HEALTHCARE BUSINESS INTELLIGENCE

by Soumya Tejas with **The School of Management, University at Buffalo**
September 2017

CONTENTS

BUSINESS OBJECTIVES

Background: Some time ago, for our evening clinic (visits after 5:30pm) we explored the idea of providing telephone visits as an alternative to a face-to-face visit to patients to improve service. We felt that this would be a more convenient alternative for the patient to get their medical needs met without having to come in. Attached is about 6 months' worth of visits, which includes the telephone encounters. After we implemented the program, we wanted to use the visit data to determine how to staff for the evening appointments.

Query: How many physicians would you need to staff the after-hours (appointments after 5:30pm) telephone clinic? Assume, that 1 physician can see 4 telephone visits per hour.

We have three dates in the appointment data – appointment_date, booking_date, and checking_date. For the business query, we need to pick the right field between appointment_date and checkin_date. By the below query, we are checking if data for both fields is available for cases where patients have showed up.

```
proc sql;
    title 'completeness check - checkin_date';
    select count(*) 'Missing data records'
    from
    (select * from work.app1
    where checkin_date is null
    and show_code='Y'
    union
    select * from work.app2
    where checkin_date is null
    and show_code='Y');
```

completeness check - checkin_date

Missing data records
52

```
proc sql;
    title 'completeness check - appointment_date';
    select count(*) 'Missing data records'
    from
    (select * from work.app1
    where appointment_date is null
    and show_code='Y'
    union
    select * from work.app2
    where appointment_date is null
    and show_code='Y');
```

completeness check - appointment_date

Missing data records
0

Considering the above analysis, we will use appointment_date and appointment_time to answer the business queries.

Merging provider's specialization with appointment dataset:

```
/*renaming column name in provider, to merge with appointment data*/
data work.provider;
set work.provider;
rename prov_id=provider_id;
run;

/*merging provider's specialization*/
data app1_n;
    merge work.app1(keep=appointment_date appointment_time show_code provider_id patient_id)
    work.provider(keep=provider_id prov_spec);
run;

data app2_n;
    merge work.app2(keep=appointment_date appointment_time show_code provider_id patient_id)
    work.provider(keep=provider_id prov_spec);
run;
```

And running the same code for the second part of appointment dataset.

In order to assess the demand of telephonic visits, the reords for in-person visits are removed, and the two subsets of appointment are concatenated.

```
proc sql;
    delete from work.app1_n
    where appointment_type='In-Person Visit';
run;

proc sql;
    delete from work.app2_n
    where appointment_type='In-Person Visit';
run;

data work.appointment;
set work.app1_n work.app2_n;
run;
```

We have 13,219 appointments data for telephonic visit to assess information from.

```
proc sql;
    select count(distinct(appointment_date))
    from work.appointment;
run;
```

The appointments are booked for 193 days. Let us now look at the average number of physicians required in each hour, to assess hourly-demand.

```
proc sql;
    title 'Demand distribution';
    select timeid 'hour', count(*)/(193*4) 'no_of_providers'
    from work.appointment
    group by timeid
    order by timeid desc;
run;
```

Demand distribution

hour	no_of_providers
21	0.001295
20	0.272021
19	1.875648
18	2.406736
17	1.446891
16	1.25
15	1.358808
14	1.414508
13	2.551813
12	0.5
11	1.163212
10	1.278497
9	1.564767
8	0.03886

