Data Warehouse Modelling Project

Group 0-1-8  
Mrad Azoury, Juan Pablo Garcia, Martin Hofbauer, Sheena Miles, Caterina Selman,   
Jonathan Serrano,Victor Vu

**TABLE OF CONTENTS**

Table of contents 2

Analysis Overview 3

Assumptions 3

REdundant or consolidated variables 3

Variables with Missing values 3

Merged Variables 5

deleted variables 5

Data warehouse selection explanation 6

data warehouse design 8

The model 8

explanations 8

references 9

Appendix 9

**Analysis Overview**

The dataset analyzed provides the status of H-2A visa cases in the United States from the years 2015-2017. Based on the assumption that the stakeholder of this data is the US Citizenship and Immigration Services (USCIS), we understand the value in this data to be the ability to analyze historical trends in visa applications related to their employers and job descriptions, as well as the ability to relate these visa case attributes to the processing time of each visa case.

In the United States, employers can apply for H-2A visas to bring foreigners to the country to work in agricultural jobs on a temporary basis. The process of applying for these certifications is done through a State Workforce Agency, as described in the *H-2A Program Employer Handbook*. A single H2A visa case number is composed of a number of employer applications requesting employees for the same job specifications (i.e. crop, worksite location, job descriptions, hours, pay, etc.).

Once the case is submitted to the USCIS, it goes through a standardized approval process. After the case is “*Received*” and a received date is recorded, a decision of “*Certified*” or “*Denied*” is made, and the respective decision date is recorded. Alternatively, the case could be changed to the status “*Withdrawn*”. These defined states of the visa application workflow are congruent with the business process we are representing in our data warehouse structure.

From this business process, we believe that the USCIS would benefit from understanding the general time frame between when a case is received to when a decision is made. Other useful business cases could be to easily look for cases that have been received, but have not been yet processed, or to view the number of visa cases that are certified and/or denied for a particular job description or employer description. For these purposes, we have designed a data warehouse that offers easy access to this information.

**Assumptions**

* The stakeholder of this data is the USCIS (US Citizenship and Immigration Services), the agency that stores the data in order to analyze all details of past visa cases, i.e. employers, job descriptions, processing times, etc.
* Processing dates, status and job descriptions are all the same for a specific case, regardless of the employer: Although there a few exceptions for this in the data set (where data is missing, or it appears that typos were made) this is the case for Visa case numbers. We are therefore assuming it is part of the business case and workflow of visa applications.

**Redundant or Consolidated Variables**

Variables with Missing values

The dataset contains variables with empty cells. For variables that are either mandatory in the visa application form (see appendix) or binary, a *Cannot be null* constraint has been set. In other words, rows with no values for these attributes are not integrated into the database. For other attributes with missing values, a value might not always be required (e.g. law firm), or the variable was created in a later dataset which justifies the high number of missing values. These are kept in the dataset/warehouse.

*Interpretation of the table: variable (% of missing values over the 3 datasets).*

|  |  |
| --- | --- |
| **Set a *cannot be null constraint*** | **Set a *can be null* constraint** |
| Certification\_begin\_date (0.1%) | Lawfirm\_name (33.7%) |
| Certification\_end\_date (0.1%) | Agent\_attorney\_city (33.6%) |
| Employer\_address1 (0.1%) | Overtime\_rate\_from (97.0%) |
| Employer\_city (0.1%) | Overtime\_rate\_to (98.2%) |
| Employer\_state (0.2%) | Supervise\_how\_many (99.7%) |
| Employer\_postal\_code (0.2%) | Num\_months\_training (99.5%) |
| Employer\_phone (0.1%) | Name\_reqd\_training (99.7%) |
| Job\_title (<0.1%) | Emp\_exp\_num\_months (20.3%) |
| Soc\_title (0.1%) | Trade\_name\_dba (96.3%) |
| Primary Crop (3.5%) |  |
| Nbr\_workers\_requested (0.2%) |  |
| Nbr\_workers\_certified (0.2%) |  |
| Nature\_of\_temporary\_need (0.1%) |  |
| Basic\_number\_of\_hours (0.1%) |  |
| Hourly\_work\_schedule\_am (0.2%) |  |
| Hourly\_work\_schedule\_pm (0.2%) |  |
| Basic rate of pay (<0.1%) |  |
| Basic\_unit\_of\_pay (0.1%) |  |
| Supervise\_other\_emp (0.1%) |  |
| Training\_req (<0.1%) |  |
| Emp\_experience\_reqd (<0.1%) |  |
| Worksite\_city (<0.1%) |  |
| Swa\_name (1.5%) |  |
| Job\_idnumber (2.3%) |  |
| Job\_start\_date (3.8%) |  |
| Job\_end\_date (3.8%) |  |
| Agent\_attorney\_name (33.2%)\* |  |
| Primary\_sub (29.7%)\* |  |
| Organizational\_flag (24.5%)\* |  |

\*Deleting rows with missing values from agent\_attorney\_name, primary\_sub and organizational\_flag would entail deleting a large percentage of the total dataset. Therefore, for these attributes, a *cannot be null* constraint is set going forward, but past empty values will be kept.

