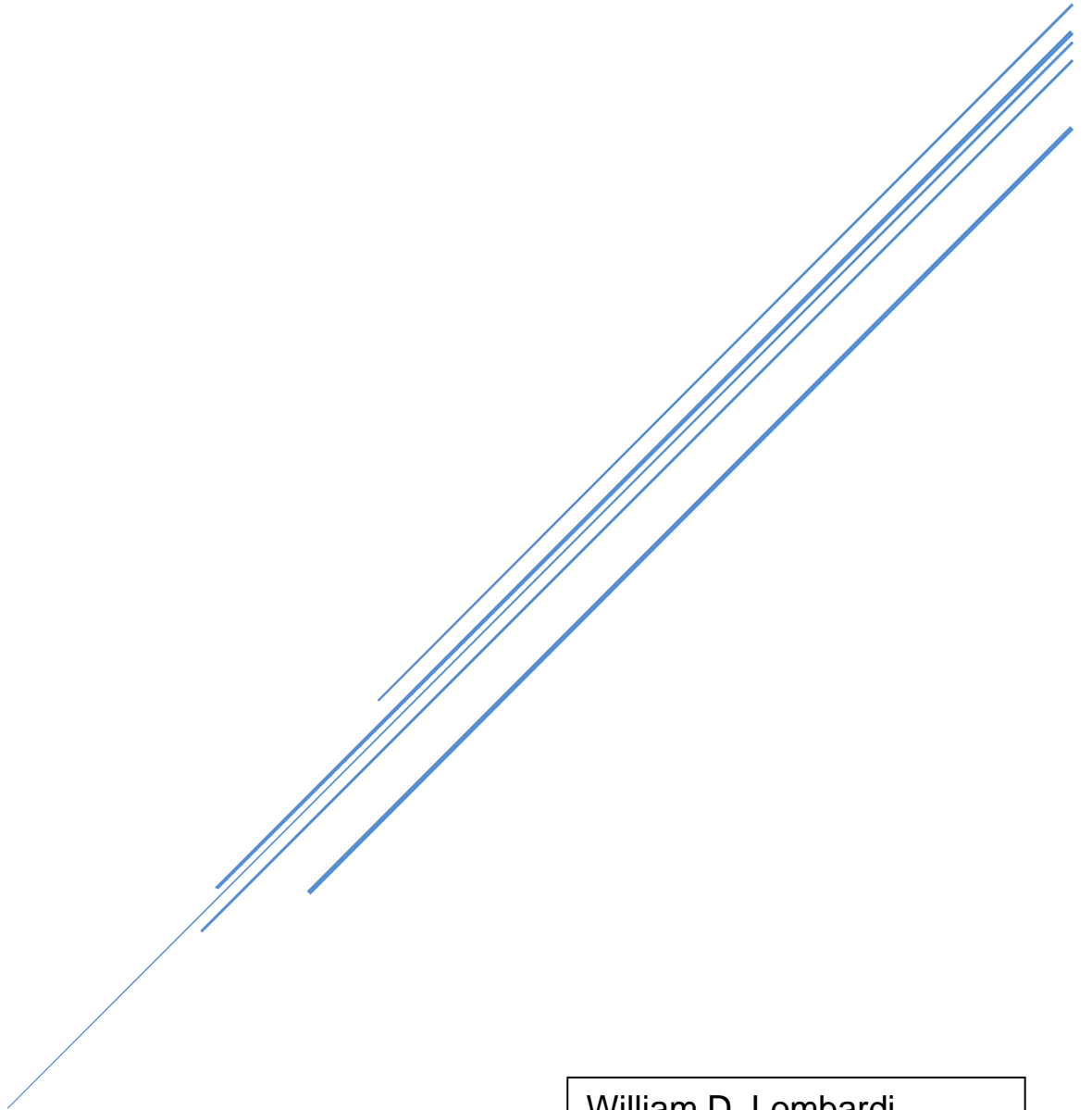


FINAL PROJECT (GROUP 1)

Data Warehouse



William D. Lombardi Rohit Anchan Akshit Salian Fernando Granato
--

Company Description

Name: BBank

Location: 210 Whatever street, Syracuse NY

Main Products: Checking accounts, savings accounts and student loans.

Market: Undergraduate and Graduate students from Syracuse area.

Data Warehouse Mission Statement

BBank is a small, local bank focused on college students in Syracuse area that believes that smaller banks offer personalized services with little online support and bigger banks offer more standardized services with higher online support. Some of the main services they currently provide are: checking accounts, savings accounts, student loans. With all that in mind, they are currently looking to build a new Data Warehouse to support the current systems without creating a new system or new interface. They also want to expand their online services and to improve reliability, performance and security. They also want to comply with GDPR and other industry regulations.

- *Business Identification:* small, local bank focused on students (grad and undergrad) in Syracuse area. Main services provided are: checking accounts, savings accounts, student loans.
- *Business reasons:* Our sponsors want to expand their online services and to improve reliability, performance and security. They also want to comply with GDPR and other industry regulations.
- *Market:* The market currently comprises of big global banks and small local banks. Smaller banks offer personalized services with little online support and bigger banks offer more standardized services with higher online support.

