Group Retirement Saving Data Warehouse
Database Design Document

Version 1.7

03/24/2019

Concordia University

CEBD 1250

Prepared by

William Acosta (40110286)

Adel Tibourtine (40068607)

Table of Contents

Database Design Document 1
1. Overview 3
2. Assumptions/Constraints/Risks 4
2.1. Assumptions 4
2.2. Constraints 4
2.3. Risks 4
3. Design Decisions 5
3.1. Key Factors Influencing Design 5
3.2. Functional Design Decisions 5
3.3. Database Management System Decisions 6
3.4. Security and Privacy Design Decisions 6
3.5. Performance and Maintenance Design Decisions 7
4. Detailed Database Design 7
4.1. Roles and Responsibilities 10
4.2. Performance Monitoring and Database Efficiency 10
4.2.1. Operational Implications 10
4.2.2. Data Transfer Requirements 11
4.2.3. Data Formats 11
Appendix A: Acronyms 12
Table - Acronyms 12
Appendix B: DDLs 13

1. Overview

The company operates in financial services and has eight independent business units that provide different products and services (banking, insurance and investments). Almost every business unit has its own core systems and data architecture.

In this document, we will focus on the group retirement business unit. It offers B2B group RRSP (Registered retirement saving plan) services including employee's assets management and investment based on participant's long-term financial objectives. A Group retirement saving is designed to encourage employees to save at work by contributing through payroll deductions as well as voluntary contributions. Both employers and employees may contribute depending on the rules of the plan.

Like an individual RRSP, participants (employees) decide how their money is invested. Employers (plan sponsors) will provide a range of investment options to choose from. The whole plan design is administered by the financial provider.

The group retirement saving business unit operational teams use the core system to implement groups following the investment options, contribution and eligibility rules. All transactions are processed through the core system and stored in its database. At the initial stage of implementation, the employer load, on a SFTP server, a demographic file that contains all the information on employees (name, date of birth, gender, position, employment status, address and other HR related information).

Current challenges:

-In the absence of a standard methodology throughout the company, the consolidation of the information is laborious and requires significant effort.
-The only way for the team to analyze customer data and provide insights to improve business outcomes and customer experience is to run manual extractions from specific tables by subject. For example, if the need is to analyze contribution patterns by demographic dimensions, the team has to run two separate extractions, one for contribution transactions and one for the demographic information and merge them using Excel. On the other hand, given the limitation of the system, extractions could not be run for the whole line of business (all lists of customers) but needs to be specific to one group at the time.
-The "pressure" on core business systems is high given the large number of manual extractions of data in the absence of a BI practice and the appropriate tools.
-The company does not leverage the information asset. If BI solutions are more or less in line with the best practices for some BUs, the logic of BI is absent in others.
-Several reports, often based on the same data, are generated in a decentralized way in different solutions in rudimentary form.
-Customer information, organized by line of business is fragmented at company level. It is currently hard for the organization to know basic information such as number of unique customers, their ownership and their complete profile.

-Delays in information system loading time, coupled with widely varying retrieval date from one system to another, impede the timely delivery of data. This misalignment also causes problems when reconciliation data and causes many reconciliation efforts by users.

The business unit has a legacy tool called “extractions menu” that runs SQL queries to extract data in Excel format with the limitations explained above.

2. Assumptions/Constraints/Risks

2.1. Assumptions

The business need driving the development of a business intelligence infrastructure is to help the analysts and the management to better use data in order to drive growth and customer awareness and analysis. The analysts that consume Data (staff) are financial analysts, market researchers and customer relationship managers who need to get access to automated dashboards and reports to monitor the business, provide recommendations to the leadership or to plan sponsors (employers) and identify growth opportunities (increase participants enrollment and engagement, cross-selling of other financial products, increase assets retention...etc.). The existing process of data extraction, manual combination and analysis using Microsoft office is time consuming and inefficient and hence does not fit with the end-users needs.

The business unit has a small team of SQL developers that maintain the extractions menu discussed previously and the existing Oracle database as well as two core system experts who understand the different functionalities related to the existing infrastructure.