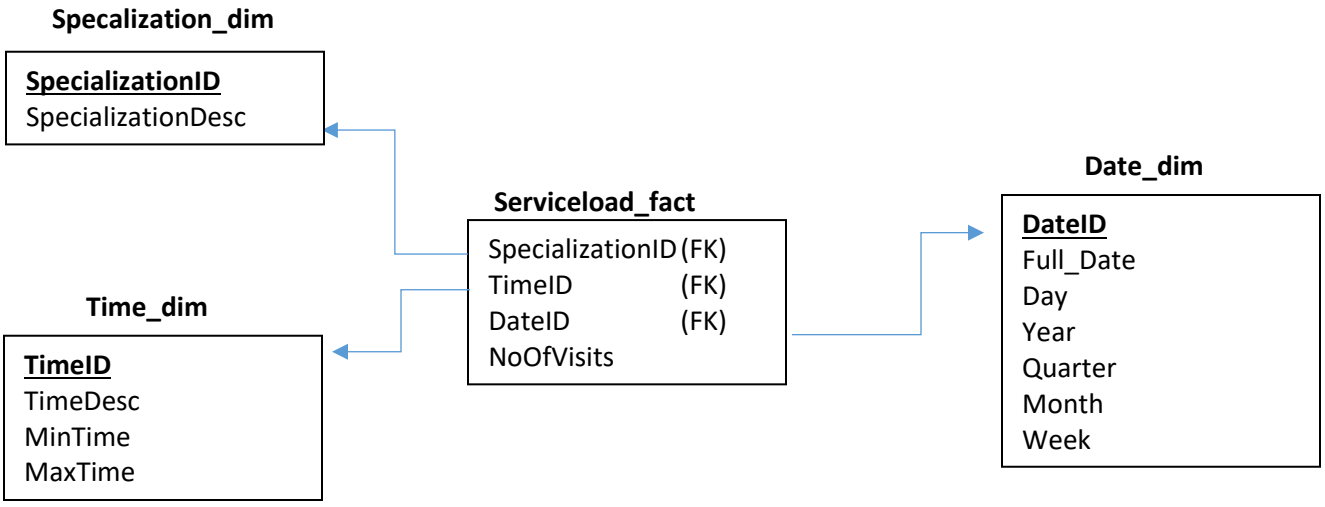
Which suggests the below need:

From	To	No_providers
17:30	18:30	2
18:30	19:30	2
19:30	20:30	1

We need two providers in the first two hours of service, and just one for the last hour of service. In the next section, let us look at the dataset with more granularity.

BUSINESS INTELLIGENCE USING DATA WAREHOUSE

In order to address the business requirements, a data warehouse model is designed using the Kimball approach.



Specialization_dim:

Dimension table for medical specialization - family practice, internal medicine and OBGYN.

Time_dim:

A mini dimension table, which records the category of appointment hour.

Date_dim:

A standard dimension across the industry, to assess date in different formats / scale.

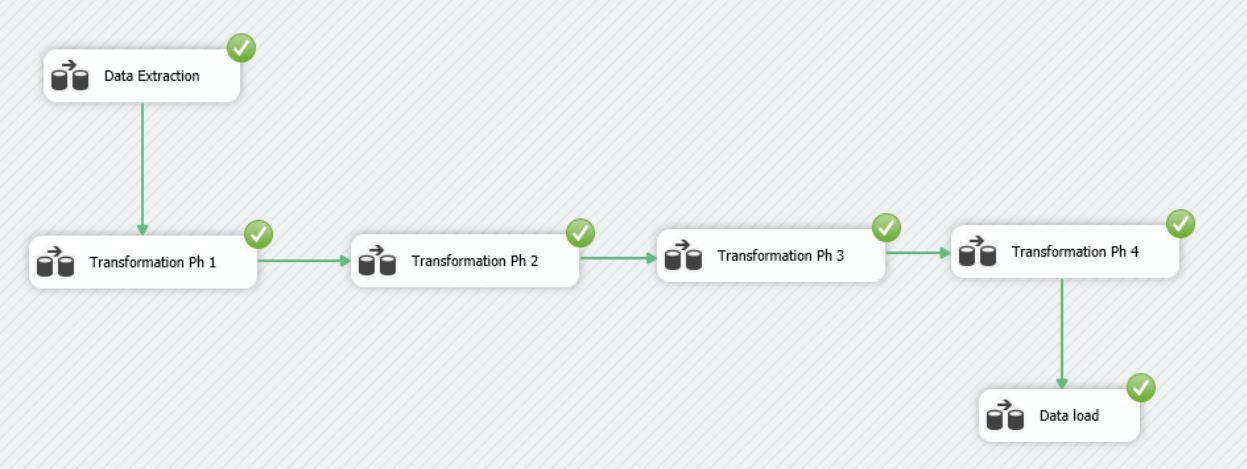
Serviceload_fact:

A fact table, to assess service load at specialization, time category, and day of the week basis.