Business Case

BBank is currently looking to offer more online services to their current customers, which are undergraduate and graduate students. They are looking to build a new Data Warehouse that supports the current system but, not develop a new interface or change the existing systems. The new system should comply with GDPR and should be accessed all times of the day.

Expectations:

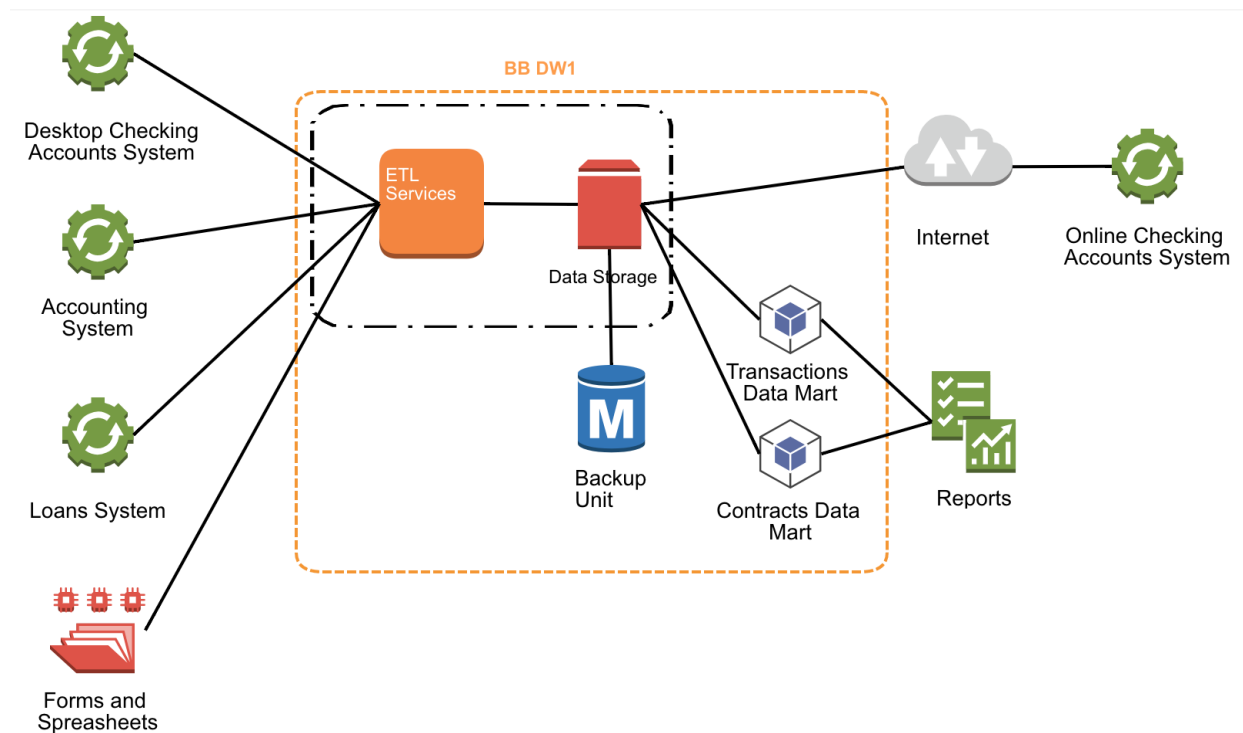
- To offer more online services through the new Data Warehouse in order to support the current system by creating a new system and to not develop a new interface or make and bring about changes in the legacy system. The security of the new system complies with the GDPR and could be accessible at all times of the day.
- We intend to deal with expectations by documenting the solutions we are going to propose and by being clear and honest about its` capabilities and limitations. No development will be done until the project is clear and approved.

Project Overview

Proposed Solution / Justification:

After analysing the current situation at the bank our team proposed to create a Data Warehouse to help the bank with its needs. The solution we intend to implement is a centralized data infrastructure where different systems and external sources could converge to make online services not only possible, but also safe, reliable and fast. Another important advantage of our proposed solution is the creation of a powerful tool the bank may use to perform analysis over multiple sources of data.

In our initial analysis we found out the bank uses multiple sources of data, including 2 softwares and broad variety of spreadsheets and forms. After the implementation of the proposed solution, all these components may be connected and used as source for analysis.
































Project Scope In/ Project Scope Out

Here are some of things that we as group has come up with we believe that are the following:

In Scope	Out Scope
<ul style="list-style-type: none"> • Create a secure and reliable DW to connect/bridge new systems with legacy systems. • Create documentation for inputs/outputs. • Train current maintenance team. • Infrastructure update recommendation. 	<ul style="list-style-type: none"> • Develop interface systems. • Improve or make changes to legacy system.

