	ACTION_PER_YEAR
1	125

THERE IS AN AVERAGE OF 125 ACTION TAKEN PER YEAR BY CUSTOMER AS RESPONSES TO A MARKETING CAMPAIGNS.

7- Which customer rank (gold, platinum, titanium) are most profitable to us.

```
SELECT TOP 10
STATUS, SUM(PRICE_) TOTAL
FROM Reservations_ R, Customer C
WHERE R.Customer_ID = C.Customer_ID
GROUP BY STATUS
ORDER BY TOTAL DESC;
```

	STATUS	TOTAL
1	NULL	18679009
2	platinum	10184943
3	titanium	9710818
4	gold	8821710

Null represents non-frequent flyers.

8- Which booking channel/payment method/fare basis is most rewarding/popular (6 in 1).

REWARDING

```
SELECT C.Name CHANNEL, SUM(PRICE_) TOTAL
FROM Reservations_ R, Channel_ C
WHERE R.Channel_ID = C.Channel_ID
GROUP BY C.NAME
ORDER BY TOTAL DESC;
SELECT FB.Name FARE, SUM(PRICE_) TOTAL
FROM Reservations_ R, Fare_Basis_Class FB
WHERE R.FBC_ID = FB.FBC_ID
GROUP BY FB.NAME ORDER BY TOTAL DESC;
```

	CHANNEL	TOTAL
1	Other	12207272
2	website	11882743
3	Application	11823432
4	Company	11483033

	FARE	TOTAL
1	VIP	9781170
2	Economy	9628309
3	firstClass	9529973
4	Cheap	9269121
5	Cabin	9187907

	PAYMENT	TOTAL
1	Card	8004264
2	Miles	7938480
3	Credit	7928427
4	NotPaid	7898033
5	Cash	7847588
6	Others	7779688

```sql
SELECT PM.Name PAYMENT, SUM(PRICE_) TOTAL
FROM Reservations_ R, Payment_Method PM
WHERE R.PM_ID = PM.PM_ID
GROUP BY PM.NAME
ORDER BY TOTAL DESC;
```

----------POPULAR----------

```sql
SELECT C.Name CHANNEL, COUNT(*) TOTAL
FROM Reservations_ R, Channel_ C
WHERE R.Channel_ID = C.Channel_ID
GROUP BY C.NAME
ORDER BY TOTAL DESC;

SELECT FB.Name FARE, COUNT(*) TOTAL
FROM Reservations_ R, Fare_Basis_Class FB
WHERE R.FBC_ID = FB.FBC_ID
GROUP BY FB.NAME
ORDER BY TOTAL DESC;

SELECT PM.Name PAYMENT, COUNT(*) TOTAL
FROM Reservations_ R, Payment_Method PM
WHERE R.PM_ID = PM.PM_ID
GROUP BY PM.NAME
ORDER BY TOTAL DESC;
```

	CHANNEL	TOTAL
1	Other	3807
2	Application	3766
3	website	3724
4	Company	3703

	FARE	TOTAL
1	VIP	3048
2	firstClass	3020
3	Cabin	2998
4	Economy	2979
5	Cheap	2955

	PAYMENT	TOTAL
1	Card	2536
2	Cash	2536
3	NotPaid	2526
4	Miles	2502
5	Credit	2453
6	Others	2447

9- What are the aspects that we need to improve to achieve better customer satisfaction.

This fact table (customer services) is my favorite as u can use ReGex to get patterns from any feedback, inquiry, or complaint to get insights about how to improve customer experience here is just a peak on the power of this fact table:

- Here is the most complained channel/payment method/crew member (We could do much more, but this is just a peak)

```sql
SELECT PM.Name payment, COUNT(*) FREQ, SUM(Severity_) SEV
FROM Customer_Services CS, Payment_Method PM, ACTION A
WHERE CS.PM_ID = PM.PM_ID
AND CS.Action_ID = A.Action_ID
AND A.TYPE = 'Complaint'
GROUP BY PM.Name ORDER BY FREQ DESC;
```

```sql
SELECT C.Name CHANNEL, COUNT(*) FREQ, SUM(Severity_) SEV
FROM Customer_Services CS, Channel_ C, ACTION A
WHERE CS.Channel_ID = C.Channel_ID
AND CS.Action_ID = A.Action_ID
AND A.TYPE = 'Complaint'
GROUP BY C.Name ORDER BY FREQ DESC;

SELECT M.FName + ' ' + M.LNname + ' - ' + M.Position
MEMBER, COUNT(*) FREQ, SUM(Severity_) SEV
FROM Customer_Services CS, Crew_Member M, ACTION A
WHERE CS.Member_ID = M.Member_ID
AND CS.Action_ID = A.Action_ID
AND A.TYPE = 'Complaint'
GROUP BY M.FName + ' ' + M.LNname + ' - ' + M.Position
ORDER BY FREQ DESC;
```

