

Low Level Design

Financial Data Analytics

Written By	Prateek Lapalika
Document Version	0.1
Last Revised Date	21-11-2022

DOCUMENT CONTROL

Change Record:

VERSION	DATE	AUTHOR	COMMENTS
0.1	18- Nov - 2022	Prateek Lapalikar	Introduction and architecture defined
0.2	19 - Nov - 2022	Palkesh Gupta	Architecture & Architecture description appended and updated.

Reviews:

VERSION	DATE	REVIEWER	COMMENTS

Approval Status:

VERSION	REVIEW DATE	REVIEWED BY		APPROVED BY	COMMENTS

Contents

1.	Introduction	04
1.1	What is Low-Level Design Document?	04
1.2	Scope.....	04
2.	Architecture	05
2.1	Power Bi Server.....	05
2.2	On Premise	06
2.3	On Cloud	06
2.4	Power Bi Services	07
3.	Architecture Description	08
4.	Unit Test Case.....	11

1. Introduction

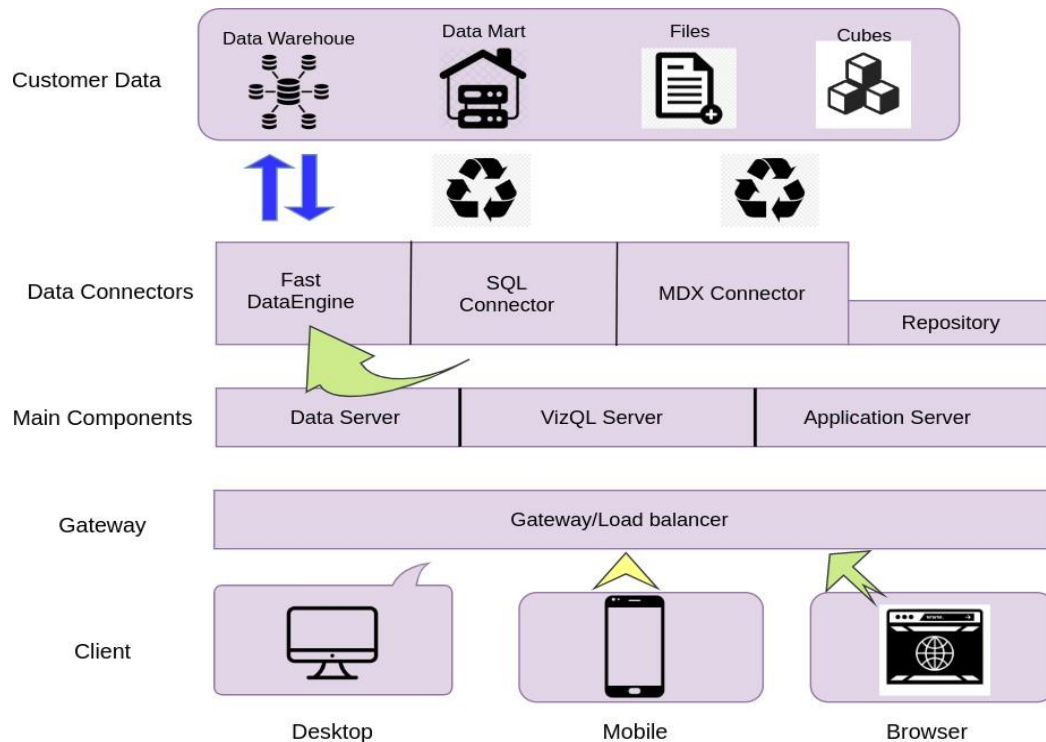
1.1 What is Low-Level design document?

The goal of the LDD or Low-level design document (LLDD) is to give the internal logic design of the actual program code for the House Price Prediction dashboard. LDD describes the class diagrams with the methods and relations between classes and programs specs. It describes the modules so that the programmer can directly code the program from the document.

1.2 Scope

Low-level design (LLD) is a component-level design process that follows a step-by-step refinement process. The process can be used for designing data structures, required software architecture, source code and ultimately, performance algorithms. Overall, the data organization may be defined during requirement analysis and then refined during data design work.

