High Level Design (HLD)

Analyze International Debt Statistics

By - Aman Kumar

Document Version Control

Date Issued	Version	Description	Author
10th April 202	1.0	First Version of Complete HLD	Aman Kumar

Contents

Abstract

International debt is the external debt a nation, its government, or its citizens are obligated to pay. When foreign entities have more claims on a country's economy than domestic entities have on the economies of other nations, that country experiences an external deficit. Loans to both government and private sector groups within a nation make up that nation's foreign debt. Other governments, international institutions (mainly the International Monetary Fund and The World Bank), as well as private sector lenders in other nations, extend loans that are a part of a country's foreign debt.

This project determines the international debt of various countries and analyse the data concluding which country has the maximum amount of debt and compare the various parameters from distinct countries to total international debt.

1 Introduction

1.1 Why this High-Level Design Document?

The purpose of this High-Level Design (HLD) Document is to add the necessary detail to the current project description to represent a suitable model for coding. This document is also intended to help detect contradictions prior to coding, and can be used as a reference manual for how the modules interact at a high level.

The HLD will:

- Present all of the design aspects and define them in detail
- Describe the user interface being implemented
- Describe the hardware and software interfaces
- Describe the performance requirements
- Include design features and the architecture of the project
- List and describe the non-functional attributes like:
  - Security
  - Reliability
  - Maintainability
  - Portability
  - Reusability
  - Application
  - Compatibility
  - Resource
  - Utilization
  - Serviceability

1.2 Scope

The HLD documentation presents the structure of the system, such as the database architecture, application architecture (layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly-technical terms which should be understandable to the administrators of the system.

2 General Description

2.1 Product Perspective & Problem Statement

It talks about the phenomena known as the global debt crisis. First and foremost, it is crucial to take into account the problems, such as the causes of the development; hence, the social, economic, financial, political, and technological variables will be examined.

2.2 Tools used

Business Intelligence tools and libraries works such as Numpy, Pandas, Excel, R, Tableau, Power BI are used to build the whole framework.

3 Design Details

3.1 Functional Architecture

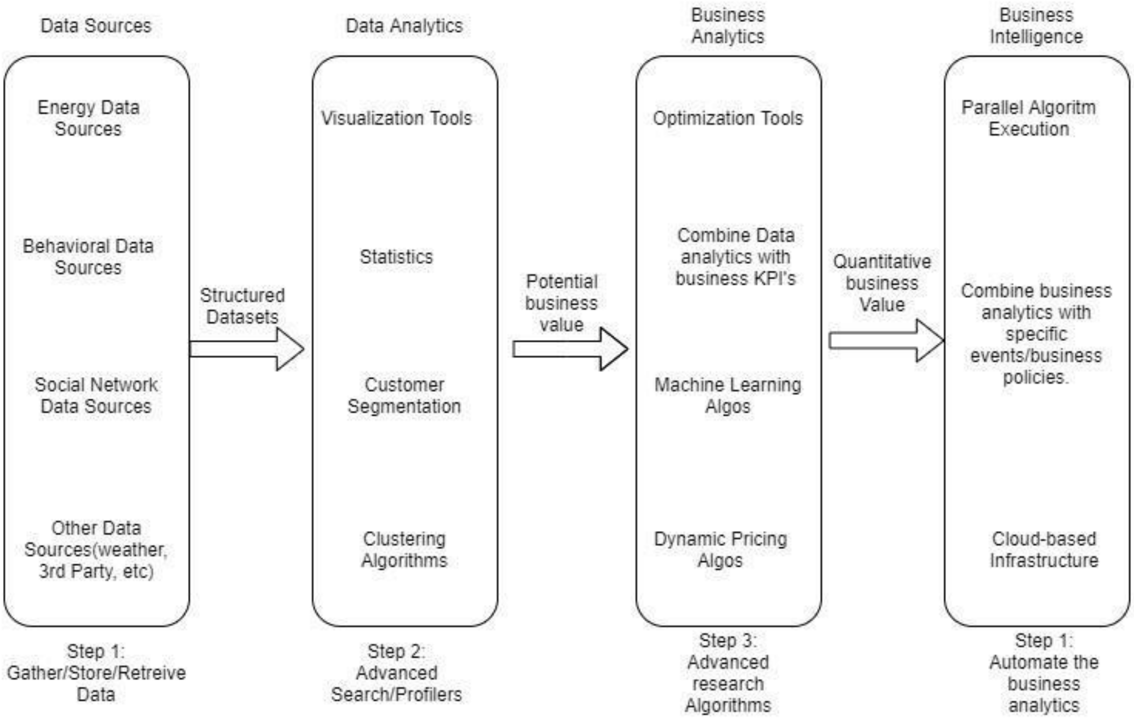
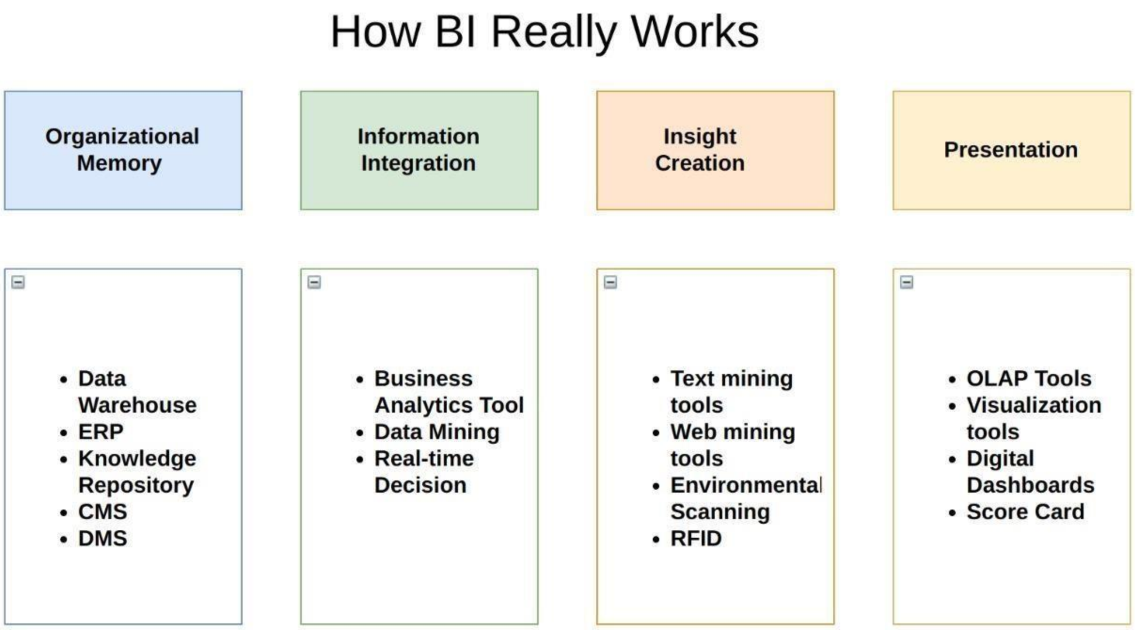