	payment	FREQ	SEV
1	Cash	48	278

	CHANNEL	FREQ	SEV
1	Application	64	367

	MEMBER	FREQ	SEV
1	Baron Gawne - Web Developer I	5	29

10- Which flights receive the best/worst customer feedback.

```sql
SELECT top 1 --worst
FLIGHT_ID, COUNT(*) AS COMPLAINTS
, SUM(CS.Severity_) AS SEV
FROM CUSTOMER_SERVICES CS, ACTION A
WHERE CS.ACTION_ID = A.ACTION_ID
AND A.TYPE = 'Complaint'
GROUP BY FLIGHT_ID
ORDER BY COUNT(*) DESC;
```

	FLIGHT_ID	COMPLAINTS	SEV
1	2055	2	14

```sql
SELECT --best
FLIGHT_ID, COUNT(*) AS FEEDBACK
, SUM(CS.Severity_) AS SEV
FROM CUSTOMER_SERVICES CS, ACTION A
WHERE CS.ACTION_ID = A.ACTION_ID
AND A.TYPE = 'Feedback'
GROUP BY FLIGHT_ID
ORDER BY SUM(CS.Severity_) desc;
```

	FLIGHT_ID	FEEDBACK	SEV
1	1750	2	17

11- Which crew members are most successful/lovable.

```
select M.FName + ' ' + M.LNname
MEMBER,
COUNT(*) FREQUENCY
from Flight F, Crew_Member M
where F.Captain_ID = M.Member_ID
GROUP BY M.FName + ' ' + M.LNname
ORDER BY COUNT(*) DESC;
```

	MEMBER	FREQUENCY
1	Drucie Fruchon	340
2	Sarajane Hardingham	332
3	Ossie Sherringham	325
4	Abe Dibdin	323
5	Myranda Gooly	315

12- How to improve our business.

We have some insights up till now about who/what is doing a good business and we can know much more I am just showing off 1 of a billion of what this data warehouse design can do.

13- What are the booking patterns of frequent flyers, and what types of fare classes do they typically book.

```
SELECT C.STATUS, CH.NAME CHANNEL, PM.NAME PAYMENT,
COUNT(R.CUSTOMER_ID) FREQUENCY
FROM RESERVATIONS_ R,
CUSTOMER C, PAYMENT_METHOD
PM, CHANNEL_ CH
WHERE R.CUSTOMER_ID =
C.CUSTOMER_ID
AND R.PM_ID = PM.PM_ID
AND R.CHANNEL_ID =
CH.CHANNEL_ID
AND C.STATUS IS NOT NULL
GROUP BY C.STATUS, CH.NAME,
PM.NAME
HAVING COUNT(R.CUSTOMER_ID) >
135ORDER BY STATUS, CHANNEL,
PAYMENT;
```

	STATUS	CHANNEL	PAYMENT	FREQUENCY
1	gold	Application	Cash	138
2	platinum	Application	Card	151
3	platinum	Application	Credit	137
4	platinum	Application	Miles	144
5	platinum	Other	Card	139
6	platinum	Other	Cash	145
7	platinum	Other	Credit	141
8	platinum	Other	NotPaid	140
9	platinum	website	Cash	137
10	platinum	website	NotPaid	139
11	platinum	website	Others	147
12	titanium	Company	Cash	137
13	titanium	Other	NotPaid	138
14	titanium	website	Miles	143
15	titanium	website	Others	137

```
SELECT C.STATUS, FBC.Name FARE,
COUNT(R.CUSTOMER_ID) FREQUENCY
FROM RESERVATIONS_ R, CUSTOMER C, Fare_Basis_Class FBC
WHERE R.CUSTOMER_ID = C.CUSTOMER_ID
AND R.FBC_ID = FBC.FBC_ID
```

```
AND C.STATUS IS NOT NULL
GROUP BY C.STATUS, FBC.NAME
HAVING COUNT(R.CUSTOMER_ID) > 600
ORDER BY STATUS, FARE;
```

	STATUS	FARE	FREQUENCY
1	gold	VIP	601
2	platinum	Cabin	624
3	platinum	Cheap	645
4	platinum	Economy	617
5	platinum	firstClass	626
6	platinum	VIP	638
7	titanium	Cabin	618
8	titanium	firstClass	616
9	titanium	VIP	626

14- How do customer demographics, such as age or income level, impact travel behavior and preferences.

The sky is our limit here, I can state zillion SQL statement here everyone shows a different insight like what is the channel/payment method/fare basis class/airport preferred by each age/gender/city/country and way much more we have 21 SQL statements lets just make it 25 and that will be it.