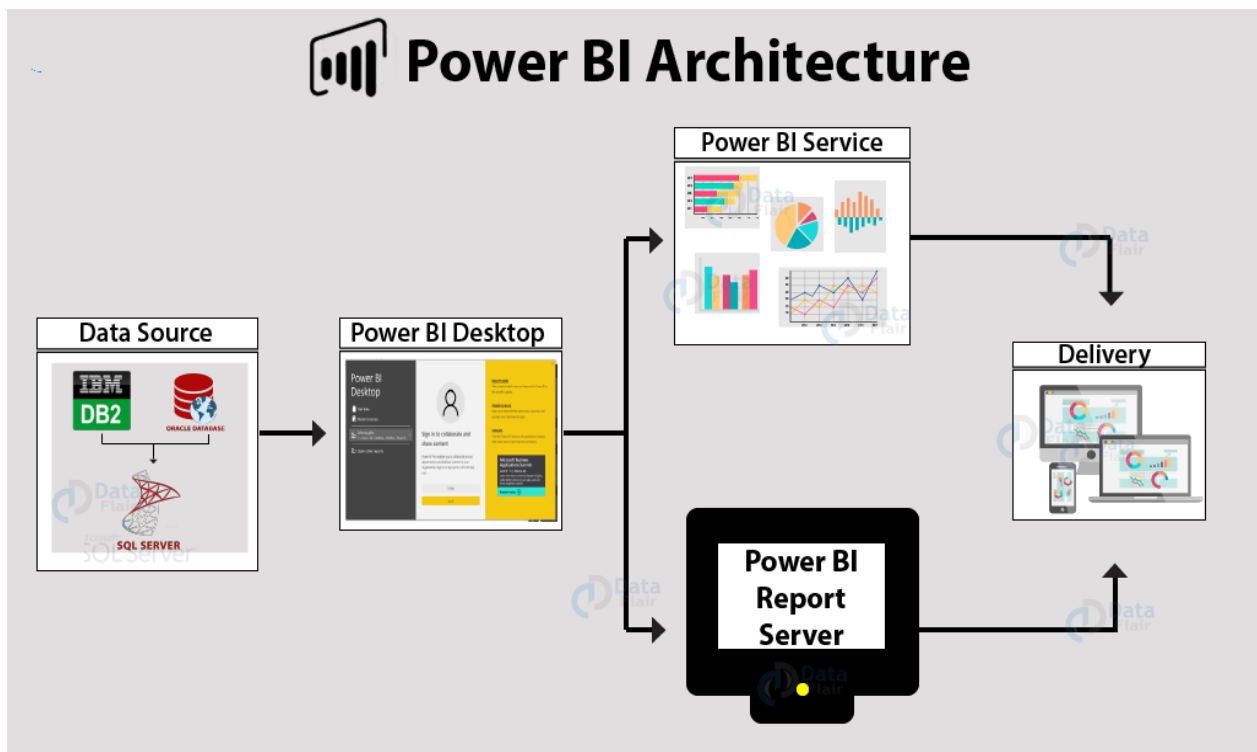
2. Architecture



Power Bi Server Architecture

Power BI is a business suite that includes several technologies that work together. To deliver outstanding business intelligence solutions, Microsoft Power BI technology consists of a group of components

The following diagram shows Power Bi Server's architecture:



On-Premise

Power BI Desktop is a companion development, authoring, and publishing tool. You can import data from data sources to Power BI Desktop and use it to create reports and then publish them on a Power BI Service or Power BI Report Server.

You can also publish Excel workbooks directly using Power BI Publisher for Excel to the Power BI Report Server. The SQL Server Data tools and Report Publisher help in *creating datasets, KPIs, mobile reports, paginated reports*, etc. The reports from all kinds of reports are published to the Power BI Report Server from where they are distributed to the end-users.

On-Cloud

An important component in Power BI architecture is the Power BI Gateway. The Power BI Gateway acts as a secure channel to transport data from on-premise data sources to on-cloud apps or sites.

On the cloud side of the architecture, resides a lot of components. Like a complete Power BI suite having *dataflows, datasets, dashboards, reports, Power BI Embedded, Power BI Premium*, etc. You can embed your reports and dashboards into *Teams, SharePoint, custom applications*, etc. There are on-cloud data sources as well that connects to Power BI tools via direct connections.