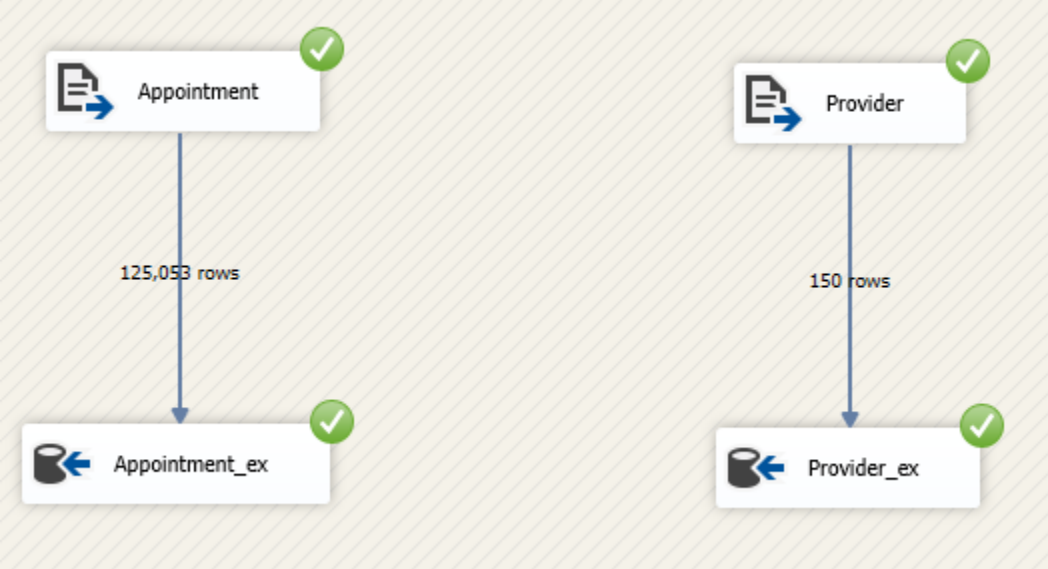
Tools used – SSIS for ETL, Tableau 10.0 for data visualization.

ETL

The ETL process comprises of eight transformations conducted in four data flows. Apart from that, we have one data flow for extraction, and one for loading.

