Architecture Design

Analysing Google Apps Store

Written By	**Swaraj Ranjan Behera**
Document Version	0.1
Last Revised Date	25-April-2025

DOCUMENT CONTROL

Change Record:

VERSION	DATE	AUTHOR	COMMENTS
0.1	25- April-2025	**Swaraj Ranjan Behera**	Introduction and architecture defined

Reviews:

VERSION	DATE	REVIEWER	COMMENTS
0.1	25-April - 2025	**Swaraj Ranjan Behera**	Unit test cases to be added

Approval Status:

VERSION	REVIEW DATE	REVIEWED BY		APPROVED BY	COMMENTS

Contents

1. Introduction

1.1 What is an Architecture Design Document?

Any software project requires an architectural design to represent the overall system structure. According to the IEEE, architectural design is defined as:

"The process of defining a collection of hardware and software components and their interfaces to establish the framework for the development of a computer system."

In the context of this project, the software being developed is a data analysis and visualization system for apps available on the Google Play Store.

Every architectural style describes a system category that consists of the following:

- **A set of components** – e.g., databases, data pipelines, or data visualization modules – that perform the core functions of the system.
- **A set of connectors** – which support coordination, communication, and cooperation between components (e.g., between Python scripts and Power BI).
- **Integration rules** – describing how individual components can be integrated to form a cohesive system.
- **Semantic models** – used by designers to understand and represent the overall behavior and properties of the system.

1.2 Scope

The Architecture Design Document (ADD) is a formal guide that outlines the structural design process for this project. It follows a step-by-step refinement model used for:

- Designing the data structures and flow
- Establishing the required software architecture (ETL + Power BI)
- Organizing source code components and interaction layers
- Enabling efficient performance through algorithmic planning

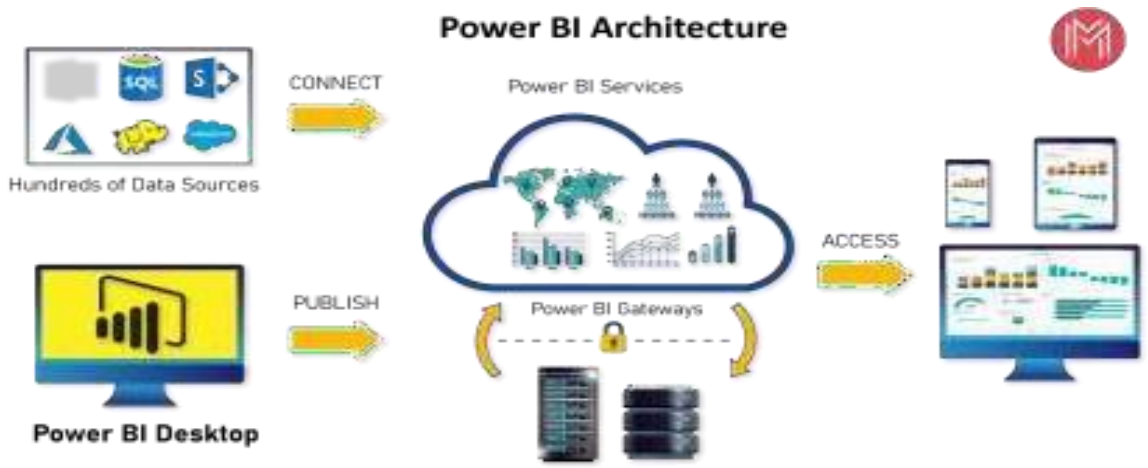
This document lays the foundation for system development from requirement analysis to data flow design and visual dashboard creation using Power BI.

2. Architecture

Microsoft Power BI is a comprehensive business intelligence suite designed to transform raw data into meaningful insights through interactive dashboards and visualizations. It consists of various interconnected technologies that work together to deliver a full end-to-end data solution. In the context of the **Google Play Store Data Analysis project**, Power BI helps clean, model, analyze, and visualize data collected from app listings.

Here's a breakdown of the major Power BI components used (or referenced) in the architecture:

- **Power Query**:
  Used within Power BI Desktop to perform data mash-up and transformation tasks such as filtering, cleaning, and reshaping the raw CSV data from the Google Play Store.
- **Power BI Desktop**:
  The primary development tool used in this project. It allows data import, transformation using Power Query, creation of data models, and building of interactive reports and dashboards.
- **Power BI Mobile**:
  Enables users to view published dashboards on mobile devices (Android, iOS, and Windows). Useful for sharing insights on the go.
- **Power Pivot**:
  Supports data modeling and relationships within the Power BI Desktop environment, enabling fast calculations with DAX (Data Analysis Expressions).
- **Power View**:
  Offers an interactive data visualization experience within Power BI Desktop for detailed insights.
- **Power Map** (now part of 3D Maps in Excel):
  Not actively used in this project but useful for creating 3D geographical visualizations if location-based insights were needed in future versions.
- **Power Q&A**:
  Allows users to interact with the report using natural language queries (e.g., "Top-rated apps in the Tools category"). It's part of Power BI Service and can be useful for executive-level reports.