2.2. Constraints

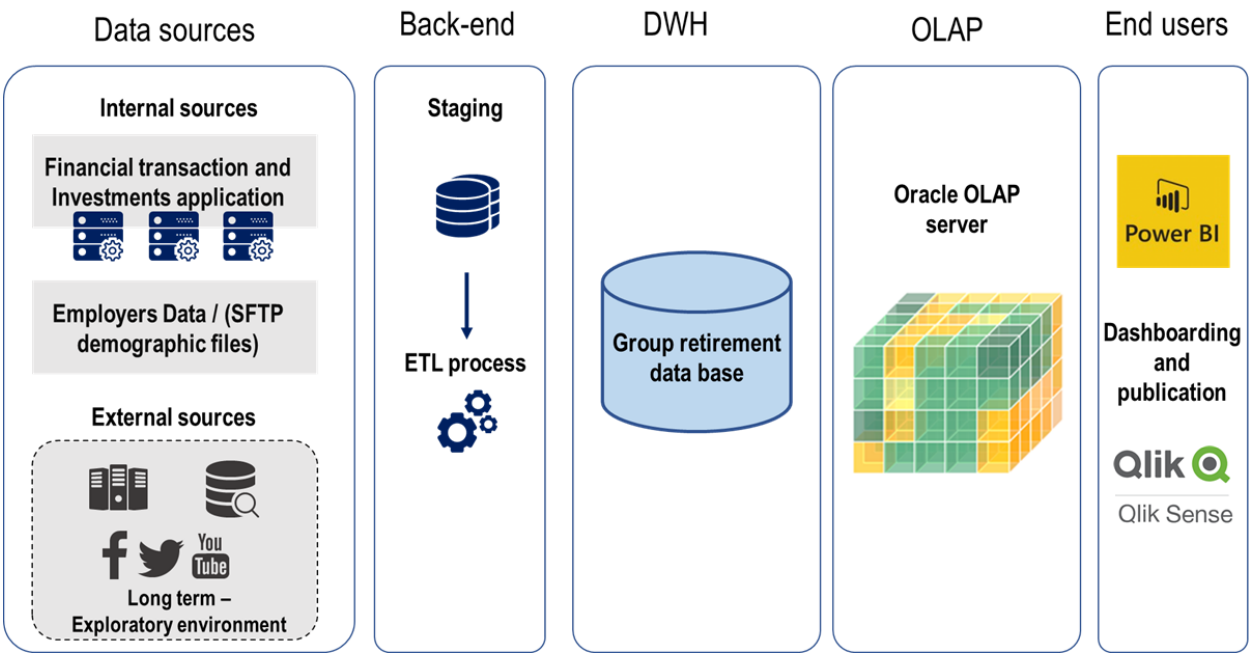
The main constraint that may have significant impact on the database design would be related to the technical skills of the existing resources. The team is composed of SQL developers who understand how the existing Oracle database is designed but may have some challenges if the new infrastructure is NoSQL based.

2.3. Risks

The main risk is related to the storage and use of participant’s personal information. Knowing that retirement plans are registered and shared with the government, participants have to share their personal data such Social Insurance Number (SIN) with the financial institution in order to produce financial statement and other government related documents. Although SIN is not used for analytical purposes, it is stored in the database. To ensure that personal information are confidentially stored, SIN will be encrypted using a Triple Data Encryption Algorithm (TDEA) in the ETL phase.

On the other hand, data access is managed to provide targeted access to the required data based on the need and the tasks to be performed. Finally, any access to sensitive data will be traced (audited).

3. Design Decisions



3.1. Key Factors Influencing Design

The key factors influencing the functional design are mainly business driven:

-Reporting short-term objectives: Provide end-users with “codeless” user interface and data to help them create dashboards and reports and measure performance.

-Advanced analytics long term objectives: The database will provide an exploratory (flexible) environment in order to allow non conventional analysis and exploratory methodologies without impacting the robust one designed for reporting purpose.

-We choose a relational database since the aim of the project is to produce dashboards and reports to help decision-making and to provide a precise overview on the main dimensions of the business. The fact that we have a team of SQL developers made this decision easier to take.