Work Breakdown Structure

		Task Mode ▾	Task Name ▾	Duration ▾	Start ▾	Finish ▾	Predecessors ▾
1			1 Project Planning	6 days	Mon 11/5/18	Mon 11/12/18	
2			1.1 Kickoff Meeting with Stakeholders	1 day?	Mon 11/5/18	Mon 11/5/18	
3			1.2 Develop Project Charter	4 days	Tue 11/6/18	Fri 11/9/18	2
4			1.3 Start Project	1 day	Mon 11/12/18	Mon 11/12/18	3
5			2 Pre-implementation	79 days	Tue 11/13/18	Fri 3/1/19	
6			2.1 Collect Requirements	19 days	Tue 11/13/18	Fri 12/7/18	
7			2.1.1 Conduct Interviews	15 days	Tue 11/13/18	Mon 12/3/18	4
8			2.1.2 Review current documentation	10 days	Tue 11/13/18	Mon 11/26/18	4
9			2.1.3 Review business processes and current systems	10 days	Tue 11/13/18	Mon 11/26/18	4
10			2.1.4 Produce Requirements Report	4 days	Tue 12/4/18	Fri 12/7/18	7,8,9
11			2.2 Design DW	15 days	Mon 12/10/18	Fri 12/28/18	
12			2.2.1 Brainstorm Sessions	5 days	Mon 12/10/18	Fri 12/14/18	10
13			2.2.2 Blueprint DW	15 days	Mon 12/10/18	Fri 12/28/18	12SS
14			2.2.3 Blueprint Infrastructure	15 days?	Mon 12/10/18	Fri 12/28/18	12SS
15			2.3 Feasibility Analysis	8 days	Mon 12/31/18	Wed 1/9/19	
16			2.3.1 Technical, Financial, Organizational reports	8 days	Mon 12/31/18	Wed 1/9/19	13,14
17			2.4 Approach Stakeholders to approve project design	2 days	Thu 1/24/19	Fri 1/25/19	
18			2.5 Hire Team	22 days	Mon 1/14/19	Tue 2/12/19	
19			2.5.1 Prepare Profiles and Start looking for candidates	5 days	Mon 1/14/19	Fri 1/18/19	
20			2.5.2 Conduct Interviews	5 days	Mon 1/21/19	Fri 1/25/19	19
21			2.5.3 Hire	2 days	Mon 1/28/19	Tue 1/29/19	17,20
22			2.5.4 Train Team	10 days	Wed 1/30/19	Tue 2/12/19	21
23			2.6 Procurement	35 days	Mon 1/14/19	Fri 3/1/19	
24			2.6.1 Identify Software licences and equipment needed	5 days	Mon 1/14/19	Fri 1/18/19	
25			2.6.2 Buy/Hire Equipment/Services	24 days	Mon 1/28/19	Thu 2/28/19	17
26			2.6.3 Receive Licences/Equipment	35 days	Mon 1/14/19	Fri 3/1/19	24SS
27			2.7 Create Project Schedule	2 days	Mon 1/28/19	Tue 1/29/19	17
28			2.8 Consolidate Project Plan	5 days	Mon 1/28/19	Fri 2/1/19	17

29		▲ 3 Implementation	173 days	Mon 1/28/19	Wed 9/25/19	
30		3.1 Set Development Environment	10 days	Mon 1/28/19	Fri 2/8/19	17
31		3.2 First Sprint Meeting	1 day?	Mon 2/11/19	Mon 2/11/19	30
32		3.3 Phase 1 Development iterations	65 days	Tue 2/12/19	Mon 5/13/19	31
33		3.4 Project Review	5 days	Tue 5/14/19	Mon 5/20/19	32
34		3.5 Phase 2 Development iterations	65 days	Tue 5/21/19	Mon 8/19/19	33
35		3.6 Product Benchmark	5 days	Tue 8/20/19	Mon 8/26/19	34
36		3.7 Production Infrastructure Setup	15 days	Tue 8/27/19	Mon 9/16/19	35
37		3.8 Final Development Iterations	15 days	Tue 8/27/19	Mon 9/16/19	35
38		3.9 Final Test, Benchmark and Tuning	5 days	Tue 9/17/19	Mon 9/23/19	37
39		3.10 Loading Legacy Data	20 days	Tue 8/27/19	Mon 9/23/19	35
40		3.11 Deploy on Production	2 days	Tue 9/24/19	Wed 9/25/19	39,38
41		▲ 4 Maintenance and Support	15 days	Thu 9/26/19	Wed 10/16/19	
42		4.1 Consolidate Final Documentation	8 days	Thu 9/26/19	Mon 10/7/19	40
43		4.2 Train Users	15 days	Thu 9/26/19	Wed 10/16/19	40
44		▲ 5 End Project	10 days	Thu 10/17/19	Wed 10/30/19	
45		5.1 Final Report	10 days	Thu 10/17/19	Wed 10/30/19	42,43

Estimated Cost and ROI

The total estimated cost to build the data warehouse would be around USD 250,000. It is expected that a company's profit margin is expected to increase by 25% when a data warehouse is build. So we are expecting that the Bank's profit margin will increase considerably once the Data Warehouse is build completely.