--AGE PATTERN FOR BOTH CHANNEL AND AIRPORT—

```
CREATE VIEW AGE_ AS
(SELECT CASE
WHEN YEAR(GETDATE()) -  YEAR(CU.DOB) > 50
THEN 'Old'
WHEN YEAR(GETDATE()) -  YEAR(CU.DOB) > 30
THEN 'Grown Up'
WHEN YEAR(GETDATE()) -  YEAR(CU.DOB) > 18
THEN 'Youth'
ELSE  'Kid' END AS AGE,
c.Name as channel, A.Name AS AIRPORT
FROM Reservations_ R, Channel_ C , Customer CU,
Bridge_Flight_ B, Airport A
WHERE R.Channel_ID = C.Channel_ID
AND R.Customer_ID = CU.Customer_ID
AND B.Flight_ID = R.Flight_ID
AND B.SRCAirport_ID = A.Airport_ID
AND CU.DOB IS NOT NULL);

SELECT AGE, Channel, COUNT(*) AS
FREQUENCY
FROM AGE_
GROUP BY AGE , CHANNEL
ORDER BY AGE, CHANNEL, FREQUENCY
DESC;
```

	AGE	Channel	FREQUENCY
1	Grown Up	Application	299
2	Grown Up	Company	259
3	Grown Up	Other	268
4	Grown Up	website	289
5	Kid	Application	157
6	Kid	Company	153
7	Kid	Other	158
8	Kid	website	172
9	Old	Application	293
10	Old	Company	325
11	Old	Other	331
12	Old	website	307
13	Youth	Application	164
14	Youth	Company	165
15	Youth	Other	183
16	Youth	website	162

```
SELECT AGE,AIRPORT, COUNT(*)
AS FREQUENCY
FROM AGE_
GROUP BY AGE , AIRPORT
HAVING COUNT(*) > 11
ORDER BY AGE, AIRPORT,
FREQUENCY DESC;
```

	AGE	AIRPORT	FREQUENCY
1	Grown Up	Attu Heliport	12
2	Grown Up	Black Hills Airport-Clyde Ice Field	12
3	Grown Up	Montgomery County Airpark	12
4	Grown Up	Springdale Municipal Airport	13
5	Old	Antrim County Airport	12
6	Old	Botopasi Airport	14
7	Old	Grand Marais Cook County Airport	12
8	Old	La Baule-Escoublac Airport	12
9	Old	Mount Hotham Airport	12
10	Old	Reyes Murillo Airport	12
11	Old	Rouyn Noranda Airport	14
12	Old	Wallal Airport	13
13	Youth	Wallal Airport	15

--GENDER PATTERN FOR BOTH FARE BASIS AND PAYMENT METHOD—

```
SELECT C.GENDER, FB.Name FARE_BASIS,COUNT(*) FREQUENCY
FROM Reservations_ R, Customer C,
Fare_Basis_Class FB
WHERE R.Customer_ID = C.Customer_ID
AND R.FBC_ID = FB.FBC_ID
GROUP BY C.Gender, FB.NAME
ORDER BY GENDER, FARE_BASIS,
FREQUENCY DESC;
```

	GENDER	FARE_BASIS	FREQUENCY
1	F	Cabin	1467
2	F	Cheap	1427
3	F	Economy	1481
4	F	firstClass	1484
5	F	VIP	1543
6	M	Cabin	1531
7	M	Cheap	1528
8	M	Economy	1498
9	M	firstClass	1536
10	M	VIP	1505

```
SELECT C.GENDER, PM.Name
PAYMENT,COUNT(*) FREQUENCY
FROM Reservations_ R, Customer C,
Payment_Method PM
WHERE R.Customer_ID = C.Customer_ID
AND R.PM_ID = PM.PM_ID
GROUP BY C.Gender, PM.NAME
ORDER BY GENDER, PAYMENT, FREQUENCY
DESC;
```

	GENDER	PAYMENT	FREQUENCY
1	F	Card	1272
2	F	Cash	1268
3	F	Credit	1182
4	F	Miles	1245
5	F	NotPaid	1214
6	F	Others	1221
7	M	Card	1264
8	M	Cash	1268
9	M	Credit	1271
10	M	Miles	1257
11	M	NotPaid	1312
12	M	Others	1226

STEP 12

In step 13 which is the last one I just want to point out, how it was easy for us to answer all the questions stated before the design, and how we got the ability to answer any question that may come into your mind, and all of it was possible as the design was perfectly modeled, covering each little aspect of the business, making it the easiest thing in the world to answer a complicated question with a simple statement.

Here I covered a 25-business question out of zillion possible ones, finally thank you for reaching this step, it was a fun project, Enjoy.