Rail Delay Infographics and Prediction System

Submitted in partial fulfilment of the requirements of
the degree of

BACHELOR OF TECHNOLOGY

National Institute of Technology Delhi

by

Pedapati Gnanadeep (141100041)
Shruti Bhadoriya (141100007)

Under the guidance of

Ms.Pooja Gupta
Assistant Professor, NIT Delhi

Department of Computer Science Engineering
NATIONAL INSTITUTE OF TECHNOLOGY DELHI
2017 - 2018

Acknowledgement

The satisfaction that is generated by the successful completion of a task would remain unfulfilled without mentioning people who have encouraged and guided us at every step towards the completion of the task. First, we would like to extend our sincere thanks to our project mentor **Ms. Pooja Gupta,** for her guidance and support. Throughout the project she always gave her valuable advice and suggestions for the betterment of our work.

This opportunity gave us a chance to sharpen our working methodology to a higher extent and to solve the problems in a better and easy way, so that it can be presented in a better and understandable manner.

Our heartfelt gratitude also goes to **Dr. Anurag Singh**, Head of the Department, Computer Science Department, National Institute of Technology Delhi, for providing us with the opportunity to avail the excellent facilities and infrastructure of the institute.

Shruti Bhadoriya(141100007)
Pedapati Gnanadeep(141100041)

Declaration

I declare that this written submission represents my ideas in my own words and where other's ideas or words have been included, I have adequately cited and referenced the original sources, I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea /data / fact/ source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not properly cited or from whom proper permission has not been taken when needed.

Name:Pedapati Gnanadeep
Roll no:141100041
Date: 30/11/2017

Ms. Pooja Gupta
Assistant Professor

Declaration

I declare that this written submission represents my ideas in my own words and where other's ideas or words have been included, I have adequately cited and referenced the original sources, I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea /data / fact/ source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not properly cited or from whom proper permission has not been taken when needed.

Name:Shruti Bhadoriya	Ms. Pooja Gupta
Roll no:141100007	Assistant Professor
Date: 30/11/2017	

ABSTRACT

Indian Railways owing to its cost-efficiency remains the most convenient and sought- after mode of transportation. One of the major problems encountered by the users is to make a decision about the train they would opt for based on the delay, seat availability and other factors. We have observed that the data that is present is scattered and reaching to the conclusion would often require intensive search thus consuming a lot of time. Our aim is to collect relevant information, analyse the data and create visual based report which are easier to comprehend and would expedite the decision making process. Along with the visual reports, with the data that we have obtained, we aim to develop an algorithm which would predict the best available train for the given route.

TABLE OF CONTENTS

List of Figures

List of Abbreviation

- **BI:** Business Intelligence
- **SQL:** Structured Query Language
- **T-SQL :** Transact Structured Query Language
- **CTE:** Common Table Expression
- **CSV :**Comma Separated Values
- **API:** Application Program Interface
- **SSIS :** SQL Server Integration Services
- **SSMS:** SQL Server Management Studio
- **WORA:** Write once, run anywhere
- **ETL:** Extract Transform and Load
- **DOM:** Document Object Model
- **HTML:** Hyper Text Markup Language
- **CSS:** Cascading Style Sheet
- **UI:** User Interface
- **JAR:** Java Archive
- **URL:** Uniform Resource Locator
- **NTES:** National Train Enquiry System
- **CRIS:** Centre for Railway Information Systems

1

Introduction

Indian Railways owing to its cost-efficiency remains the most convenient and sought-after mode of transportation. Indian railways is seventh largest employer in the world according to Forbes, but the question arises, in spite of the voluminous number of employees, does passengers have access to sufficient information to make their travel hassle free? Time and again Indian Railways have been critically questioned on the failure to handle the large runtime delays and the latent factors behind the cause.

1.1 Motivation

To cite the official figures in the year 2012-13,on any given day, there were 12617 trains running which carried 8.42 billion i.e. 842.1 crore passengers where one could find that 23.07 million passengers travelled on daily basis. The number of people adopting railways as their mode of commute is massive. Frequent delays and lack of information combined is a principal problem confronted by the passengers.

Other major problems encountered by the passengers is to make a decision about the train they would opt for, based on the delay, seat availability, priority, amount of traffic, connectivity and other additional factors not to be overlooked. We have observed that the data that is present accounts for only a past few days of record and reaching to the conclusion would often require intensive search thus consuming a lot of time.

The delay factor substitutes one of the major factor for selecting a particular train for the desired route, since it is not always possible to find a train to a given destination. In such scenarios, we then try to explore other options by searching the connecting train to the same destination via an intermediate stop. If we are not equipped with sufficient trend in previous delay, there are chances that you might miss the train

One of the major challenges that we faced was the collection of data, the historical data was not available directly at some source, from where it could be obtained. The website and sources currently operating have not yet fulfilled the objective that we aim to implement. We have thoroughly examined various sources and found them to be lacking in terms of volume of data, wieldy and straightforward representation of this data and prediction in the same regard.

Our aim is to collect relevant information, analyse the data and create visual based report which are easier to comprehend and would expedite the decision making process. The interactive visual reports would be of great assistance to the user, but at times the user might not be equipped with sufficient ability to apprehend the visual reports to arrive at conclusion. For such scenarios we aim to simplify the task further by making use of the data at our disposal to predict or suggest the train to the user measured on various metrics like delay, type of train etc.

1.2 Problem Formulation

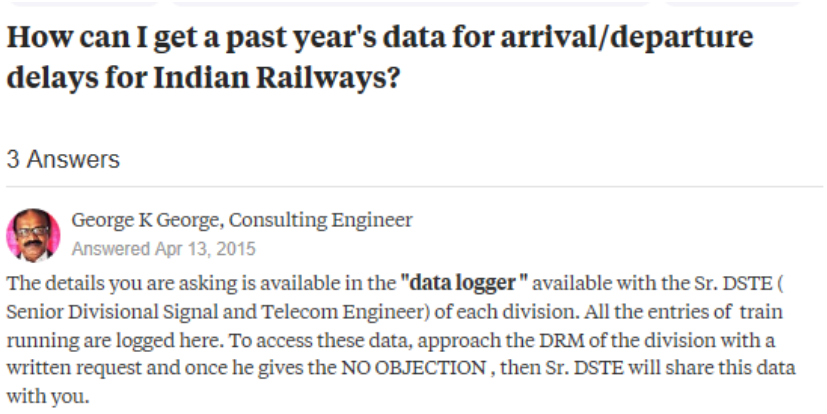
Usually for a given route, there are many trains that travel via the same route as desired by the person, but this necessarily doesn't mean that all the trains travelling on same route are going to take same time, some of them might get delayed frequently while for some other delay might be seldom. The main goal of the project is to develop application which would be helpful to the targeted audience in selecting better alternative from the available options. We intend to develop interactive visual based reports which can be customized as per the need using Power BI along with prediction of the best trains for a given route using machine learning and artificial intelligence

2

Literature Review

Official figures released by Indian Railways[1]

Investigation about the presence of historical data[2]



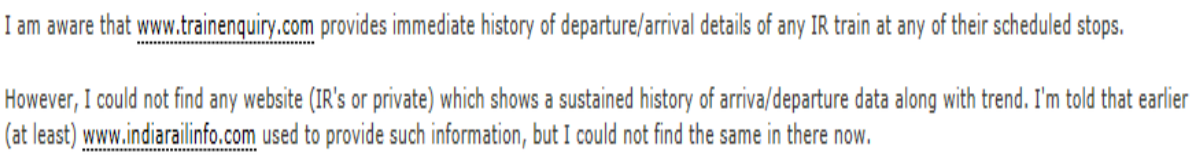
How can I get a past year's data for arrival/departure delays for Indian Railways?

3 Answers

George K George, Consulting Engineer
Answered Apr 13, 2015

The details you are asking is available in the **"data logger"** available with the Sr. DSTE (Senior Divisional Signal and Telecom Engineer) of each division. All the entries of train running are logged here. To access these data, approach the DRM of the division with a written request and once he gives the NO OBJECTION , then Sr. DSTE will share this data with you.