Project Team Members

<u>Part of what team?</u>	<u>Title</u>	<u>Skills/Responsibilities</u>
Implementation Team	Data Warehouse Designer	Has knowledge in the following: <ul style="list-style-type: none"> - SQL Server - DB Architecture - Python
Implementation Team	Data Warehouse and Platform developer	Has knowledge in the following: <ul style="list-style-type: none"> - SQL Server - Python - Legacy systems platform
Implementation Team	Data Quality Analyst	Has knowledge in the following: <ul style="list-style-type: none"> - Data profiling - Data discovery - Data Analytics

Implementation Team	BI Analyst (ETL and Reports)	Has knowledge in the following: <ul style="list-style-type: none"> - Report design - DB Performance - SQL Report Server
Implementation Team	Product Owner - From the bank	Has knowledge in the following: <ul style="list-style-type: none"> - Business process knowledge - Agile method
Implementation Team	External QA	Has knowledge in the following: <ul style="list-style-type: none"> - Quality Assurance
Management Team	Project Manager (configuration manager)	In charge of the following: <ul style="list-style-type: none"> - Document Tracker - Organized - Good time management - Able to communicate on command
Management Team	Budget and Procurement Officer	In charge of the following: <ul style="list-style-type: none"> - Reviews and approves/denies funding - Makes sure the resources are used to the fullest potential - Negotiate with suppliers and sponsor
Management Team	Project Sponsor	In charge of the following: <ul style="list-style-type: none"> - Funds the project

Project Stakeholders

Two main stakeholders were identified on our project. The first stakeholder is the board of directors of BBank and the second are public agencies that regulate banking market.

Business Processes

Interview Questions

- A. Why do you need to this data warehouse?
- B. When do you want this data warehouse?
- C. What are some of the requirements you want to implement?

Business Process:

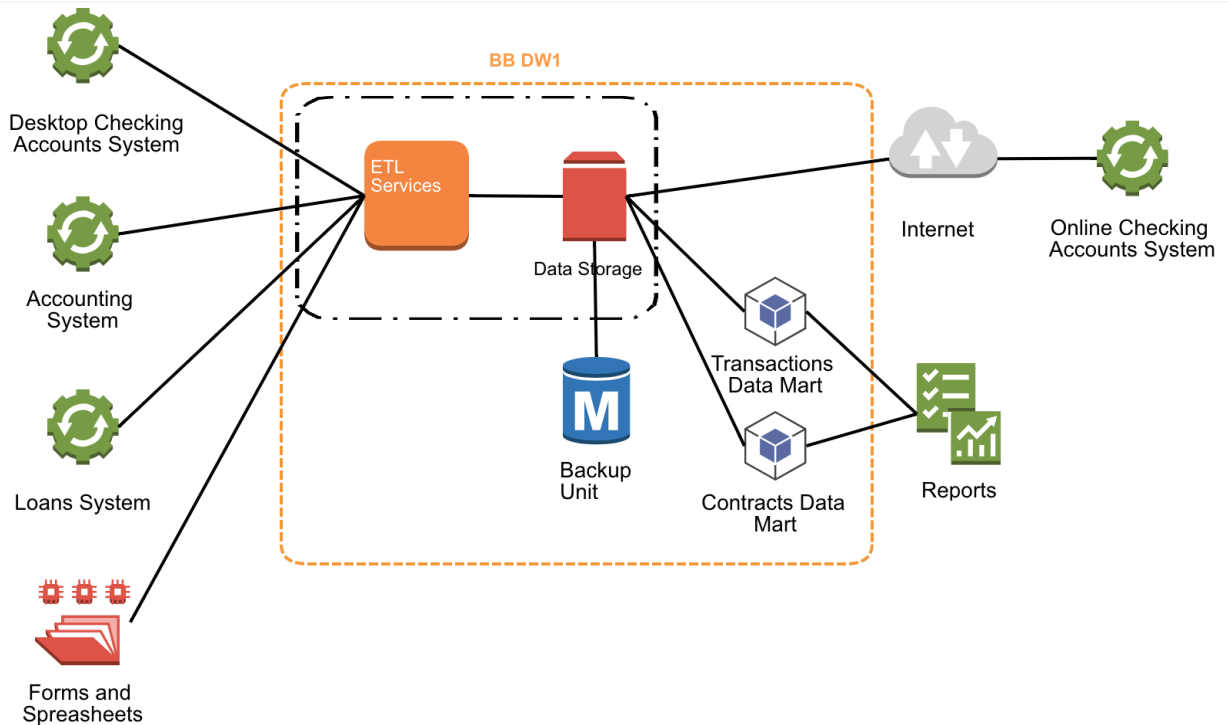
- 1) Open/Close Checking Account: This process involves opening and closing new/ old checking account
- 2) Bank Transactions: Bank transactions is the process wherein transactions which are done get recorded
- 3) Hire student loan
 - a) Sign loan contract
 - b) Monitor payments
 - c) End contract
- 4) Monitor transactions - outliers