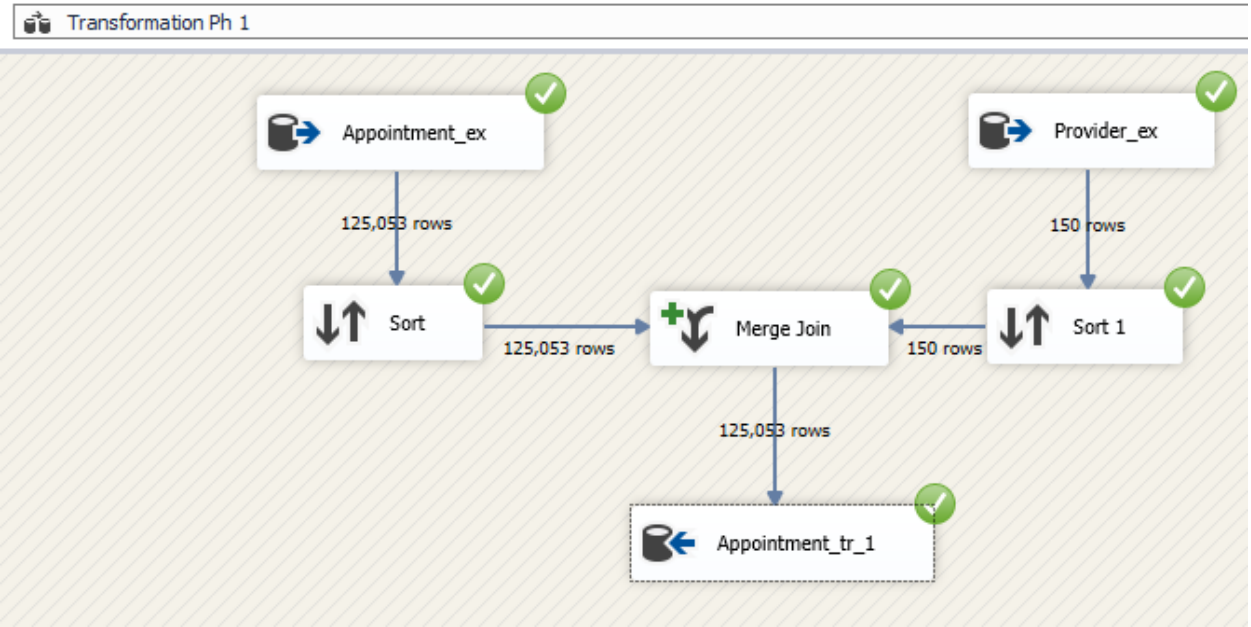


Data extraction:

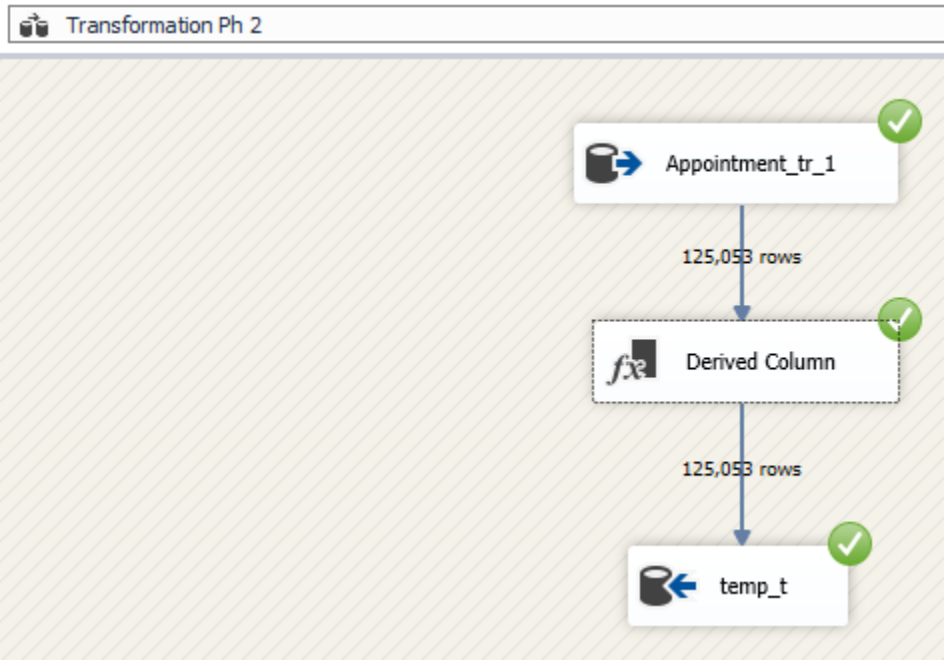


Here we are simply transferring data from the csv files to SQL Server database tables. Once the data is extracted, the SSIS package will proceed with transformation.