Figure 2.1: Data availability for Indian Railways



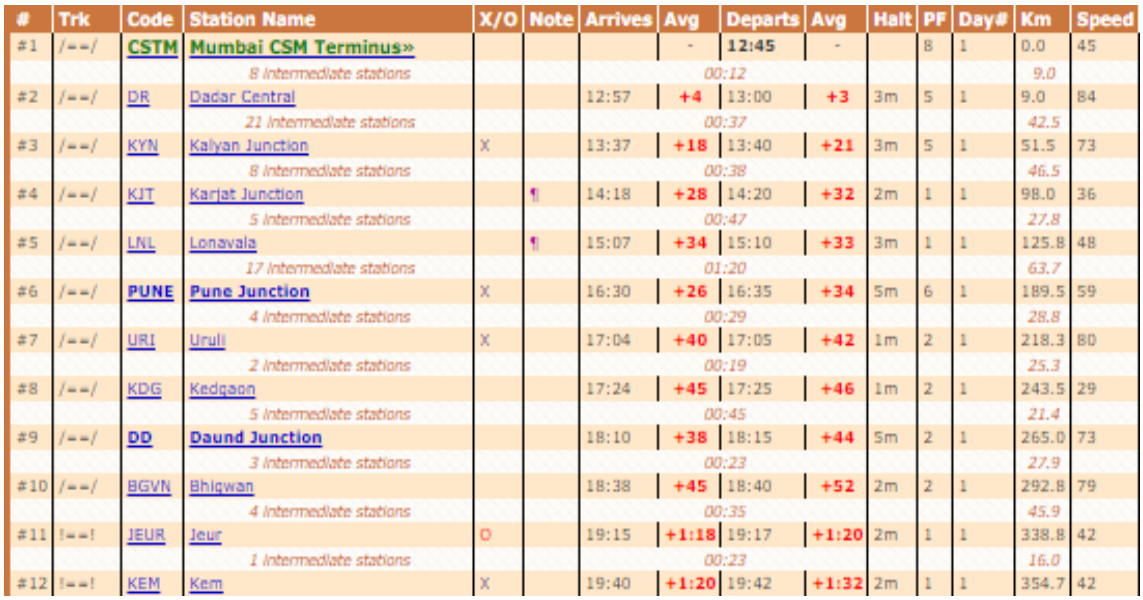
I am aware that www.trainenquiry.com provides immediate history of departure/arrival details of any IR train at any of their scheduled stops.

However, I could not find any website (IR's or private) which shows a sustained history of arriva/departure data along with trend. I'm told that earlier (at least) www.indiarailinfo.com used to provide such information, but I could not find the same in there now.

Figure 2.2: Customer Query for Delay Status_1

Project work with a slight resemblance with the project that we are currently implementing[3]

Data displayed on this site might not be precise. Moreover, it shows the data corresponding to average delay only.



#	Trk	Code	Station Name	X/O	Note	Arrives	Avg	Departs	Avg	Halt	PF	Day#	Km	Speed
#1	/--/	CSTM	Mumbai CSM Terminus»				-	12:45	-		8	1	0.0	45
			8 intermediate stations				00:12						9.0	
#2	/--/	DR	Dadar Central			12:57	+4	13:00	+3	3m	5	1	9.0	84
			21 intermediate stations				00:37						42.5	
#3	/--/	KYN	Kalyan Junction	X		13:37	+18	13:40	+21	3m	5	1	51.5	73
			8 intermediate stations				00:38						46.5	
#4	/--/	KJT	Karjat Junction		¶	14:18	+28	14:20	+32	2m	1	1	98.0	36
			5 intermediate stations				00:47						27.8	
#5	/--/	LNL	Lonavala		¶	15:07	+34	15:10	+33	3m	1	1	125.8	48
			17 intermediate stations				01:20						63.7	
#6	/--/	PUNE	Pune Junction	X		16:30	+26	16:35	+34	5m	6	1	189.5	59
			4 intermediate stations				00:29						28.8	
#7	/--/	URI	Uruli	X		17:04	+40	17:05	+42	1m	2	1	218.3	80
			2 intermediate stations				00:19						25.3	
#8	/--/	KDG	Kedgaon			17:24	+45	17:25	+46	1m	2	1	243.5	29
			5 intermediate stations				00:45						21.4	
#9	/--/	DD	Daund Junction			18:10	+38	18:15	+44	5m	2	1	265.0	73
			3 intermediate stations				00:23						27.9	
#10	/--/	BGVN	Bhigwan			18:38	+45	18:40	+52	2m	2	1	292.8	79
			4 intermediate stations				00:35						45.9	
#11	!--!	JEUR	Jeur	O		19:15	+1:18	19:17	+1:20	2m	1	1	338.8	42
			1 intermediate stations				00:23						16.0	
#12	!--!	KEM	Kem	X		19:40	+1:20	19:42	+1:32	2m	1	1	354.7	42

Figure 2.3: Snippet of IndianRail.info website

Provides the running instances corresponding to past two to three runs only.

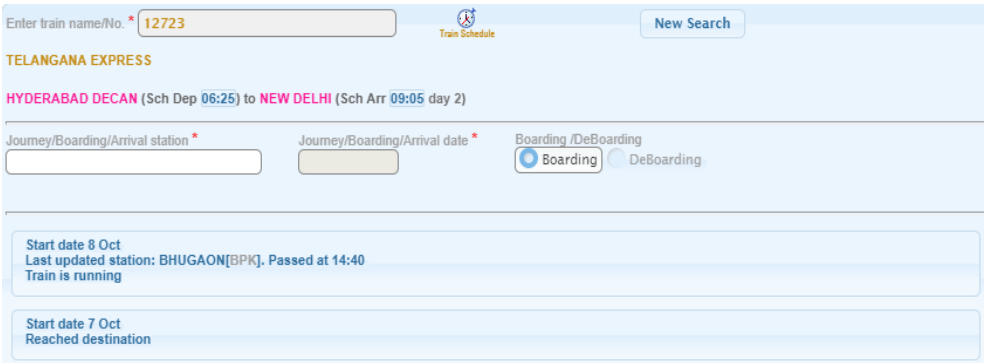


Figure 2.4: Snippet of NTES website

Provides number of times the train has been delayed but with no precise details.

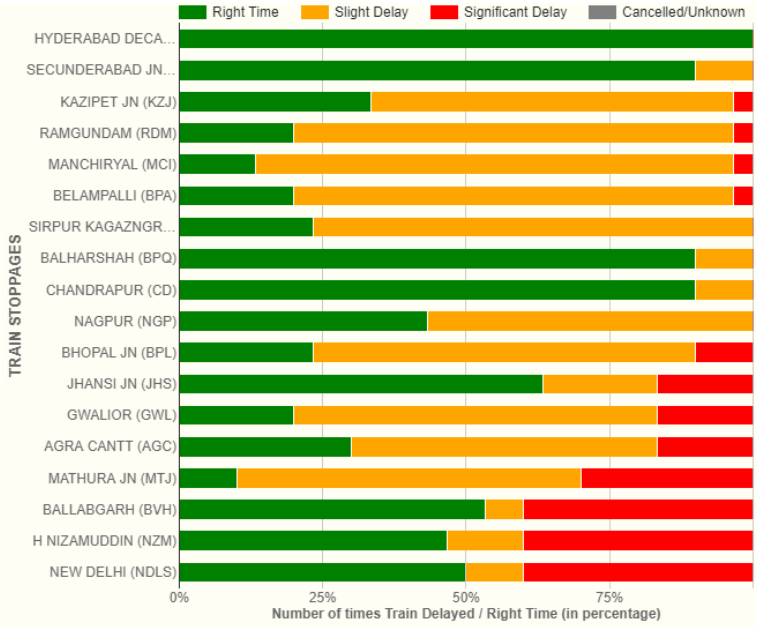


Figure 2.5: Snippet of etrain.info website

People enquiring on public forums about the delay trends and delay information[4]

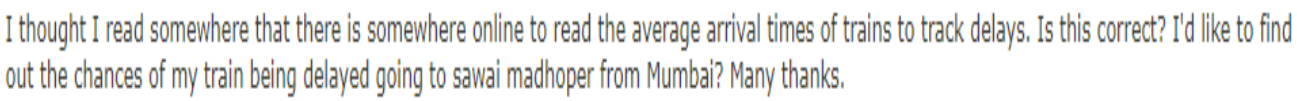
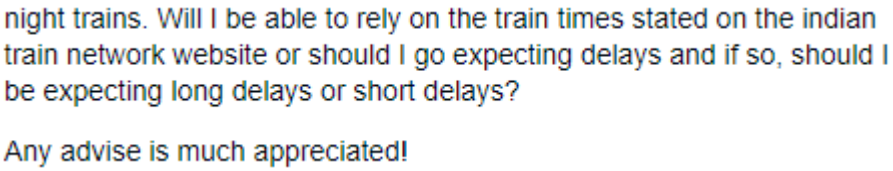

I thought I read somewhere that there is somewhere online to read the average arrival times of trains to track delays. Is this correct? I'd like to find out the chances of my train being delayed going to sawai madhoper from Mumbai? Many thanks.