Bus Matrix

Business Process Name	Fact Table	Fact Grain Type	Granularity	Facts	Account	Time	Transactions Amounts	Transaction Type	Transaction Details	Contract	Date	Due Amount	Paid Amount	Payment Date
When Transaction Occurs	Fact_Transactions	Accumulating Snapshot	One transaction every row	Account_ID, Date_ID, Time_ID, Transactions_Amounts, Transaction_Type, Transaction_Details	X	X	X	X	X		X			
When Loan Payment is paid	Fact_Payments	Accumulating Snapshot	One Loan Payment per row	Contract_ID, Date_ID, Time_ID, Due_Amount, Paid_Amount, Payment_Date		X				X	X	X	X	X

Design

The following diagram presents the design we are proposing for the data warehouse:



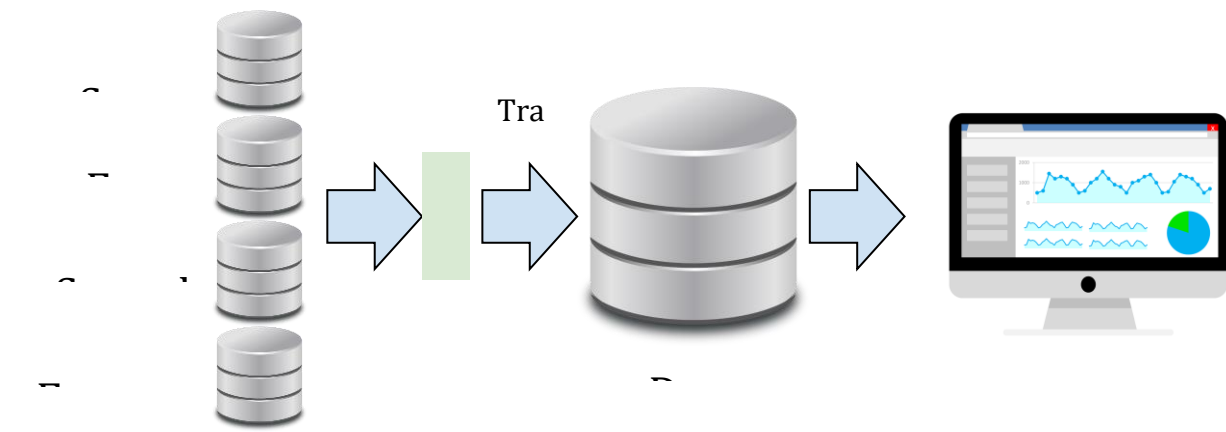
- Data Sources: 3 systems will be providing data for the data warehouse: Desktop Checking Account System, Accounting System and Loans System. Other external sources like spreadsheets and forms will also provide data to be loaded on the DW.
- ETL Systems: ETL systems will provide a proper interface between Data Sources and the Data Warehouse. Specific ETL systems will be developed for each group of data on each source. ETL systems will extract the data from sources, make this data adequate to the DW, and then populating it. Each ETL system should run as often as it is needed.
- Data Storage: This is the data base unit that will centralize all the data on the DW. The design of the dimensional database will be detailed on further topics of this report.
- Backup Unit: Backup systems.
- Data Marts: Custom database units built specifically to perform analysis regarding each of the business processes. On the current version of the DW there will be two data marts, one for transactions and one for loan contracts.
- Internet Interface: The online banking system will use an online interface to have access to DW information in order to provide online services to clients. This is a two way interface where the online system may produce new data and input it through ETL process.
- Reports: Those are internal or external reports built to provide information for the bank, customers or suppliers.

Issues Lists

Since our Data Warehouse is for a local bank, here is a list of problems that we thought of:

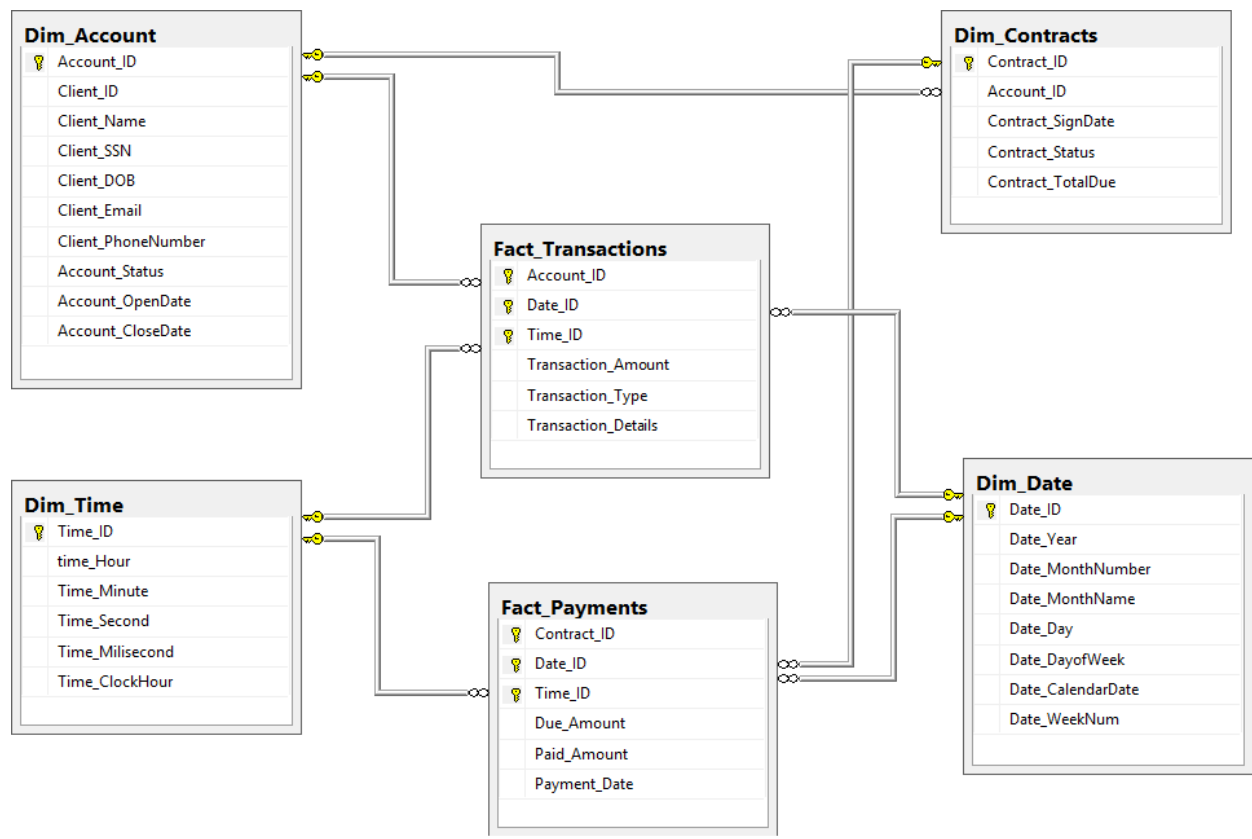
- Funding the project
- Purchasing the Infrastructure
- Not enough time to build the Data Warehouse
- Training the current employees on the Data Warehouse
- Making sure the legacy works with the new Data Warehouse we plan to implement
- Making sure the legacy data works with the new Data Warehouse we plan to implement
- Making sure the legacy and current data is secure
- Making sure the Data Warehouse complies with Business Regulation
- Maintaining the Data Warehouse we plan to implement
- Making sure that users of the Data Warehouse accept
- Making sure we hire the correct people that can either us build or/and support the new Data Warehouse
- Making sure the Data Warehouse is what the Banks wants
- Making sure the Data Warehouse is built with quality not just something that is put together.
- Making sure the Data Warehouse gets support from the stateholders
- Making sure we document everything that goes on with the Data Warehouse

ETL Process



There is a going to be 4 databases (System, Forms, Spreadsheets, External Data) that needs to be extracted and staged to be able to transformed and load into the data warehouse which can be transformed into analytics. Something worth mentioning is when data is being extracted, we have to make sure the data is not redundant and making it is good consistent to prevent issues later on.

Star Schema



Data Dictionary

1. Dim_Accounts

Column Name	Opt.	Format	Length	Description	Type of Key
Account_ID	yes	INT	Default	Unique ID for the dimension table	PK
Client_ID	no	INT	Default	Unique ID for a potential client	

Client_Name	no	VARCHAR	99	Potential client name	
Client_DOB	no	DATE	Default	Potential client birth date	
Client_SSN	no	INTEGER	9	Potential client social security number	
Client_PhoneNumber	yes	VARCHAR	10	Potential client phone	
Client_Email	yes	VARCHAR	50	Potential client email address	
Account_Status	no	BIT		Potential client account status	
Account_OpenDate	no	DATE		Potential client open date	
Account_CloseDate	no	DATE		Potential client close date	

2. Dim_Contracts

Column Name	Optional	Format	Length	Description	Type of Key
Contract_ID	no	INTEGER	10	Unique identification for a contract	PK
Account_ID	no	INTEGER	8	Account number of the client signing the contract	
Contract_SignDate	no	DATE		Date the contract was signed by both partys'.	
Contract_Status	no	BIT		Status of the contract.	
Contract_TotalDue	no	FLOAT		Total amount of fees charged	

3. Fact_Payments

Column Name	Optional	Format	Length	Description	Type of Key
Contract_ID	no	INTEGER	Default	Unique identification for a contract	PK,FK
Date_ID	no	INTEGER	Default	Account number of the client signing the contract	PK,FK
Time_ID	no	INTEGER	Default	Unique Time ID	PK,FK
Due_Amount	no	Float	Default	Total amount due	
Paid_Amount	no	Float	Default	Total amount paid	
Payment_Date	yes	Date	Default	The date when the last payment was recieved	