Data Sources
Energy Data Sources
Behavioral Data Sources
Social Network Data Sources
Other Data Sources(weather, 3rd Party, etc)
Step 1: Gather/Store/Retreive Data
Structured Datasets
Data Analytics
Visualization Tools
Statistics
Customer Segmentation
Clustering Algorithms
Step 2: Advanced Search/Profilers
Potential business value
Business Analytics
Optimization Tools
Combine Data analytics with business KPI's
Machine Learning Algos
Dynamic Pricing Algos
Step 3: Advanced research Algorithms
Quantitative business Value
Business Intelligence
Parallel Algoritm Execution
Combine business analytics with specific events/business policies.
Cloud-based Infrastructure
Step 1: Automate the business analytics

Figure 1: Functional Architecture of Business Intelligence



How BI Really Works

Organizational Memory	Information Integration	Insight Creation	Presentation
• Data Warehouse • ERP • Knowledge Repository • CMS • DMS	• Business Analytics Tool • Data Mining • Real-time Decision	• Text mining tools • Web mining tools • Environmental Scanning • RFID	• OLAP Tools • Visualization tools • Digital Dashboards • Score Card

3.2 Optimization

Your data strategy drives performance

- Minimize the number of fields
- Minimize the number of records
- Optimize extracts to speed up future queries by materializing calculations, removing columns and the use of accelerated views

Reduce the marks (data points) in your view

- Practice guided analytics. There's no need to fit everything you plan to show in a single view. Compile related views and connect them with action filters to travel from overview to highly-granular views at the speed of thought.
- Remove unneeded dimensions from the detail shelf.
- Explore. Try displaying your data in different types of views. **Limit your filters by**

  number and type

- Reduce the number of filters in use. Excessive filters on a view will create a more complex query, which takes longer to return results. Double-check your filters and remove any that aren't necessary.
- Use an include filter. Exclude filters load the entire domain of a dimension, while include filters do not. An include filter runs much faster than an exclude filter, especially for dimensions with many members.
- Use a continuous date filter. Continuous date filters (relative and range-of-date filters) can take advantage of the indexing properties in your database and are faster than discrete date filters.
- Use Boolean or numeric filters. Computers process integers and Booleans (t/f) much faster than strings.
- Use parameters and action filters. These reduce the query load (and work across data sources).

Optimize and materialize your calculations

- Perform calculations in the database
- Reduce the number of nested calculations.
- Reduce the granularity of LOD or table calculations in the view. The more granular the calculation, the longer it takes.
  - LODs - Look at the number of unique dimension members in the calculation.
  - Table Calculations - the more marks in the view, the longer it will take to calculate.

- Where possible, use MIN or MAX instead of AVG. AVG requires more processing than MIN or MAX. Often rows will be duplicated and display the same result with MIN, MAX, or AVG.
- Make groups with calculations. Like include filters, calculated groups load only named members of the domain, whereas Tableau's group function loads the entire domain.
- Use Booleans or numeric calculations instead of string calculations. Computers can process integers and Booleans (t/f) much faster than strings. Boolean>Int>Float>Date>DateTime>String

4 KPIs

Dashboards will be implemented to display and indicate certain KPIs and relevant indicators for the disease.

As and when, the system starts to capture the historical/periodic data for a user, the dashboards will be included to display charts over time with progress on various indicators or factors

4.1 KPIs (Key Performance Indicators)

Key indicators displaying a summary of the International Debt and its relationship with different metrics

1. Finding the number of distinct countries.
2. Finding out the distinct debt indicators.
3. Totalling the amount of debt owed by the countries.
4. Country with the highest debt.
5. Country with the lowest debt.
6. Average amount of debt across indicators.
7. The highest number of principal repayments.
8. The most common debt indicator.