2.1 Data Sources

A key strength of Power BI lies in its ability to connect to a wide variety of data sources—ranging from local files and databases to cloud-based services and live connections. When importing data from on premise or online services, the dataset size is limited to 1 GB in Power BI Free and Pro versions.

Some commonly used data sources in Power BI include:

- Excel
- Text/CSV
- XML
- JSON
- Oracle Database
- IBM DB2 Database
- MySQL Database
- PostgreSQL Database
- Sybase Database
- Teradata Database
- SAP HANA Database
- SAP Business Warehouse Server
- Amazon Redshift
- Impala
- Google Big Query (Beta)
- Azure SQL Database
- Salesforce Reports
- Google Analytics
- Facebook
- GitHub

For the **Google Play Store Analysis Project**, data is primarily sourced from CSV files containing app metadata such as names, ratings, reviews, installs, categories, and more.

2.2 Power BI Desktop

Power BI Desktop is a client-side application and the main development environment used to build analytical reports and dashboards. It offers a robust set of tools that allow users to:

- Connect to various data sources
- Clean and transform data using Power Query
- Perform data modeling with relationships and hierarchies
- Define business metrics and calculated columns using DAX
- Design visual reports with interactive elements

It is freely available and can be installed on Windows systems. In this project, Power BI Desktop serves as the primary platform for preparing the data pipeline, developing visuals, and publishing the final report to the Power BI Service

2.3 Power BI Service

Power BI Service is a web-based platform that allows users to publish, share, and collaborate on reports and dashboards created in Power BI Desktop. It is accessible through browsers and acts as a central hub for report management and real-time data monitoring.

Power BI Service is available in three versions:

- **Free Version** – Offers basic functionality for individual use.
- **Pro Version** – Enables sharing, collaboration, and data refreshes.
- **Premium Version** – Provides enhanced capacity, dedicated cloud resources, and on premise report deployment via Power BI Report Server.

Power BI Service is also referred to as:

- **PowerBI.com**
- **Power BI Workspace**
- **Power BI Site**
- **Power BI Web Portal**

It includes additional features like natural language Q&A, data alerts, and scheduled refreshes for reports and dashboards.

2.4 Power BI Report Server

Power BI Report Server is an on-premise solution similar to Power BI Service but designed for organizations that prefer to host their reports locally instead of using cloud services. It allows report development, deployment, and sharing within a secure, internal environment.

This option is ideal for companies with strict data governance or compliance requirements. A **Power BI Premium license** is required to use Power BI Report Server.

2.5 Power BI Gateway

Power BI Gateway acts as a secure bridge between on-premise data sources and Power BI services (both cloud and on-premise). It allows real-time data refreshes and scheduled updates from local databases to Power BI reports and dashboards.

Organizations typically use gateways to ensure data security while enabling seamless analysis from cloud-based Power BI platforms.

There are two types:

- **Personal Gateway** – For individual use
- **Enterprise Gateway** – For organizational use with centralized management

2.6 Power BI Mobile

Power BI Mobile is a native application available for **Android**, **iOS**, and **Windows** mobile devices. It allows users to access, view, and interact with their reports and dashboards on the go.

The mobile app supports features such as:

- Push notifications
- Touch-based navigation
- Mobile-optimized visualizations

2.7 Power BI Embedded

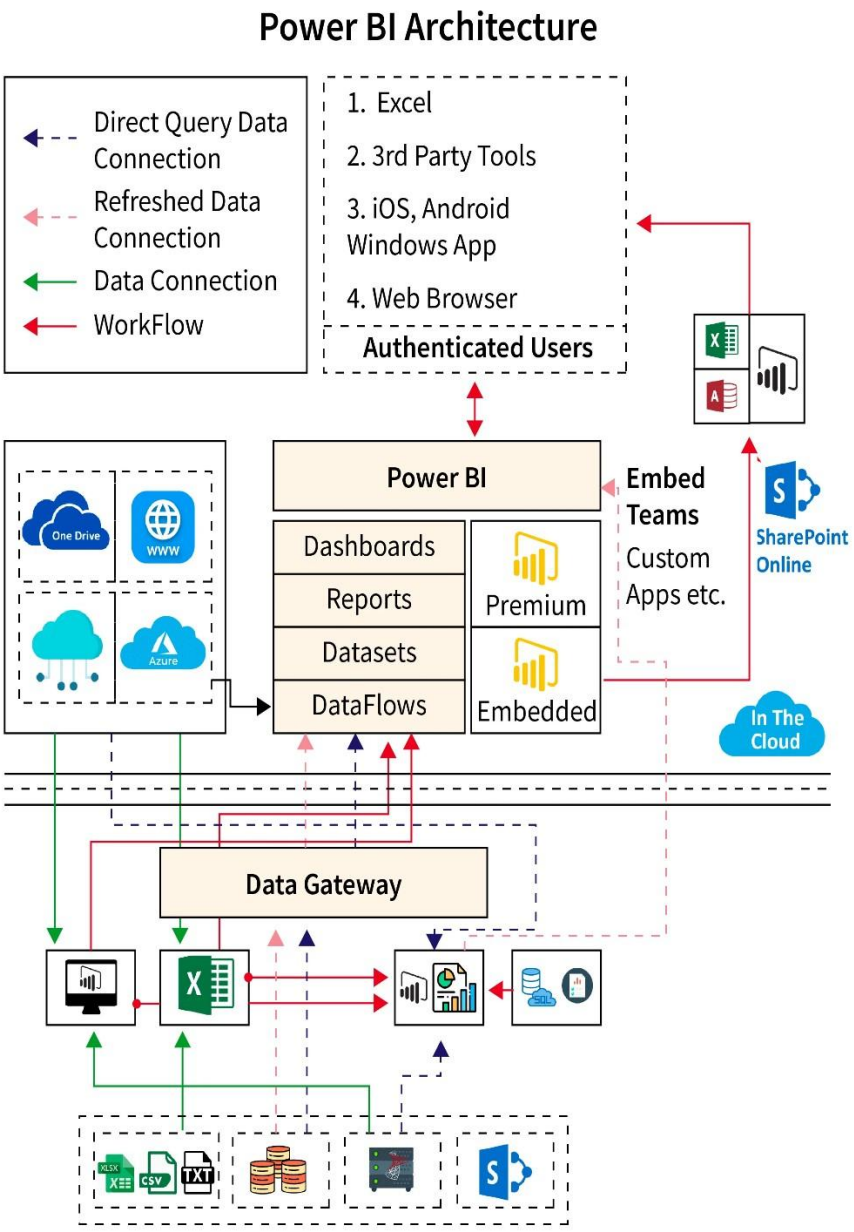
Power BI Embedded is a service designed for developers to integrate Power BI reports and dashboards into custom applications using APIs and SDKs. This allows organizations to deliver analytics to end-users without requiring them to access Power BI directly.