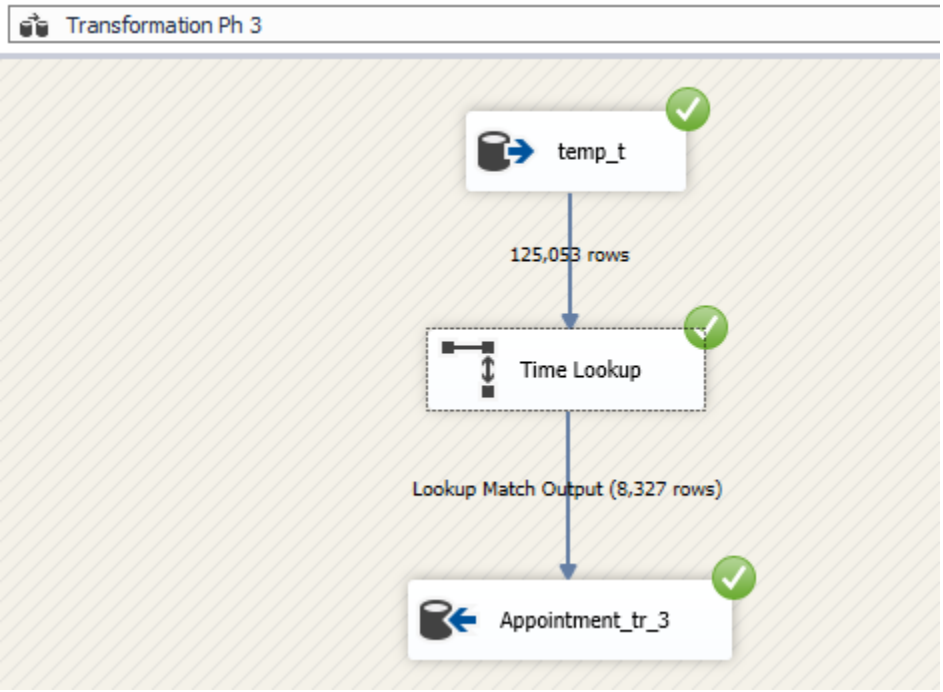
The goal of first phase of transformation is to merge provider's specialization in the appointment data.



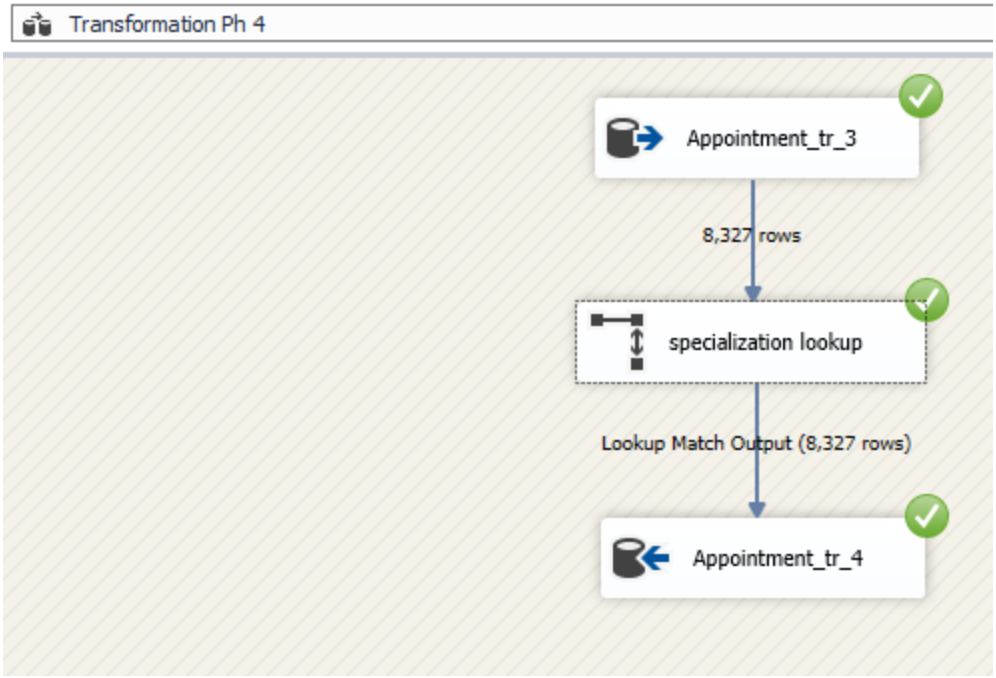
We then derive hours and minutes from the appointment date-time field, so that we can divide time into time-ranges for further assessment. We do this using derived column transformation with datepart function.