Figure 2.6: Customer Query for Delay Status_2


night trains. Will I be able to rely on the train times stated on the indian train network website or should I go expecting delays and if so, should I be expecting long delays or short delays?

Any advise is much appreciated!

Figure 2.7: Customer Query for Delay Status_3

A research paper trying to investigate the reason behind the delay caused[5].

3

Tools and Technologies Used

Project Rail Info-Graphics and Prediction System would be implemented in two parts:

- First Phase is where the Info-Graphics for the relevant data would be implemented making use of business intelligence.
- While the second phase would utilise the data to make prediction using machine learning.

In order to implement the first phase the several tools, technologies and framework had been used. Following is the description of the same.

3.1 Selenium Web Driver

Selenium web driver was significantly used during the project, we mostly worked on the part of the project where the data pull process was to be automated. **Selenium** is a portable software-testing framework used for web applications. Selenium is used to automate browsers. Basically, it is used as tool for automating web applications for testing purposes, but it is not limited to that only. Web-based administration tasks which tends to be tedious and consuming manpower can be automated as well with the help of selenium.

Selenium web driver is mostly used in the cases where:

- You want to create robust, browser-based regression automation suites and tests.
- You want to scale and distribute scripts across many environments.

Selenium WebDriver is originally a collection of language specific-bindings which is used to drive browser - the way it is meant to be driven. Selenium Remote Control which has been officially deprecated is the predecessor of Selenium Web Driver.

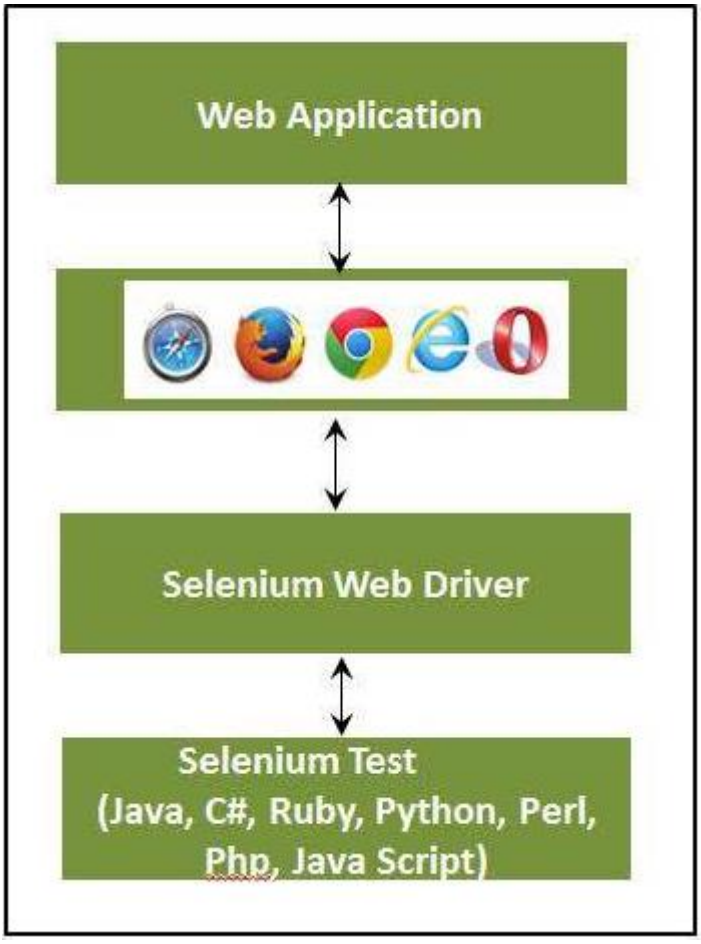


Figure 3.1: Web Driver Architecture

3.1.1 Selenium Locators

Locator is a command that tells Selenium Web Driver which GUI elements (say Text Box, Buttons, Check Boxes etc) its needs to operate on. Different types of locators have been mentioned below.

3.1.1.1 Locating by ID

As the ID's are unique for each element, so this is the most common method to locate elements.
Format: id=*elements' id*

3.1.1.2 Locating by Name

Using this method to locate the element is almost same as that of locating by ID except for, the prefix "name=" is used instead of "id=".
Format: name=*element's name*

3.1.1.3 Locating by Link Text

This is used in case where hyperlinks are present. In order to access the link text we select the string between anchor tags which is the hyperlink text and is prefixed by "link=".
Format: link=*link_text*

3.1.1.4 Locating by CSS Selector

CSS Selectors are string patterns which are useful in identification of elements using different combinations of HTML tag like id, attribute, class and name.

Commonly used combination of css selector is discussed below.

- Tag and ID
- Tag and class
- Tag and attribute
- Tag, class, and attribute
- Inner text

3.1.1.5 Locating by DOM (Document Object Model)

The Document Object Model (DOM), refers as to how the HTML webpage and its elements are structured. Selenium can use the DOM to access page elements.

There are four ways to access an element via DOM:

- getElementById
- getElementsByName
- dom:name
- dom:index

3.1.1.6 Locating by Xpath

XPath is the language used when locating XML (Extensible Markup Language) nodes. Since HTML can be thought of as an implementation of XML, we can also use XPath in locating HTML elements. It can access almost any element, even those without class, name, or id attributes.

There are two types of Xpath

- Absolute XPath
- Relative Xpath

3.1.2 Implicit wait and Explicit wait in Selenium

3.1.2.1 Why wait is required in Selenium?

Most of the web applications are developed using Ajax and Javascript. Different elements in a page may load at different time at the time when webpage is loaded.

This makes it difficult to locate elements as each element has different loading time and in case if the element is not found then the exception will be thrown “ElementNotVisibleException”. This problem is solved by the use of wait in the program which would wait as per the conditions defined by the programmer.

3.1.2.2 Implicit Wait

In implicit wait the web driver waits for a certain period of time before throwing “NoSuchElemenExcpetion” or “ElementNotVisibleException”.By default the waiting time for the driver is set to 0. Once the time to wait is set then the web driver wait for that amount of time before throwing any exception. Implicit wait condition holds true for the entire time for which the browser is open. Therefore any search for elements on the page would wait for the time for which implicit wait has been imposed.

3.1.2.3 Explicit Wait

Explicit wait is defined as the wait time for which the web driver waits for a certain condition to happen before going further in the code. One of the type of explicit wait is where you define for how much time will the browser will wait irrespective of the any condition, it is Thread.sleep(), this can make program slow. There are some methods predefined which help in writing code that would wait only for as long as requirement. WebDriverWait along combination of ExpectedCondition is one of the way in which the task can be accomplished.

Some of the expected condition are listed as follows:

- elementToBeClickable(By locator) : waits until an element is visible and enabled
- elementToBeSelected(WebElement element): waits until an element is selected
- presenceOfElementLocated(By locator) : waits until presence of an element
- textToBePresentInElement(By locator, String text): waits until specific text is present in the an element.

3.1.3 Why use selenium over other conventional scraping tools?

- Selenium web driver is capable of pulling data from web pages with dynamic content with much ease.
- With the help of selenium you could use it to send the value to a website i.e. fill a field or an entire form through your program, unlike other tools which are incapable of doing this.
- Most of the conventional data scraping tool operate on Internet explorer, while with selenium you can overcome this limitation, since it is operable on a variety of browsers like internet explorer, google chrome, mozilla firefox, opera etc.