3.2. Functional Design Decisions

The database will provide two environments:

The Robust environment (covered in this document): The robust environment is a set of capacities (infrastructures, tools, informational components) aiming to prepare (acquisition, quality management, data transformation, aggregations by subject, documentation...) the data identified to be consumed (reports, KPIs, TBD...) by the end users according to their access rights.

This is an environment involving process that relies exclusively on data from qualified sources. It is subject to all governance rules in terms of use and security.

Users of this environment will interact with data using Microsoft Power BI in order to create dashboards and reports based on predefined measures and dimensions. It offers a “codeless” user interface to design and prepare data. Also, DAX (Data Analysis Expressions) used in Power BI is intuitive for the end-users and is closer to basic Excel language.

In the long run, the customer relationship manager may use also QlikView since it allows to create analytical applications and to publish dashboards to an external environment, for example to their plan sponsors (employers).

The exploratory environment (not covered in this document): The exploratory environment aims to provide specialists (statisticians, data scientists, expert users, etc.) with a set of capabilities to carry out exploratory, advanced analytics and discoveries on data that can be voluminous and unstructured. It consists in setting up an environment equipped with specialized statistical software (example: SAS, R, Python ...), in disk space to accommodate large volumes of data, in computing capacities to handle these large volumes and in data from any source the organization may need (eg robust environment data, social media, Stat Canada, Web analytics ...).

3.3. Database Management System Decisions

The business unit decided to use Oracle technology for Data Warehousing. The main reason is related to the experience of the existing team with Oracle since the same provider has been used to store core system data. The second reason is related to the technical support, the supplier offered to support the DWH project and the migration plan.

3.4. Security and Privacy Design Decisions

There will be a list of usernames. A valid username and password must be used to access the data warehouse. The DBA will maintain this list of users and prompt a change of password every three months for added security.

The DBA will also assign roles to the users. These roles will grant and limit privileges of the user.

Full access: DBA

Unrestricted resource use

Ability to create tables and update schema as well as delete privileges.

Query access: Data analysts

Ability to access, view, and query data warehouse.

Resource use is restricted

View access: Department staff

Ability to view predetermined queries made by data analysts for reporting purposes only. All sensitive information such as specific identifying information is not accessible.

Resource use is severely restricted

3.5. Performance and Maintenance Design Decisions

Service Level Agreement

The data warehouse is expected to be available except during scheduled maintenance hours once a month. The DBA and the SA will be available during business hours 9am to 5pm EST Monday to Friday for any questions or issues. Customer support tickets will be reviewed during business hours but can be created any time.

Backup Strategy

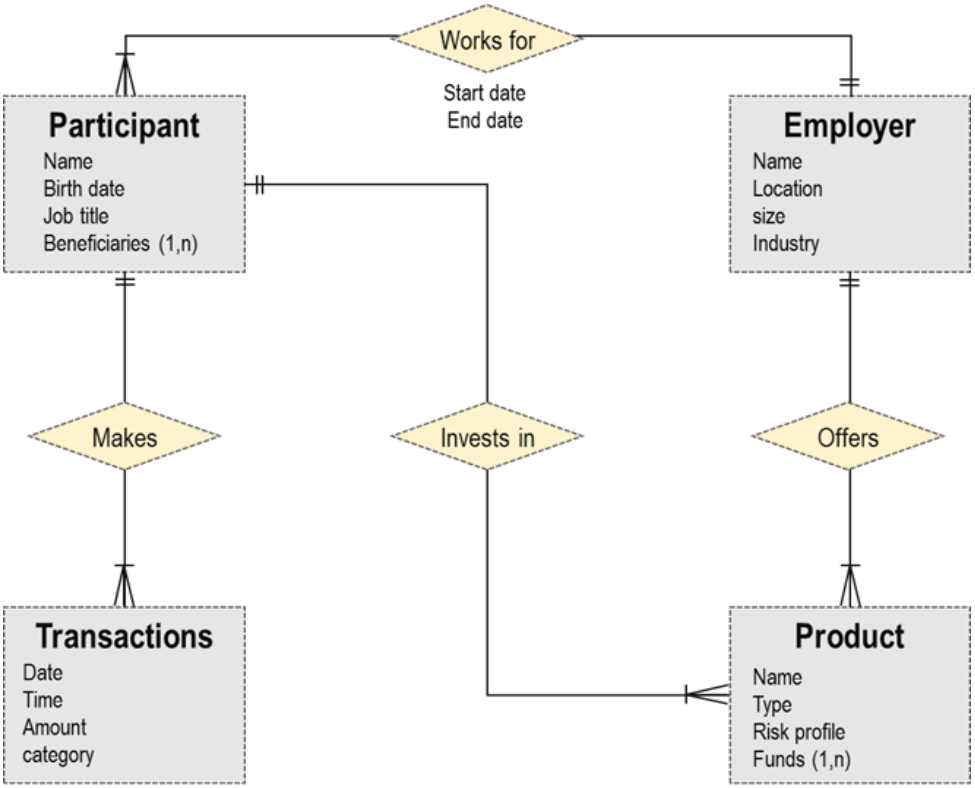
The backup strategy will follow the Oracle ETL backup strategy. Backups of the data warehouse will be made monthly. The extraction of the data from the sources will occur daily in the staging area and this data in the staging area will be backed up. These files will be kept until they are loaded into the data warehouse following the ETL process. In a recovery scenario, the backup of the data warehouse would be loaded and the data in the staging area would be processed again to bring the data warehouse up to date as usual. Thus, the data in the staging area is key because it must be stored and backed up so that new data is not lost in case of failure prior to updating the data warehouse.