4. Fact Transactions

Column Name	Opt	Format	Length	Description	Type of Key
Account_ID	no	INT	Default	Unique ID of the Transaction	PK,FK
Date_ID	no	INT	Default	Unique Date ID of Transaction	PK,FK
Time_ID	no	INT	Default	Unique Timestamp ID of Transaction	PK,FK
Transaction_amount	no	FLOAT	Default	Amount of the Transaction	
Transaction_Type	no	INT	Default	Code of what type of transaction it is	
Transaction_Details	yes	VARCHAR	100	Details about the transaction	

5. Dim_Time

Column Name	Optional	Format	Length	Description	Type of Key
Time_ID	no	INT	Default	Unique Identification	PK

				of time	
time_Hour	no	INT	Default	Integer for the hour	
Time_Minute	no	INT	Default	Integer for the minute	
Time_Second	no	INT	Default	Integer for the second	
Time_Milisecond	no	INT	Default	Integer for the Millisecond	
Time_ClockHour	no	Time	Default	The time at that moment	

6. Dim_Date

Column Name	Optional	Format	Length	Description	Type of Key
Date_ID	no	INT	Default	Unique identification of the date	PK
Date_Year	no	INT	Default	The year of transaction	
Date_MonthNumber	no	INT	Default	The month number of the transaction	
Date_MonthName	no	VARCHAR	Default	The month of the transaction	
Date_Day	no	INT	Default	The day of the transaction	
Date_DayOfWeek	no	VARCHAR	Default	The day of week	
Date_CalenderDate	no	DATE	Default	The complete date	
Date_WeekNum	no	INT	Default	The week number	

Visualizations

1. **Balance Report**

- Users are clients
- Transaction Information calculated to daily (available) balance

2. **Accounting Report**

- Users are accounting employee
- Transaction for all the clients
- Biggest transactions and outliers

3. **Contracts**

- Each clients and billing employee
- Data: Payments done, Payments due, Active Contracts

Appendix - SQL - build

```
CREATE TABLE Dim_Account (  
Account_ID INT PRIMARY KEY,  
Client_ID INT NOT NULL,  
Client_Name VARCHAR(100) NOT NULL,  
Client_SSN INT NOT NULL,  
Client_DOB Date NOT NULL,  
Client_Email VARCHAR(50),  
Client_PhoneNumber VARCHAR(20),  
Account_Status BIT NOT NULL,  
Account_OpenDate DATE NOT NULL,  
Account_CloseDate DATE  
);
```

```
CREATE TABLE Dim_Date (  
Date_ID INT PRIMARY KEY,  
Date_Year INT NOT NULL,  
Date_MonthNumber INT NOT NULL,  
Date_MonthName VARCHAR(20) NOT NULL,  
Date_Day INT NOT NULL,  
Date_DayofWeek VARCHAR(20) NOT NULL,  
Date_CalendarDate DATE NOT NULL,  
Date_WeekNum INT NOT NULL  
);
```

```
CREATE TABLE Dim_Time (  
Time_ID INT PRIMARY KEY,  
time_Hour INT NOT NULL,  
Time_Minute INT NOT NULL,
```

```
Time_Second INT NOT NULL,  
Time_Milisecond INT NOT NULL,  
Time_ClockHour TIME NOT NULL  
);
```

```
CREATE TABLE Dim_Contracts (  
Contract_ID INT PRIMARY KEY,  
Account_ID INT NOT NULL,  
Contract_SignDate DATE NOT NULL,  
Contract_Status BIT NOT NULL,  
Contract_TotalDue FLOAT NOT NULL,
```

```
CONSTRAINT accountFK FOREIGN KEY (Account_ID) REFERENCES  
Dim_Account(Account_ID)  
);
```

```
CREATE TABLE Fact_Transactions (  
Account_ID INT NOT NULL,  
Date_ID INT NOT NULL,  
Time_ID INT NOT NULL,  
Transaction_Amount FLOAT NOT NULL,  
Transaction_Type INT NOT NULL,  
Transaction_Details VARCHAR(100),
```