3.2 Transact-SQL

Transact SQL or better known as T-SQL is a set of programming extensions from Sybase and Microsoft that add several features to the Structured Query Language (SQL), including transaction control, exception and error handling, row processing and declared variables.

T-SQL is used to interact with relational databases. T-SQL expands on the SQL standard to include procedural programming, local variables, various support functions for string

processing, date processing, mathematics, etc. and changes to the DELETE and UPDATE statements.

Transact-SQL is central to using Microsoft SQL Server. All applications that communicate with an instance of SQL Server do so by sending Transact-SQL statements to the server, regardless of the user interface of the application.

T-SQL's transaction and journaling system, handles just about anything - including a power cycle or hardware failure - without database corruption, and if something gets messed up it fixes it automatically.

T-SQL support CTE. A common table expression (CTE) can be thought of as a temporary result set that is defined within the execution scope of a single SELECT, INSERT, UPDATE, DELETE, or CREATE VIEW statement. A CTE is similar to a derived table in that it is not stored as an object and lasts only for the duration of the query. Unlike a derived table, a CTE can be self-referencing and can be referenced multiple times in the same query. It simplifies complex queries and most importantly enables you to use recursion.

T-SQL is the SQL dialect that the product SQL Server is using. Transact-SQL is central to using SQL Server. All applications that communicate with an instance of SQL Server do so by sending Transact-SQL statements to the server, regardless of the user interface of the application. SQL Server is tied to Transact-SQL (T-SQL), an implementation of SQL from Microsoft that adds a set of proprietary programming extensions to the standard language

3.2.1 Difference between T-SQL and SQL

- T-SQL adds a number of features that are not available in SQL. This includes procedural programming elements and a local variable to provide more flexible control of how the application flows.

- A number of functions were also added to T-SQL to make it more powerful; functions for mathematical operations, string operations, date and time processing, and the like.
- These additions make T-SQL comply with the Turing completeness test, a test that determines the universality of a computing language. SQL is not Turing complete and is very limited in the scope of what it can do.
- Another significant difference between T-SQL and SQL is the changes done to the DELETE and UPDATE commands that are already available in SQL. With T-SQL, the DELETE and UPDATE commands both allow the inclusion of a FROM clause which allows the use of JOINs. This simplifies the filtering of records to easily pick out the entries that match a certain criteria unlike with SQL.
- SQL is non-procedural language since it deals with what data to be extracted. Whereas T-SQL is procedure language since it deals with what data to be executed and how it should be displayed.
- The SQL queries in SQL are submitted individually to the database server, while in T-SQL the batch program is written where in all commands are submitted to the server in a single go.

3.3 Power BI

Power BI is a cloud-based business analytics service from Microsoft that empowers anyone to experience any data – structured or unstructured – via simple drag-and-drop ease. Unlike many other dashboard solutions, Power BI can render live dashboards with moving charts and continuously updated visualizations for monitoring real-time streams from supported data sources.

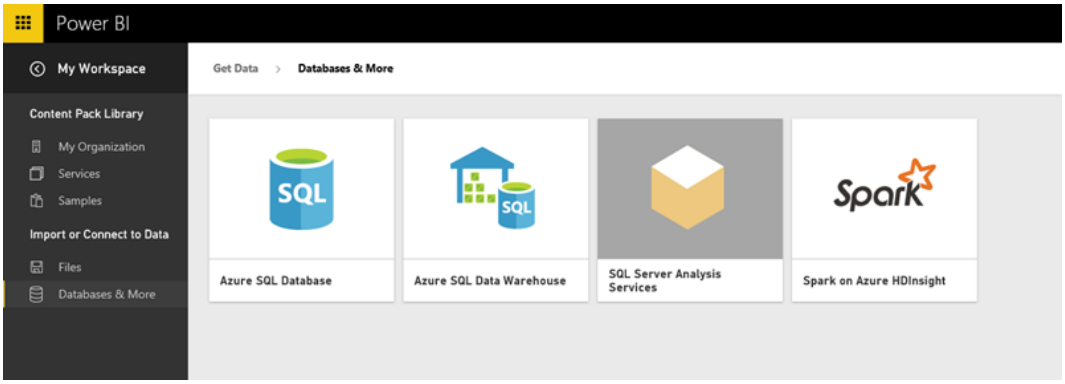


Figure 3.2: Power BI Connection

As shown in figure 3, we can use the Power BI REST API with any data source or Azure Streaming Analytics to render live Power BI Dashboards automatically. Alternatively, you can get near real-time analytics using simple "direct connect" data sources such as Analysis Services, Azure SQL Database, Azure SQL Data Warehouse or Spark with Power BI Reports.

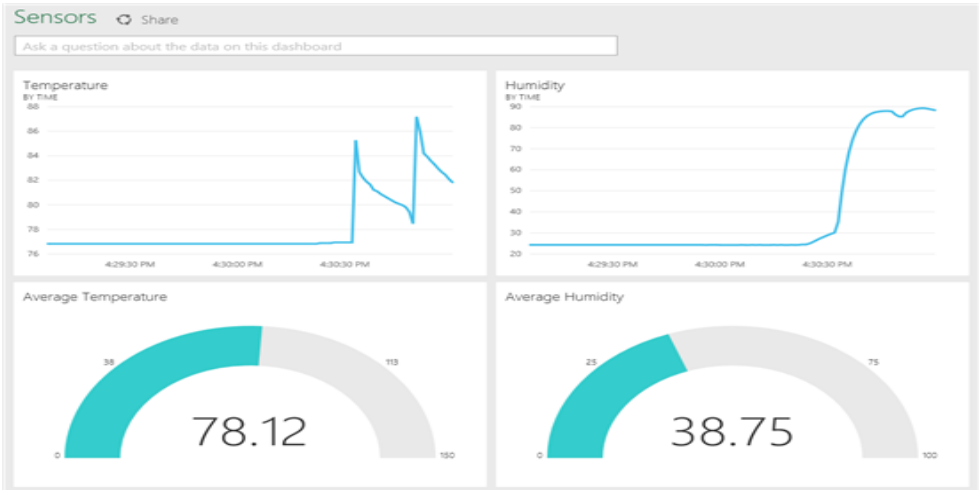


Figure 3.3: Power BI report

As shown in fig 3, We can get any type of visuals from Power BI ,Microsoft PowerBI tools makes the reporting part so easy that we can create our reports in short period of time with so much attractive and correct and real time data, milliseconds refresh data we can retrieve with our job scheduling.

You can import the data which is required to make the power BI report from various sources. It can be an excel sheet, CSV (comma separated value) file, database on your local machine, data in cloud etc.

Power BI has Q&A feature to explore your data using intuitive, natural language capabilities and receive answers in the form of charts and graphs. Q&A is different from a search engine -- Q&A only provides results about the data in Power BI. Data visualizations (aka visuals) helps us to interact with data to find business insights.

3.3.1 Power BI visualizations

The types of visualization generally used in Power BI reports are:

1. Bar Chart

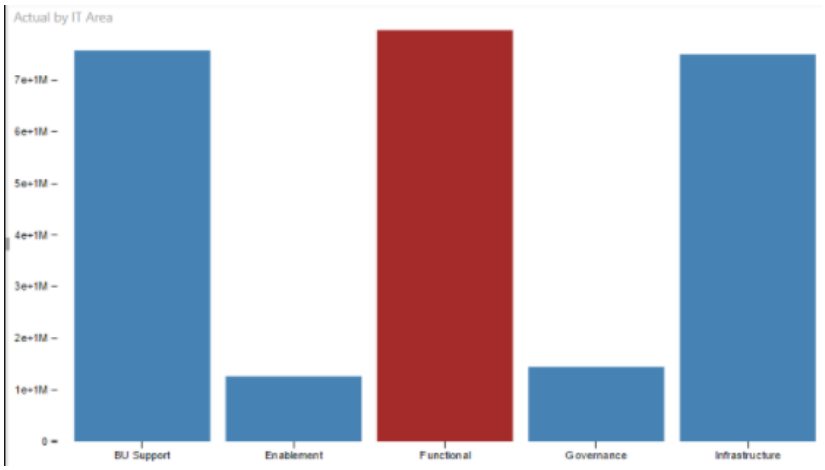


Figure 3.4: Bar chart

2. Stacked Bar Chart

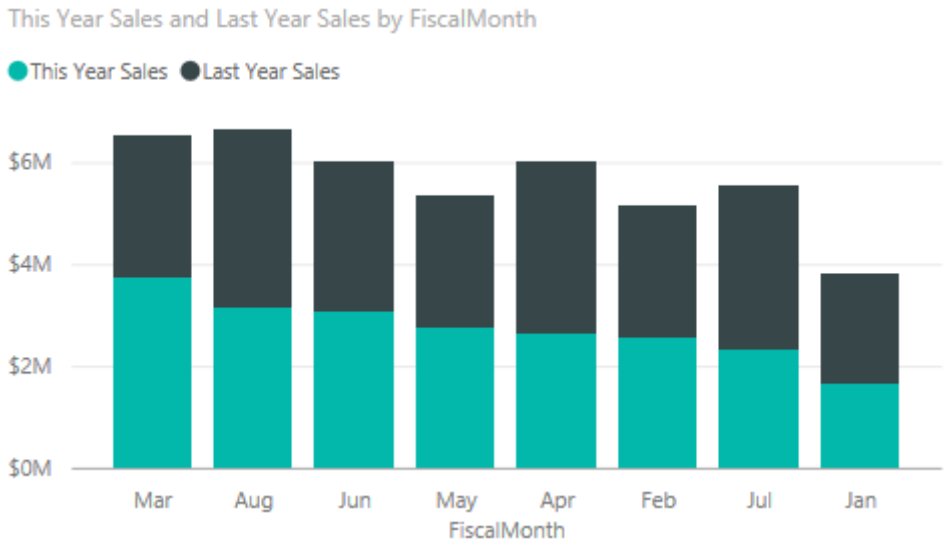


Figure 3.5: Stacked Bar Chart

3. Line Chart

Figure 3.6:Line Chart

4. Slicer

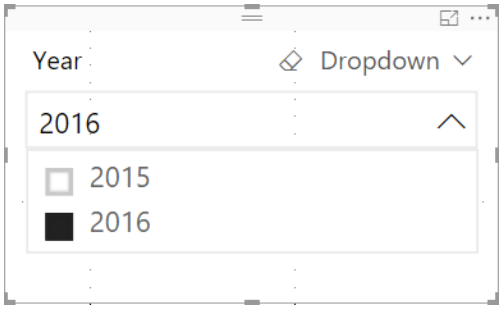


Figure 3.7: Slicer

5. Pie Chart

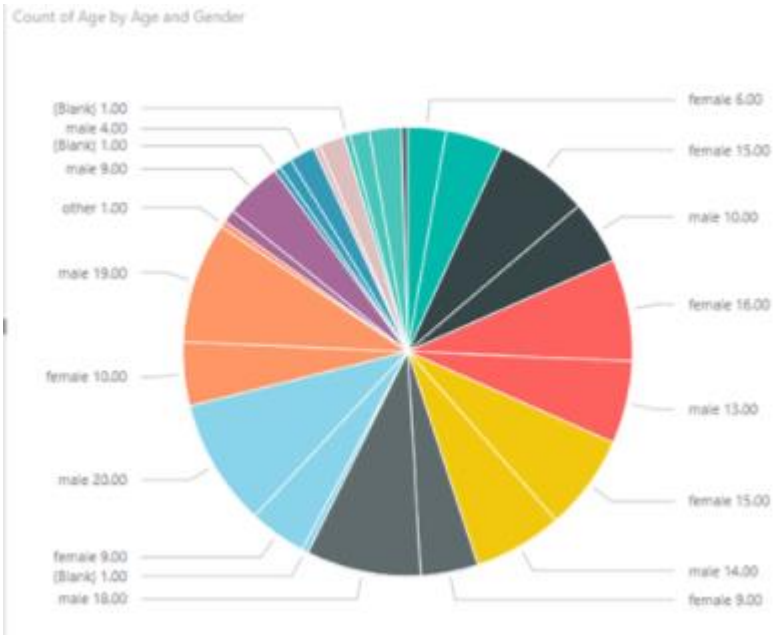


Figure 3.8: Pie Chart

3.4 Data Cubes

Data cubes are multi-dimensional database, in other word they are extension to the two dimensional relational table. Cubes can be three dimensional, four dimensional etc depending on the requirement.

OLAP (Online Analytical Processing) make use of cubes since they provide deeper insights to the data. The data is present in dimension tables and the fact table then summarizes the data in dimension table for a given attribute, so we can say that fact table stores the data in aggregated form which are often called measures. MDX (multidimensional expressions) queries are written to extract data from cubes.

Since the data is stored in aggregated form, this is often useful for the analyst to perform data analysis, establish trends, measure performance. We don't perform calculation for the data in cube, since the data in the cube has already once been analysed, processed and aggregated into the form of the cube. This implies that the data in the cube is historical and not dynamic and real-time data. These characteristics of a data cube perfectly aligns with the reporting purposes where millions of records are to be processed at a time. A data cube is a single entity where all the data for analytics purpose could be found without having to refer to different relational tables and their relations.

3.4.1 Multidimensional Model

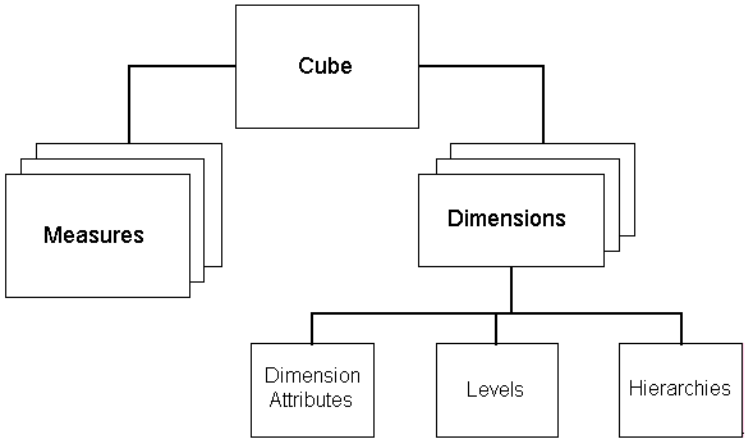


Figure 3.9: Snippet of contents of Data Cube

As can be seen from the figure above, multidimensional model consists of dimensions and measures (which are present in fact table).

Dimensions has attribute that are unique for a feature that categorizes the data. While in fact tables, only the integer aggregated values i.e. measures and foreign keys are present. Fact table contain data that is relevant in decision making process.

3.4.2 Schema

There are different types of implementation of the data cubes:

- Star Schema Model
- Snow Flake Schema
- Fact Constellation

3.4.2.1 Star Schema

It is called star schema because this schema is similar to a star where all the relationship originates from a single point source. It is similar to inner join between a fact table and multiple dimension tables. Dimension has a primary key while the fact has a foreign key. The point to be noted in star schema is that the dimension table are not related to one another, they all are only related to fact table. Performance in this schema is highly optimized.

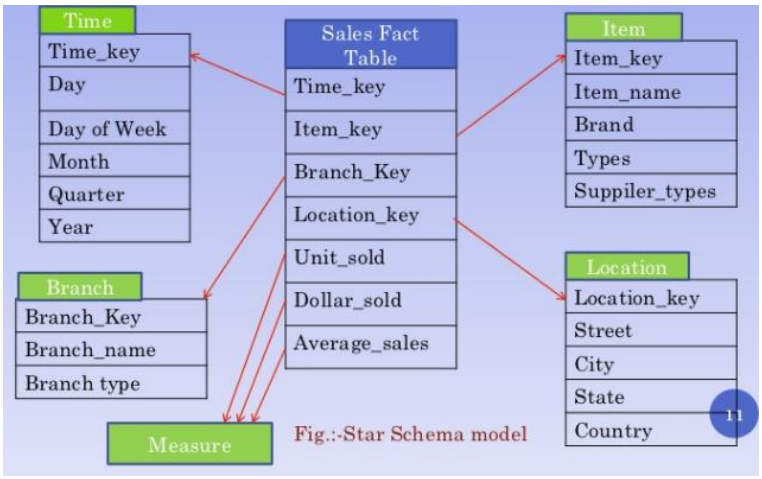


Figure 3.10: Snippet of Star Schema model

3.4.2.2 Snowflake Schema

This is similar to star schema except for the fact that in this the dimension table are normalized so that the hierarchy is created and to get rid of redundancy and anomalies.As there is no redundancy in data, the denormalized schema then occupy less space on disk. Due to normalization, the queries to extract data from snow flake model becomes quite complex when compared with queries in star-schema model.