Merged variables

Variables’ names are not consistent across datasets; therefore, the below variables have been merged:

* worksite\_state **with** worksite\_st
* worksite\_postal\_code **with** worksite\_ZIP
* case\_number **with** case\_no
* naics\_code **with** naic\_code
* full\_time\_position **with** full\_time
* soc\_code\_id **with** soc\_code

Similarly, values with the same data as another have been merged.

* employer\_address\_1 **with** employer\_address\_2

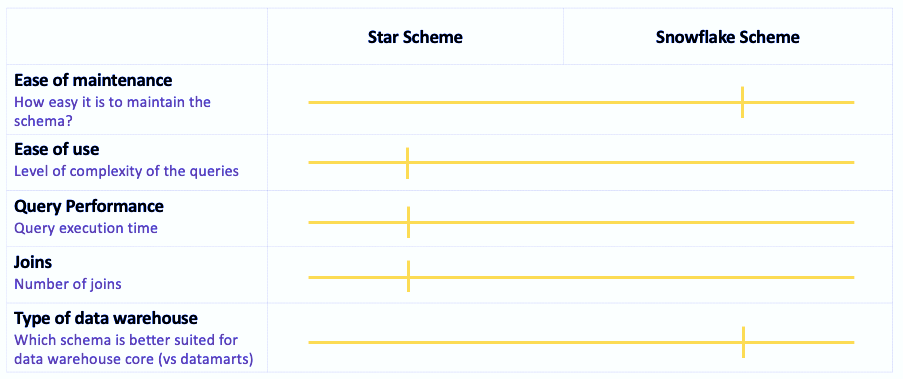
The merged variables above contain missing values, however, a *cannot be null* constraint is set in the model.

Deleted variables

|  |  |
| --- | --- |
| **Variable** | **Justification** |
| employer\_province | Redundant with other address information. Field not mandatory in the form. Furthermore, 92.2% of values are missing and among the remaining 7.8%, there are several instances of ‘N/A’. |
| employer\_phone\_ext | Field not mandatory in the form. 90.6% of data missing |
| education\_level | Field is mandatory, but the data is not relevant for this particular visa (99% of ‘None’). |
| other\_edu | Data not relevant for this particular visa. |
| second\_diploma | Data not relevant for this particular visa. |
| second\_diploma\_major | Data not relevant for this particular visa. |
| worksite\_county | Despite this field being mandatory in the H-2A visa application form, 62.9% of the values are missing. |
| serial\_id | Data not specific. Another primary key will be created in the database model. |
| Visa\_class | Data not relevant as we are analyzing one specific type of visa (H2A). 100% of the values are “H-2A”. |

**Data Warehouse Selection Explanation**

The table below compares the star scheme and the snowflake scheme based on 5 criteria that are relevant to the selection of our final scheme.

****

While queries are less complex and faster to execute in a star scheme, the snowflake scheme offers more flexibility, which was a key criterion in our data warehouse selection. We expect our database to keep growing and becoming more complex as we receive new H-2A visa applications. Furthermore, the snowflake approach reduces data redundancy. Lastly, the snowflake scheme is better suited to simplify many to many relationships, which we have in our dataset (e.g. many employers to many case numbers).

Considering that the two most important factors in our data warehouse selection process were flexibility and performance, we have decided to use a multidimensional approach, that is, a combination of a star scheme and a snowflake scheme; the starflake.

At first sight, a single dimensional approach would have been best when inspecting the dataset of H2A visas since we have data from one source with the only differentiator being the year. However, when taking a more in-depth look, we were able to outline that values such as case number, for example, have to be normalized and differed to enhance the possibility to better access them and to reduce redundancy. Therefore, the multidimensional approach was determined to be a better fit.

As discussed above in the data analysis section, there are multiple signs of data redundancy, which increases the chances of data integrity problems. These issues would complicate future modifications and maintenance. Meanwhile, the snowflake model uses [*normalized data*](https://searchsqlserver.techtarget.com/definition/normalization), which means that the data is organized inside the database in order to eliminate [*redundancy*](https://searchstorage.techtarget.com/definition/redundant) and helps to reduce the amount of data. The hierarchy of the business and its dimensions are preserved in the data model through referential integrity.