At last, there is a layer of authenticated users who share the published reports and dashboard and collaborate with one another to make educated decisions based on the insights. There are different kinds of users who consume Power BI reports and dashboards and connect through *web browsers, Excel, third-party tools, and mobile devices* (iOS, Windows, Android apps).

Power BI Service

As we have learned in the earlier sections, all the reports that you create in Power BI Desktop are published on a cloud platform known as Power BI Service.

Users can access the reports and dashboards from Power BI Service using client platforms like websites, mobile devices, etc. This means that every client who wants to access content created on Power BI needs to interact with Power BI Service. And so, we must take a look under the hood and learn how Power BI Service works.

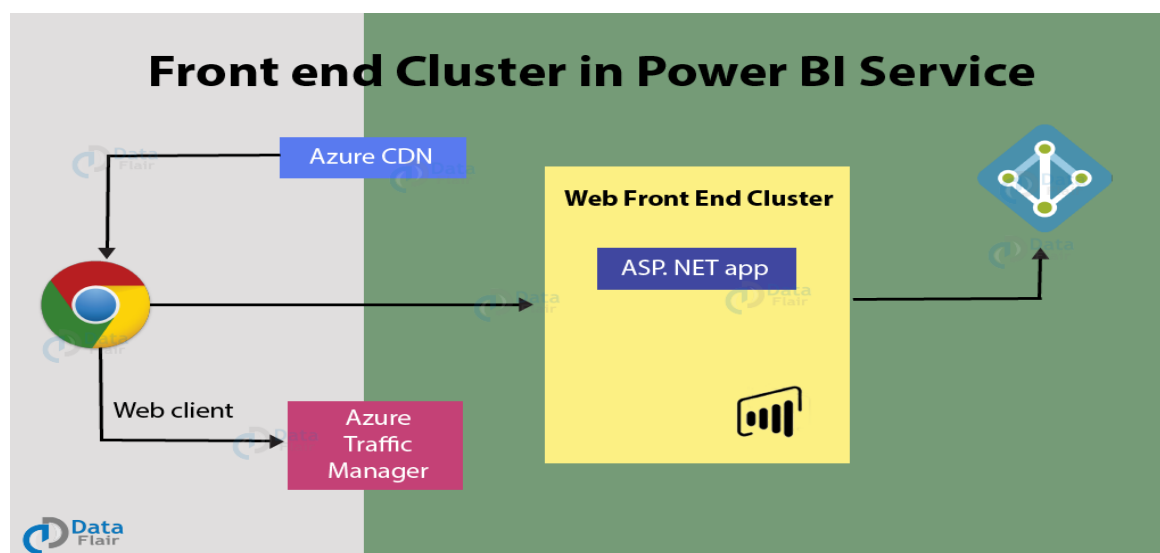
Power BI Service's architecture consists of two parts:

- A front end
- A back end

Front End cluster

The front end also called the web front-end cluster acts as an intermediary between clients and the back end. The front end services are used for establishing an initial connection and authenticating clients using Azure Active Directory. The Azure Active Directory stores user identities.

Along with this, Azure Traffic Manager is used to direct user requests to the nearest data center after authentication. Once a client/user is authenticated, the Azure Content Delivery Network (CDN) distributes static Power BI content/files to users.



Back End Cluster

The Power BI services at the back end take care of *visualizations, datasets, storage, reports, data connections, data refreshing, and other interactions* with Power BI. At the back-end, a web client has only two direct points of interaction, **Azure API Management**, and **Gateway Role**. These two components are responsible for *load balancing, authentication, authorization, routing, etc.*