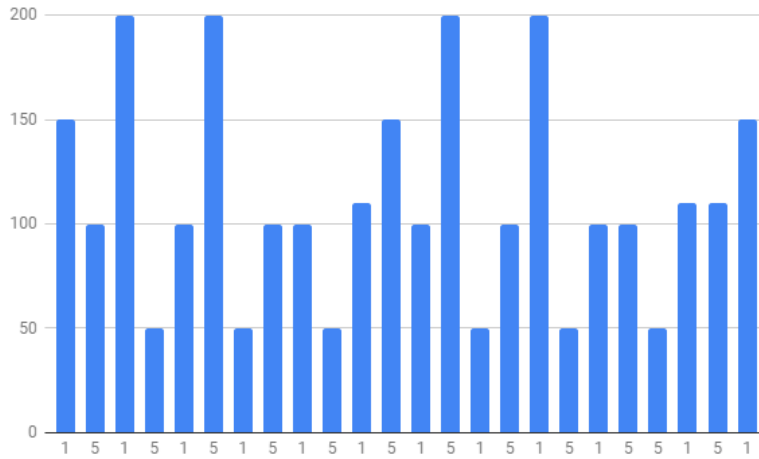
```
CONSTRAINT trasactionPK PRIMARY KEY (Account_ID, Date_ID, Time_ID),  
CONSTRAINT transaccFK FOREIGN KEY (Account_ID) REFERENCES  
Dim_Account(Account_ID),  
CONSTRAINT DateFK FOREIGN KEY (Date_ID) REFERENCES Dim_Date(Date_ID),  
CONSTRAINT timeFK FOREIGN KEY (Time_ID) REFERENCES Dim_Time(Time_ID),  
);
```

```
CREATE TABLE Fact_Payments (  
Contract_ID INT NOT NULL,  
Date_ID INT NOT NULL,  
Time_ID INT NOT NULL,  
Due_Amount FLOAT NOT NULL,  
Paid_Amount FLOAT NOT NULL,  
Payment_Date DATE,
```

```
CONSTRAINT paymentPK PRIMARY KEY (Contract_ID, Date_ID, Time_ID),  
CONSTRAINT contpayFK FOREIGN KEY (Contract_ID) REFERENCES  
Dim_Contracts(Contract_ID),  
CONSTRAINT datepayFK FOREIGN KEY (Date_ID) REFERENCES Dim_Date(Date_ID),  
CONSTRAINT timepayFK FOREIGN KEY (Time_ID) REFERENCES Dim_Time(Time_ID),
```

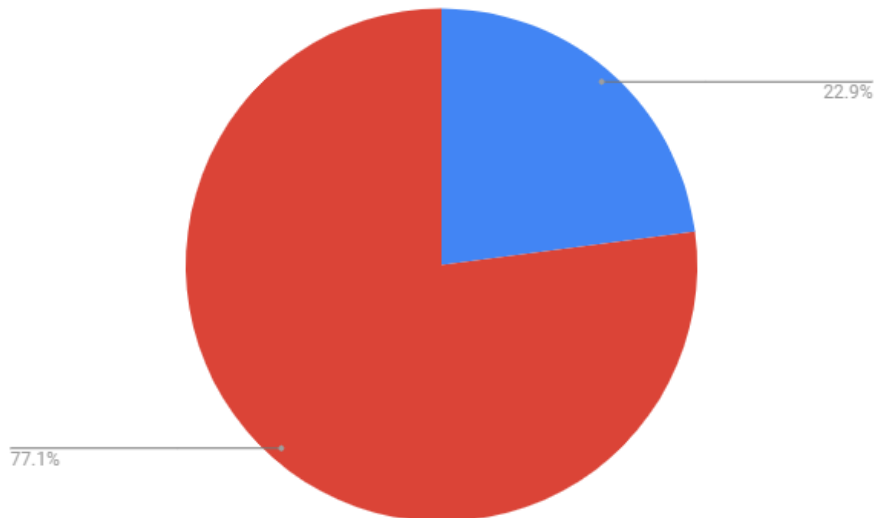
);

Trends



Loan Payment Across A Year

You can see that account #1 pays more across the year.



Transactions

About 22.9% of the transactions are credit and the other 77.08% of the transactions are debit.

Team Report

Implementation Plans

Step 1: Step up development environment for the Data Warehouse

Step 2: Hold our first sprint meeting about what should needs to be done with Data Warehouse this includes backlogs.

Step 3: Make any development iterations to the Data Warehouse
Step 4: Review current project process of the Data Warehouse
Step 5: Make any development iterations
Step 6: Benchmark the current Data Warehouse
Step 7: Finish the current infrastructure setup
Step 8: Make final development iterations
Step 9: Benchmark (Test and Tune) Data Warehouse
Step 10: Load in Legacy Data into Data Warehouse
Step 11: Deploy Data Warehouse

BI Plan

Step 1: Interview Stakeholders to determine current and desired process and create summary of business requirements
Step 2: List out crucial business questions that need to be answer
Step 3: Draft of some of the first dashboards, list out the facts, dimensions and filters
Step 4: Closing

Maintenance and Support

- Purge Data
 - The bank may want to retain many years worth of data. But, at some point, some data is going to have to come out in order to retain the system's performance. But, the current plan is to move the older data out of the system by moving it to some less-expensive means of archival storage.
- Tune the performance
 - Regardless, how expertly we designed the data warehouse, we are no expert that can foresee the real, day to day uses that it will evolve to support. We will need to periodically review how the data warehouse is being used and fine-tune its configuration to provide the best performance. In some cases, that may even mean revisiting the data warehouse design and restructuring data. There may be times where we will need to defragment and rebuild indexes. Weill We need to clear out obsolete logs and other historical data.
- Looking Deeper into Data Insights
 - We will need to review current business needs. Apart from that, once the bank collects more and more data, the stakeholders would require new reports which answers key business questions. To do this, we would dedicate a team that would look into the data, analyze it and then generate reports.