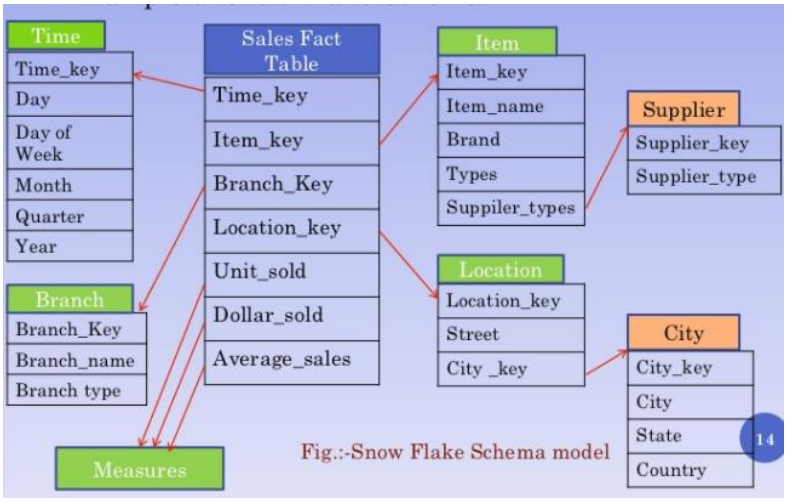


Figure 3.11: Snippet of Snow Flake Schema model

3.4.2.3 Fact Constellations

As the name suggests, it is a set of fact tables which have shared dimensions among themselves. Multiple star schema can share a fact constellation. This schema is more complex as compared to the other two schema, since it has more than one fact table with shared data which makes it hard to manage data and data relationship. Several aggregation leads to enhanced complexity.

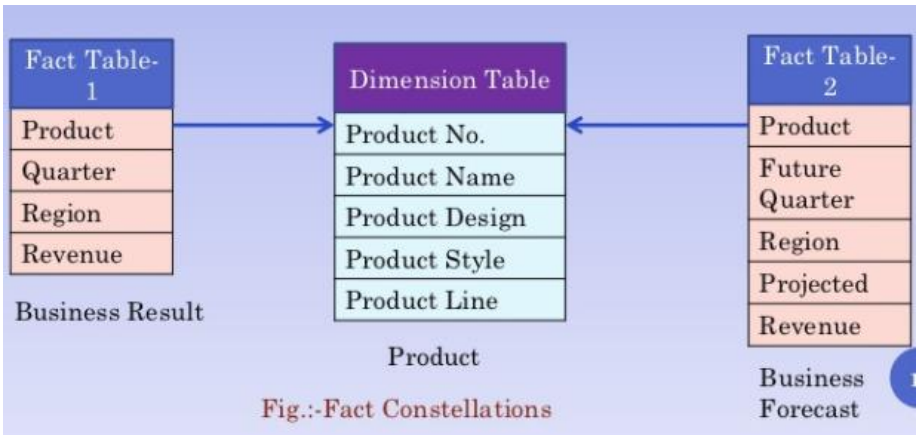


Figure 3.12: Snippet of Fact Constellation Schema model

3.5 Java

Java is high level programming which is concurrent, object-oriented, and has been designed specifically to have minimum possible implementation dependencies. It is "write once, run anywhere" (WORA), application that is being used by developers which means the compiled Java code can run on all platforms which support Java without the need for recompiling the code every time.

3.6 SSMS(SQL Server Management Studio)

SQL Server Management Studio (SSMS) is a software application first launched with Microsoft SQL Server 2005 that is used for configuring, managing, and administering all components within Microsoft SQL Server. This tool has both script editors and graphical tools which work with objects and features of the server.

4

Into the Project

As discussed earlier, the first phase of our project deals with displaying visual reports of delays for various train for a period of time to facilitate the customers in choosing the best opt train based on delay criteria.

The following are the key steps involved in the project

- **Collection of data**
- **Database Design**
- **Data cleaning**
- **Normalization of Schema**
- **Building Multidimensional Data Cubes**
- **Generating visual based reports**

In simpler terms, it is known as ETL(Extract-Transform-Load).

4.1 Collection of data

In Data Analysis, the major primary step involved is Data Collection. There are many factors which are essential in this process. Some of them are:

- What will be the sources of Data collection?
- Approach and method to collect data.
- How much amount of data must be collected?
- Is the data collected task relevant?
- Removing redundant data
- Data Validation

Determining the accurate data source is vital step the in data collection because it ultimately affects the overall performance of the project .Since we are dealing with Indian Railways, all the official information is provided by CRIS. National Train Enquiry System one of the official portal of CRIS has been chosen for source of the data collection.

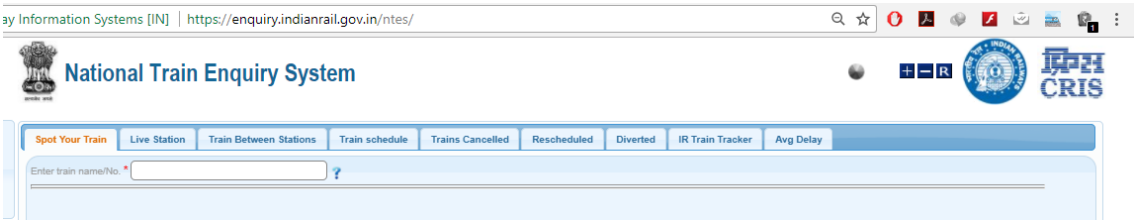
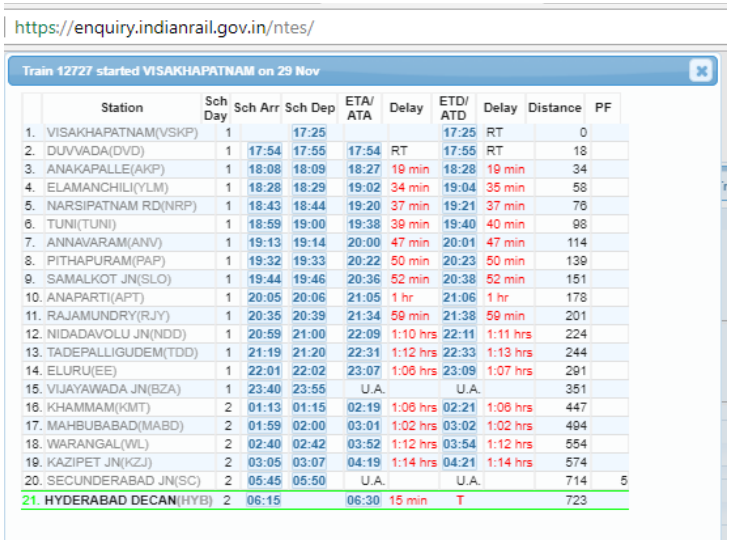