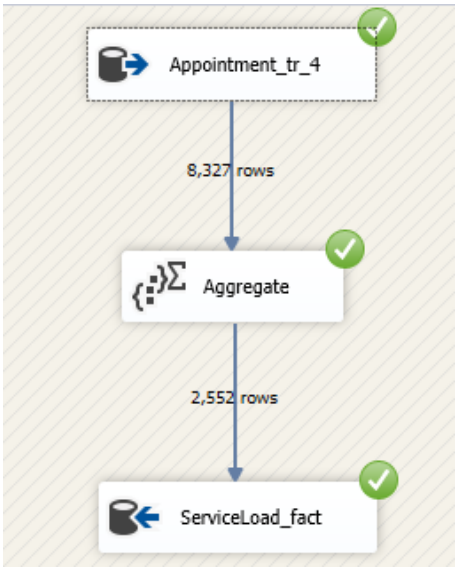
In the third phase, append the time category field to appointment data, as per defined time ranges. This is done using the Lookup transformation.



The final transformation is to replace specialization with specialization IDs. This is again, done using Lookup transformation, with assistance of the specialization dimension table.

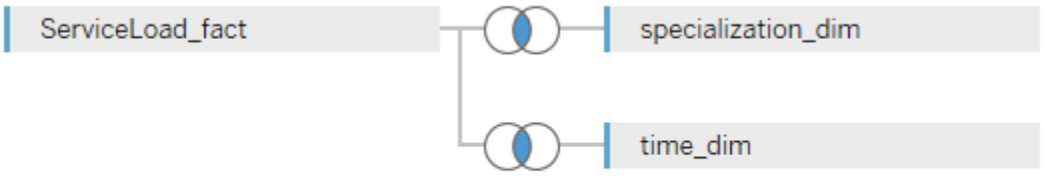


Lastly, populating data into the fact table:



DATA VISUALIZATION:

For data visualization, we are using Tableau. We connect Tableau with the SQL Server database, and use the fact and dimension tables specified in the data model.



Specialization and time dimensions are joined with the fat table, and tableau's inbuilt date hierarchy feature instead of the explicit date dimension.