4. Detailed Database Design

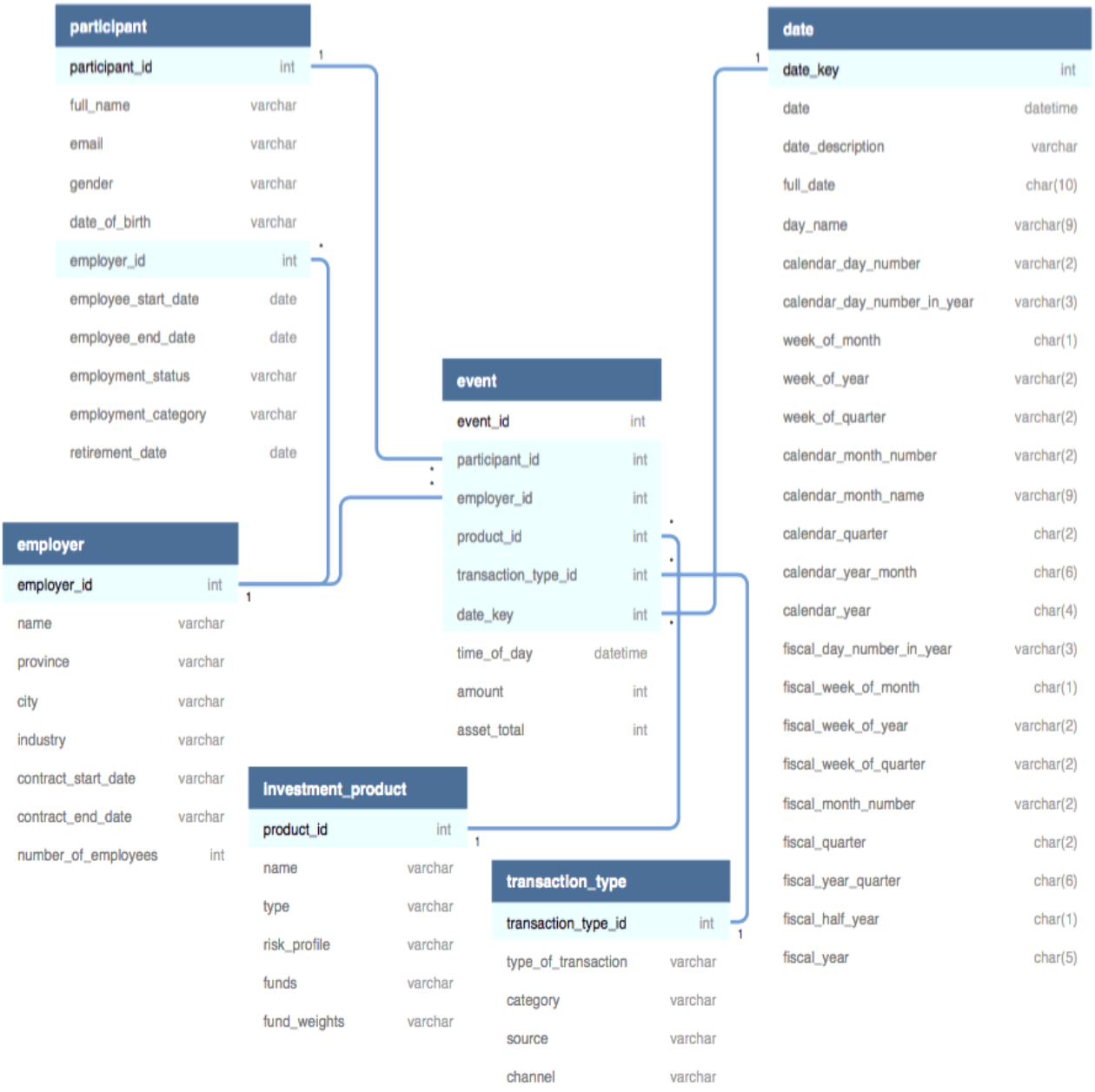
Given the relatively small size of the data warehouse (see estimated size section) the business will have no problem storing data for decades for an unforeseen period into the future in order to produce long-view historical performance reports. The long-term nature of the business and the outlook of employers offering retirement plans means the monthly transactions are not expected to grow in a manner that will become impossible to manage. Thus, an archiving strategy is not necessary for the foreseeable future.