Working of Power BI Service

- Power BI stores its data in two main repositories; **Azure block storage** and **Azure SQL database**. Azure block storage stores the datasets uploaded by users and all the metadata and system-related data is stored in the Azure SQL database.
- After Azure API Management authenticates a user request, it is sent to the Gateway Role. The Gateway Role processes the requests and directs them to suitable components like *Presentation Role, Background Job Processing Role, Data Role, and Data Movement Role*.
- For instance, the Presentation Role handles all the visualization related queries like for dashboards and reports.
- For all the data related queries, the request is sent by the Gateway Role to the Data Role or Data Movement Role.
- Power BI Service back end uses Azure Service Bus to connect on-premise [datasources](#) with the cloud. Azure Service Bus receives all the requests to fetch data from the on-premise data source. Then it processes the request and executes the query on the on-premise data source to retrieve data from it to the cloud service.
- The Azure Service Fabric manages all the microservices and components associated with running Power BI.
- Azure AD Cache helps in real-time reporting using the data stored in the in-memory of the Power BI system.

3. Architecture Description

3.1. Data Description

The Dataset contains various company names, market cap, EBITDA etc.

1. Symbol: - Represents the company in short form.
2. Name: - All the country company.
3. Sector: - Sector-wise distribution of companies.
4. Price
5. Price/Earning
6. Dividend Yield
7. Earning/Share
8. 52 Week Low
9. 52 Week High
10. Market Cap
11. EBITDA
12. Prices/Sales
13. Price/Book

3.2. Web Scrapping

Web scrapping is a technique to automatically extract content and data from websites using bots. It is also known as web data extraction or web harvesting. Web scrapping is made simple now days, many tools are used for web scrapping. Some of python libraries used for web scrapping are BeautifulSoup, Scrapy, Selenium, etc.

3.3. Data Transformation

In the Transformation Process, we will convert our original datasets with other necessary attributes format. And will merge it with the Scrapped dataset.

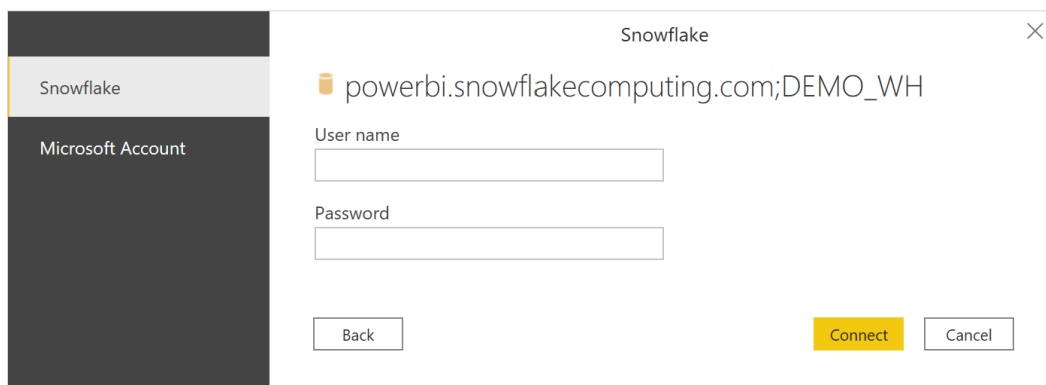
3.4. Data Insertion into Database

- Database Creation and connection - Create a database with name passed. If the database is already created, open the connection to the database.
- Table creation in the database.
- Insertion of files in the table

3.5 Make the Snowflake connection and set up the data source

Step 1: Configuring Power Bi

Launch Power Bi on your workstation and click import data select the snowflake option Then enter the url and name of dataset.



The screenshot shows a 'Snowflake' connection window. On the left, there's a sidebar with 'Snowflake' and 'Microsoft Account' options. The main area displays the Snowflake logo, the URL 'powerbi.snowflakecomputing.com;DEMO_WH', and input fields for 'User name' and 'Password'. At the bottom, there are three buttons: 'Back', 'Connect' (highlighted in yellow), and 'Cancel'.

4. Unit Test Cases

TEST CASE DESCRIPTION	EXPECTED RESULTS
Page Navigator	When (control + click) it goes to the selected page
Slicers For Sector companies	When clicked on the slicer, a dropdown should occur which describes the sector name.
Slicers For Name of companies	When clicked on the slicer, a dropdown should occur which describes the company's name.
Back Buttons	For going to the previous page press (control + click)
Reset Button	For Reset all the selected slicers press (Control + click) on the Reset
Next Button	For going to the next page press (control + click)