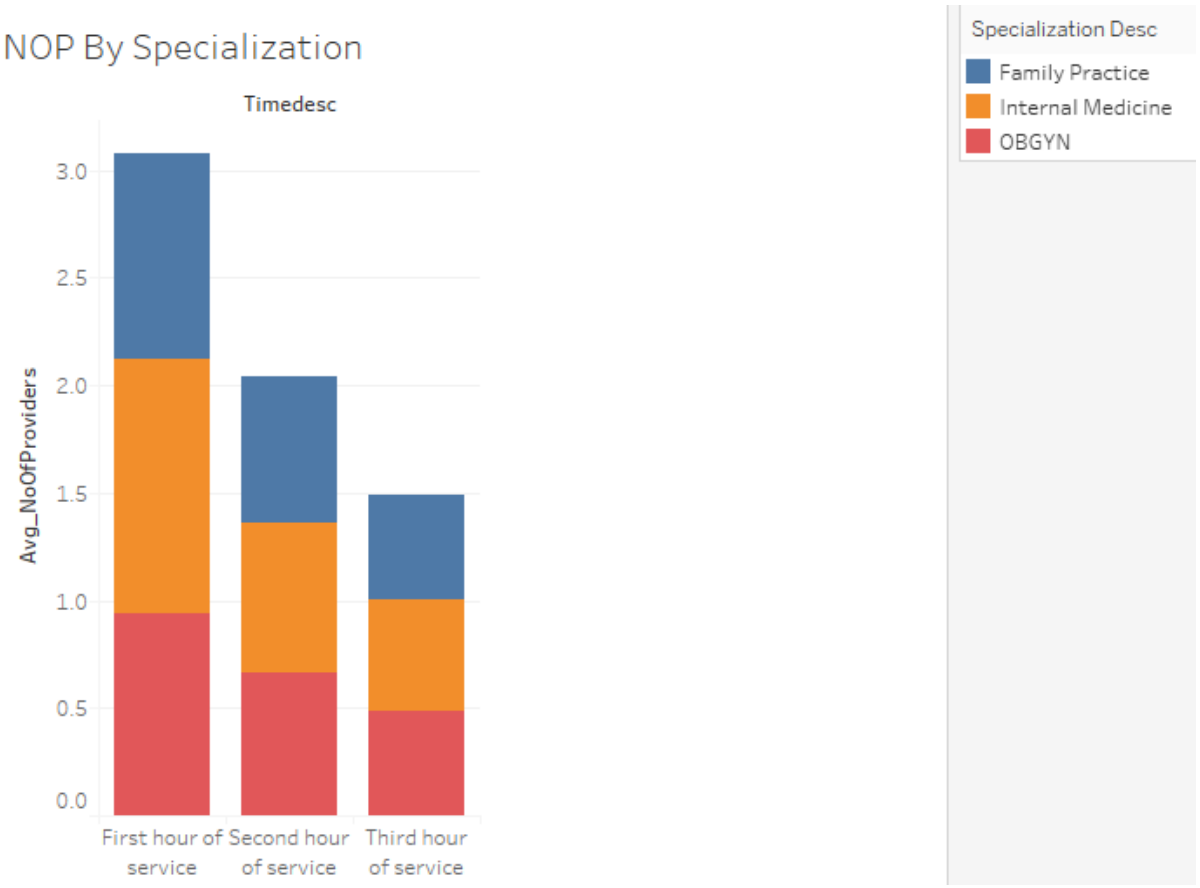
Since provider of one specialization may not be interchangeable with that of other specialization, we will keep the basic measure of looking at providers required in terms of the specialization.

The three specializations we are looking at in this dataset are:

1. OBGYN (gynecology)
2. Internal medicine
3. Family practice

The three hours of service for telephone appointment are:

1. First hour of service – 5:30 PM to 6:30 PM
2. Second hour of service – 6:30 PM to 7:30 PM
3. Third hour of service – 7:30 PM to 8:30 PM