Common use cases include:

- Embedding analytics in web portals or customer-facing apps
- Creating interactive dashboards within internal tools
- Customizing user experiences without needing full Power BI licenses

---

3. Working of Power BI Architecture



3.1 On-Premise

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

Additionally, Excel workbooks can be published directly using **Power BI Publisher for Excel** to the Power BI Report Server. Tools like **SQL Server Data Tools (SSDT)** and **Report Builder** help create datasets, KPIs, mobile reports, and paginated reports. All these reports can be published to the Power BI Report Server and then distributed to end-users securely.

3.2 On-Cloud

An essential component in Power BI architecture is the **Power BI Gateway**, which serves as a secure channel for transferring data from on-premise sources to cloud-based Power BI services.

On the cloud side, the Power BI ecosystem includes several components such as:

- **Dataflows**
- **Datasets**
- **Dashboards**
- **Reports**
- **Power BI Embedded**
- **Power BI Premium**

These components enable advanced reporting, collaboration, and analytics. Reports and dashboards can be embedded into **Microsoft Teams**, **SharePoint**, or **custom applications**.

There are also cloud-based data sources that connect to Power BI tools via **direct connections**, enabling real-time analysis.

Finally, an authentication layer manages user access. Authenticated users can view and collaborate on reports and dashboards through **web browsers, Excel, third-party tools, and mobile devices** (iOS, Windows, Android). These users make data-driven decisions by sharing and analyzing insights collaboratively.

---

4) Power BI Service

Power BI Service is the **web-based platform** through which users access reports and dashboards using client platforms such as web browsers, mobile apps, and more. Any client who wants to view or interact with content created in Power BI must go through the Power BI Service.

To understand how Power BI Service works, let's explore its architecture, which is divided into two main components:

- **Front-End Cluster**
- **Back-End Cluster**

4.1 Front-End Cluster

The **Front-End Cluster**, also called the **Web Front-End Cluster**, acts as a **bridge between the clients and the backend** services.

- It handles **initial connections and user authentication** using **Azure Active Directory (Azure AD)**, which stores user identities.
- **Azure Traffic Manager** routes user requests to the **nearest data center** after successful authentication.
- Once authenticated, **Azure Content Delivery Network (CDN)** delivers **static Power BI content** (such as visuals and layout files) quickly to the users.

4.2 Back-End Cluster

The **Back-End Cluster** manages all the **core operations and processing** behind Power BI Service.

Here's how it works:

- Power BI uses two main storage components:
  - **Azure Blob Storage** – for storing datasets uploaded by users.
  - **Azure SQL Database** – for storing metadata and system-related data.
- After the **Azure API Management** authenticates a request, it sends it to the **Gateway Role**, which is responsible for routing the request to the appropriate backend service.

  Some of the key backend roles include:

  - **Presentation Role** – Handles all visualization-related requests (like dashboards and reports).
  - **Data Role** and **Data Movement Role** – Handle data processing, fetching, and transformations.
- For **on-premise data sources**, Power BI uses **Azure Service Bus**, which:
  - Receives data requests,
  - Connects to the on-premise source,
  - Executes queries,
  - Transfers the results securely to the Power BI cloud service.
- The **Azure Service Fabric** manages all microservices required to run Power BI Service smoothly.
- **Azure AD Cache** helps enable **real-time reporting** by storing data in the in-memory cache for fast access.