Figure 4.1: NTES Website



	Station	Sch Day	Sch Arr	Sch Dep	ETA/ ATA	Delay	ETD/ ATD	Delay	Distance	PF
1.	VISAKHAPATNAM(VSKP)	1		17:25			17:25	RT	0	
2.	DUVVADA(DVD)	1	17:54	17:55	17:54	RT	17:55	RT	18	
3.	ANAKAPALLE(AKP)	1	18:08	18:09	18:27	19 min	18:28	19 min	34	
4.	ELAMANCHILI(YLM)	1	18:28	18:29	19:02	34 min	19:04	35 min	58	
5.	NARSIPATNAM RD(NRP)	1	18:43	18:44	19:20	37 min	19:21	37 min	76	
6.	TUNI(TUNI)	1	18:59	19:00	19:38	39 min	19:40	40 min	98	
7.	ANNAVARAM(ANV)	1	19:13	19:14	20:00	47 min	20:01	47 min	114	
8.	PITHAPURAM(PAP)	1	19:32	19:33	20:22	50 min	20:23	50 min	139	
9.	SAMALKOT JN(SLO)	1	19:44	19:46	20:36	52 min	20:38	52 min	151	
10.	ANAPARTI(APT)	1	20:05	20:06	21:05	1 hr	21:06	1 hr	178	
11.	RAJAMUNDRY(RJY)	1	20:35	20:39	21:34	59 min	21:38	59 min	201	
12.	NIDADAVOLU JN(NDD)	1	20:59	21:00	22:09	1:10 hrs	22:11	1:11 hrs	224	
13.	TADEPALLIGUDEM(TDD)	1	21:19	21:20	22:31	1:12 hrs	22:33	1:13 hrs	244	
14.	ELURU(EE)	1	22:01	22:02	23:07	1:06 hrs	23:09	1:07 hrs	291	
15.	VIJAYAWADA JN(BZA)	1	23:40	23:55	U.A.		U.A.		351	
16.	KHAMMAM(KMT)	2	01:13	01:15	02:19	1:06 hrs	02:21	1:06 hrs	447	
17.	MAHBUBABAD(MABD)	2	01:59	02:00	03:01	1:02 hrs	03:02	1:02 hrs	494	
18.	WARANGAL(WL)	2	02:40	02:42	03:52	1:12 hrs	03:54	1:12 hrs	554	
19.	KAZIPET JN(KZJ)	2	03:05	03:07	04:19	1:14 hrs	04:21	1:14 hrs	574	
20.	SECUNDERABAD JN(SC)	2	05:45	05:50	U.A.		U.A.		714	5
21.	HYDERABAD DECAN(HYB)	2	06:15		06:30	15 min	T		723	

Figure 4.2:Example of Train Delay Details

But as observed and stated under literature review, Since the data required is not available for download and there is no other means to collect data from CRIS, we handled this problem using **SELENIUM.**As discussed above Selenium is one of the most powerful tool to automate web browser across various platforms without any human interaction, it has been used to collect the grid data periodically. Selenium used the html element id to capture data from them. Java has been used as a wrapper language for this process

4.1.1 Scheduling the data pull:

The developed java program is compiled and turned into a executable jar .This jar file is executed through batch file with the help of Windows Scheduler which is scheduled to run daily. Generally daily data pull must be run once, but there might be server and internet issues, it has been scheduled to run twice so that we don't miss any data.

4.1.2 Multithreading:

In order to reduce the data pull time, multithreading has been used,so that many threads can run at once dumping the data from NTES website simultaneously.This helped us to reduce the overall time to less than 50 percent.

4.1.3 Error Logging:

Every time when the data pull process is initiated a log file will be generated with time stamp. It contains all the details of exceptions that might occur due to several reasons like database connectivity, incorrect SQL syntax, could not find element exception. This log files provide the necessary means to identify and rectify the error so that the automation process is uninterrupted.

4.1.4 Retry mechanism:

Many times the data source server might be overloaded and could not run the data query or there might be error in server else may be internet connectivity issue, then the automated process fails in capturing the data. Hence a proper retry mechanism must be provided which waits for a specific time and sends a request back to server to run the query. The number of times a process must retry depends on the priority of the data and other factors.

4.2 Database Design and Normalization:

The next step involved is proper database schema design. The database consists of train list table which contains list of some trains which we targeted at, Train Schedule table, Delay Status Table. Appropriate data types have been defined according to the size of variables to accommodate the values. The train list contains different types of trains like mail, superfast, rajdhani with travel time up to 40 hrs, so as to make the problem more generalized. Schema has been normalized up to 3NF.

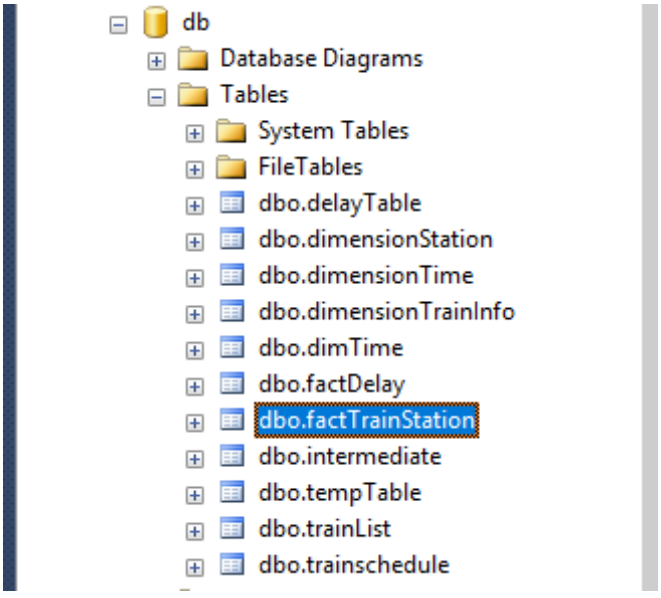


Figure 4.3:Database Schema

4.3 Data Cleaning and Formatting:

Data cleaning is the process of detecting and correcting (or removing) corrupt or inaccurate records from a record set, database and refers to identifying incomplete, incorrect, inaccurate or irrelevant parts of the data and then replacing, modifying, or deleting the dirty or coarse data. Accuracy, Completeness and Consistency are some factors to be considered. Removing duplicate data is also given much importance .The data obtained contains a lot of missing data and inconsistent data. Hence we applied some techniques to replace the missing values with some relevant data. One such technique is to replace the current delay value at a station with the previous station delay. Besides it included conversion of different format of data into a single common For example ,the delay value is presented in different formats like hr:mm/day:hr:mm/mm,which we have converted to minutes format for efficient comparison. Removing some defined constants which might be unfamiliar to the user was also a part of formatting.

Figure 4.4:Example of Missing values

4.4 Multidimensional Data Cubes

The data collected is too large, so using the database directly for providing the analytical services will simply degrade the performance of the project. So the optimal solution is to use data cubes which are defined by collection of facts and dimensions. Every dimension represents a new attribute in the database and the cells in the cube represent the measure of interest. These facts and dimensions are created using views, joins and stored procedures which are used to dump data into cube from database tables. It also included Dimensional Integrity check and cube refresh status. The cube will be refreshed periodically from the database.Primary keys and

foreign keys are used to frame the relationship between facts and dimensions.The major facts and dimensions that are included are as follows:

- Fact_Delay
- Dim_Time
- Dim_station
- Dim_trainno

Cube that was made corresponding to above dimension and facts:

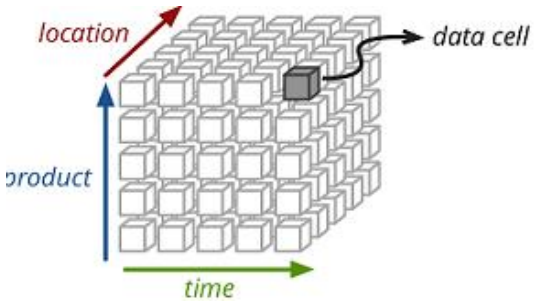


Figure 4.5: Multidimensional Data Cube.

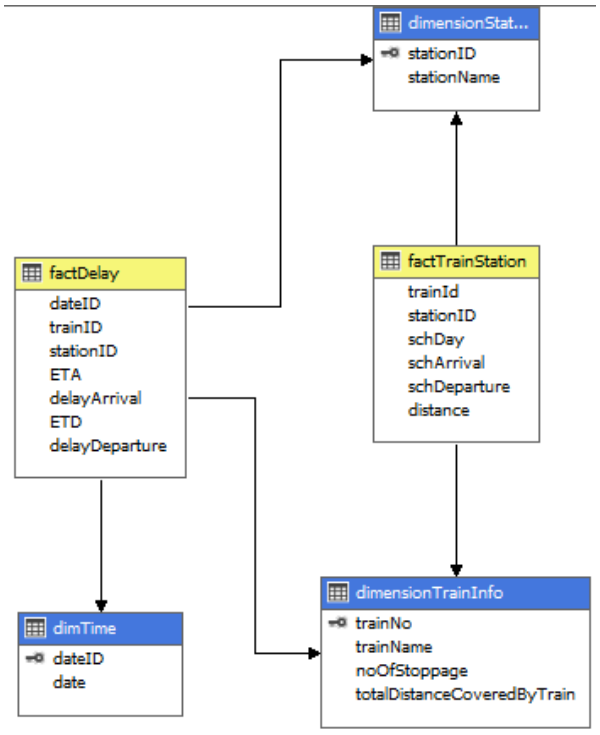


Figure 4.6: Snippet of Multidimensional Data Cube Schema

4.5 Backup:

Every time when a new data is inserted into the database, User defined procedures are triggered to create a copy of data into a backup file. Program changes are committed in GitHub.