The data temperature can be classed as warm since it will be accessed more than once a month for reporting purposes and the data warehouse will be stored using SSD. The SSD is smaller, faster, quieter, and uses less energy than HDD. The capacity of SSDs has also caught up with HDD. Given the expected size and growth of the data warehouse, the current higher price of SSD technology is negligible. It is expected that the drives will have to be swapped out near the end of their life cycles to prevent hardware failures. HDDs and SSDs both have positives and negatives in terms of durability so this was also not a concern in choosing SSDs for their performance.

Conceptual Data Model



Logical Data Model



Physical Data Model (PDM)

Refer to Appendix B: DDLs

Size Estimate

Estimated size of Data Warehouse based on fact table size

INT type = 4 bytes

DATETIME type = 8 bytes

Size of row = (8 columns * 4) + 8 = 40 bytes

Estimated transactions per month = 150,000

Growth per month (bytes) = 6,000,000 bytes

Growth per month (MB) = 6,000,000 / (1024*1024) = 5.7 MB

Growth per year = 69 MB per year

10 year estimate = 690 MB

4.1. Roles and Responsibilities

As discussed before, the business has extensive personnel with an advanced SQL skill set and Oracle database experience.

The DBA will come from within the business group itself and will come equipped with domain knowledge as well as database management experience. They will be responsible for the monthly update of the data warehouse and the ETL process from the operational database to the warehouse. Experience with scheduling software is important to automate tasks and efficiently update the system. The DBA is a key player in maintaining and making sure the data warehouse runs smoothly and is equipped for use by the analysts.

The system administrator will be in charge of granting permissions to the business groups, and their analysts, as well as maintaining any views that are created. Any requests for access will be made to the SA. This is an important role as the security and access policies will be managed and enforced by the SA. Additionally, they will oversee performance as well as backup procedures and troubleshooting as it arises. Monitoring log files and performing disaster recovery are also key tasks.

4.2. Performance Monitoring and Database Efficiency

4.2.1. Operational Implications

The process will take place the first Sunday of every month at 3am EST. The process will be initiated by scheduling software and the system will be down for maintenance until completion.

Updates will occur once a month. More frequent updates are not needed for business analysis. One month is adequate as it flows naturally from monthly statements sent to participants. Although the updates are monthly, the grain of the fact table will be far more detailed, as all customer transactions will be captured. The daily transactions will be extracted from the operational database into the staging area nightly and stored there until the data warehouse is

updated every month. This daily data will not undergo the ETL process until the full batch is ready each month.