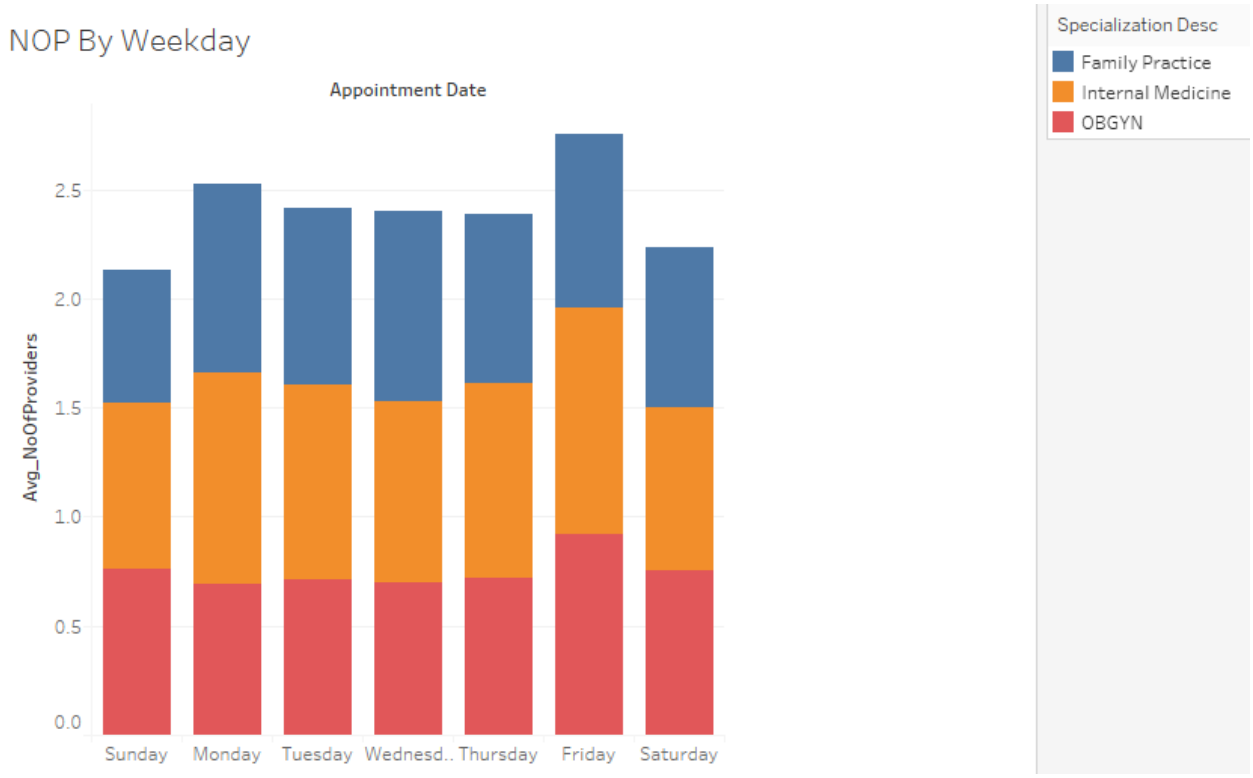
We can see that need for providers in all three specializations is decreasing as time increases, the maximum need being in the internal medicine department.

From this graph, we deduce that:

Hour of service	Internal medicine	OBGYN	Family practice	Total
First	2	1	1	4
Second	1	1	1	3
Third	1	1	1	3

Therefore, we need 4 providers in first hour, and 3 in the last two hours of service.

We can further assess this data in terms of day of the week as below:



Here, we can see that we may need 2 providers of internal medicine specialization only on Fridays, while on rest of the days, we need just 1.

INFERENCE

As per the results of analysis with SAS Base, and with the data warehouse setup, we conclude that:

a. It is best to assess number of providers in terms of specialization
b. We must also consider the weekday-wise load on telephonic service, while assessing the staffing required

The staffing requirement analyzed is as below:

For Fridays:

Hour of service	Internal medicine	OBGYN	Family practice	Total
First	2	1	1	4
Second	1	1	1	3
Third	1	1	1	3

For other days of the week:

Hour of service	Internal medicine	OBGYN	Family practice	Total
First	1	1	1	3
Second	1	1	1	3
Third	1	1	1	3

APPENDIX

Appointment Data – Data Dictionary

Field	Description
FACILITY	Abbreviation for where the appointment took place
DEPARTMENT	Abbreviation for the department where the appointment took place
PROVIDER_ID	Identifier for the Physician who saw or provided advice to the patient
APPOINTMENT_DATE	Date the appointment was scheduled to take place
APPOINTMENT_TIME	Time the appointment was scheduled to take place
PATIENT_ID	Identifier for the Patient who had the scheduled appointment
APPOINTMENT_TYPE	Type of appointment that was scheduled. Only two values: • Telephone Visit • In-Person Visit
BOOKING_DATE	Date the scheduled appointment was booked
BOOKING_TIME	Time the scheduled appointment was booked
SHOW_CODE	Indicator of whether the patient showed up for the appointment: • N/P – Patient did not show up for the scheduled appointment; Note – Check-In Date and Check-In Time will be null • Y – Patient appeared for the scheduled appointment
CHECKIN_DATE	Date the patient was registered for the appointment
CHECKIN_TIME	Time the patient was registered for the appointment