4.6 Visual Reports

After the cube has been deployed, a live connection is made between the PowerBI and cube. Various visuals have been chosen appropriately. In order to show the data analysis between various data sources we have been using slicer to compare data. At the current scenario, we presented two visual reports, one displaying delay statistics for a given train using some filters and another visual for comparison between two or more trains. Coloured cards are displayed which represent the number of times a train has been delayed maintaining level of granularity in delay.

5

Result and Discussion

The project has been implemented in 3 parts, the final phase which is the visual report implementation contains in it the result of the analysis done.

5.1 Result

Following cards have been used in the Power BI report:

- The card corresponding to green colour are the trains which had delay of less than or equal to 30 minutes.
- While the card in yellow colour has record pertaining to the trains with delay greater than 30 minutes and less than 90 minutes.
- For delay greater than 90 minutes red coloured card is used.

Following are the output of some of the results obtained from the power BI report:

- **Trends shown for individual trains for a particular station and date range**

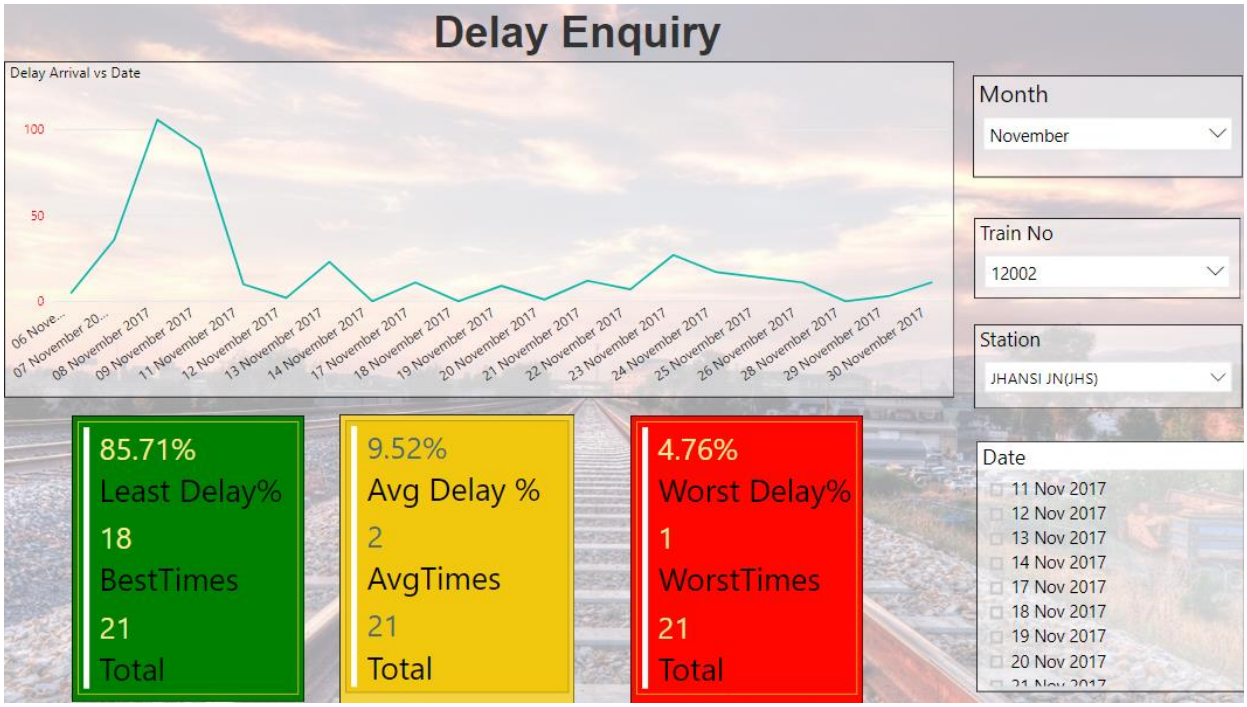


Figure 5.1: Delay and Delay percent for train no 12002 for Jhansi station is shown for entire month of November

- **Relative trend chart where user can compare delays between several trains for a particular station**

Figure 5.2: Relative trend in delay shown for Agra Cantt Station for different trains

5.2 Discussion

The project that has been developed currently taking into account the data corresponding to 27 trains and 242 stations. The data points on which we have carried out the analysis is currently more than 10000.The actual time taken for pulling data for 27 trains on everyday basis is about 30-35 minutes, but we have added multithreading to the selenium script so that the effective time taken to pull the data by 3 threads simultaneously is now about 10-12 minutes.

In future we intend to implement the same for up to 100 trains so that the data is diverse and can be used extensively for getting general trend for major stations. The project is yet under progress and we further aim to utilise the application of machine learning to predict the best trains that travel via given route. This would simplify the task of passengers and travellers and would be of great utility.

6

Summary and Conclusions

6.1 Summary

The project has been a great learning experience, tools and technologies that we have used were cutting edge and helped us to gain better insight into principles and working of business intelligence. Starting with collection of data followed by database design, designing multidimensional cubes, then making use of business intelligence to create interactive visual reports, all these phases were challenging yet interesting to work on. At every phase, wherever possible, optimization was done in order to achieve better results, like while pulling data, multithreading was done so that parallel processing could be done which would lead to decrease in time taken to pull the data and other similar improvisation over obtained results applied. Until now the key technologies and frameworks that we have used includes selenium web driver, transact-sql, multidimensional cubes, Power BI, SSMS, SSAS, java.

The next phase of the project would be to use the data to predict best train for a given route with the help of machine learning.

6.2 Conclusion

- The visual reports developed are interactive, user-friendly and customizable.
- The user can view the delay trends for a particular station for given range of dates.

- The user can view both the absolute and comparative delay analysis of one or more trains.
- Data at our disposal is accurate as data is collected from the official website of Indian Railways, CRIS (Central Rail Information System).

6.3 Scope for Future Work

Few of the areas where further work may be carried out on the project is:

- Increasing the number of trains to diversify the dataset.
- Additional number of features for comparison which are secondary reasons but could be used for analysis.
- The Power BI reports can have user defined flags for delay percent rather than static values.
- Enhancing the level of granularity.
- Machine learning can be used further for given data for prediction purpose.

Reference Links for Literature Review

[1] Official figures released by Indian Railways
Link:

http://www.indianrailways.gov.in/railwayboard/view_section.jsp?id=0%2C1%2C304%2C366%2C554%2C1451%2C1454&lang=0

[2] Investigation about the presence of historical data

Link:

https://www.quora.com/How-can-I-get-a-past-years-data-for-arrival-departure-delays-for-Indian-Railways

http://www.indiamike.com/india/indian-railways-f10/indian-railway-ontime-performance-train-delay-t139389/

https://www.quora.com/How-can-I-get-statistical-data-related-to-Indian-railways-of-last-5-or-10-years

[3] Project work with a slight resemblance with the project that we are currently implementing

Link:

https://indiarailinfo.com/

https://enquiry.indianrail.gov.in/ntes/

https://etrain.info/

[4] People enquiring on public forums about the delay trends and delay information

Link:

http://www.indiamike.com/india/indian-railways-f10/average-train-delays-t144109/

https://www.tripadvisor.in/ShowTopic-g293860-i511-k7331190-Do_Indian_trains_run_on_time-India.html

[5] A research paper trying to investigate the reason behind the delay caused

Link:

http://dev3.acmdev.org/posters/dev-final15.pdf