4.2.2. Data Transfer Requirements

The sources for the data warehouse will be an existing Oracle database for operational processes. As such, the cleaning process is straightforward considering the same departmental codes and formats will be adhered to in both the source and destination. A table-to-table ETL will occur with unnecessary operational data will be removed. The relevant operational data will be mapped onto the data warehouse schema. Updates to the dimension tables will be made before any data is added to the fact table so that any new customers or products will be captured.

Update frequency:

The dimensions of the model are relatively stable. They will be updated in the same maintenance window as the transactions, that is to say, monthly. This includes the participant table, which is subject to changes in employment status, salary, and employer.

We will be using type 2 slowly changing dimensions in order to correctly represent history. With type 2 SCD, we typically do not have to reprocess OLAP cubes or revisit preexisting aggregation tables. Type 2 is also the safest choice since if we were to choose type 1, an overwriting approach, it would take tremendous effort to make the change to type 2 should we decide in the future that it better suits our purpose. Moving from type 2 to type 1, however, would not pose a significant problem.

4.2.3. Data Formats

In lieu of a data dictionary, see DDLs for data types and lengths. As discussed, both sending and receiving systems will be relational tables which mitigates problems moving from source to data warehouse.

Appendix A: Acronyms

Table - Acronyms

Acronym	Literal Translation
DBA	Database administrator
ETL	Extract, Transform, Load
HDD	Hard disk drive
INT	Integer
MB	Megabyte
OLAP	Online Analytical Processing
RRSP	Registered retirement saving plan
SIN	Social Insurance Number
SA	System Administrator
SCD	Slowly Changing Dimension
SSD	Solid state drive

Appendix B: DDLs

```
CREATE TABLE Participant (
  participant_id integer primary key identity,
  full_name varchar,
  email varchar,
  gender varchar,
  date_of_birth varchar,
  employer_id integer,
  employee_start_date date,
  employee_end_date date,
  employment_status varchar,
  employment_category varchar,
  retirement_date date
  FOREIGN KEY (employer_id) REFERENCES Employer(employer_id)
);

CREATE TABLE Employer (
  employer_id integer primary key identity,
  name varchar,
  province varchar,
  city varchar,
  industry varchar,
  contract_start_date varchar,
  contract_end_date varchar,
  number_of_employees integer
);

 CREATE TABLE Transaction_type (
  transaction_type_id integer primary key identity,
  type_of_transaction varchar,
```

```
  category varchar,
  source varchar,
  channel varchar
);

CREATE TABLE Investment_product (
  product_id integer primary key identity,
  name varchar,
  type varchar,
  risk_profile varchar,
  funds varchar,
  fund_weights varchar
);

CREATE TABLE Date (
  date_key integer primary key,
  date date,
  date_description varchar,
  full_date char(10),
  day_name varchar(9),
  calendar_day_number varchar(2),
  calendar_day_number_in_year varchar(3),
  week_of_month char(1),
  week_of_year varchar(2),
  week_of_quarter varchar(2),
  calendar_month_number varchar(2),
  calendar_month_name varchar(9),
  calendar_quarter char(2),
  calendar_year_month char(6),
  calendar_year char(4),
  fiscal_day_number_in_year varchar(3),
```

```
  fiscal_week_of_month char(1),
  fiscal_week_of_year varchar(2),
  fiscal_week_of_quarter varchar(2),
  fiscal_month_number varchar(2),
  fiscal_quarter char(2),
  fiscal_year_quarter char(6),
  fiscal_half_year char(1),
  fiscal_year char(5)
);

CREATE TABLE Event (
  event_id integer primary key identity,
  FOREIGN KEY (participant_id) REFERENCES Participant(participant_ID),
  FOREIGN KEY (employer_id) REFERENCES Employer(employer_id) ,
  FOREIGN KEY (product_id) REFERENCES Investment_product(product_id),
  FOREIGN KEY (transaction_type_id) REFERENCES Transaction_type(transaction_type_id),
  FOREIGN KEY (date_key) REFERENCES Date(date_key),
  time_of_day datetime,
  amount integer,
  asset_total integer
);
```