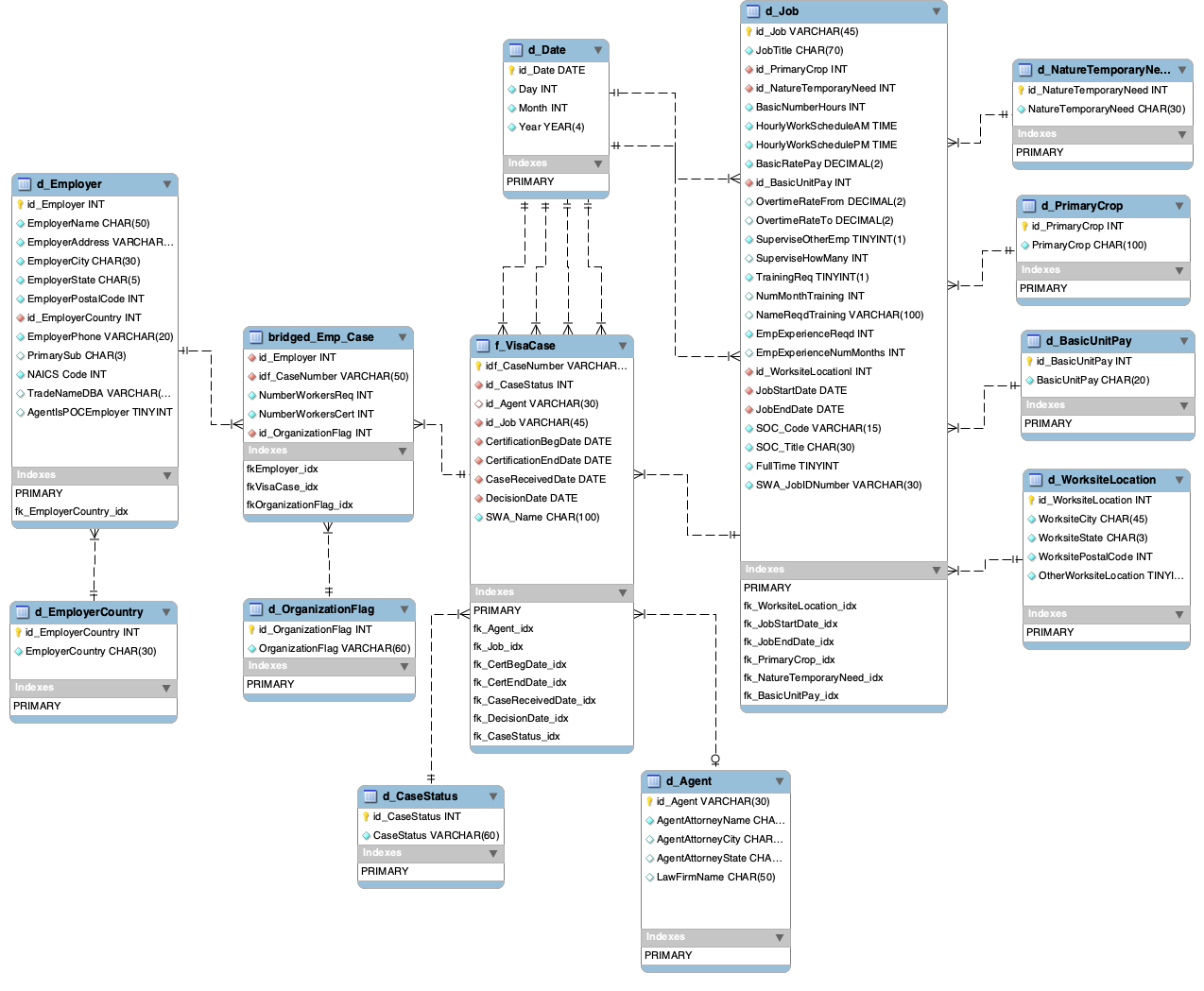
As this model is designed for data warehousing purposes, snowflaking helps reduce storage space. We believe the snowflake schemes will help the users use less space to store all of this data. As a general rule, the more normalized a data warehouse, the fewer redundant records are created. With a dataset of currently 34,000 rows over 3 years, we anticipate that the dataset will grow linearly over the years.

At the same time, certain dimensions of the model follow the star scheme, where dimension tables are denormalized, such as in the Job dimension. Further normalizing this set would make the information redundant and therefore would use storage. This helps reduce query complexity and ensure high performance of the database. Specifically, accessing several information aggregation levels should also be faster as all of them are stored in the same table. It is important to note that the snowflake dimension was built with the minimum number of joins to avoid the need for writing complex queries. For instance, the only join stem out of the Job dimension is the Worksite dimension, because when normalizing the job dimension, a better and clearer picture of the inner relations is produced. Furthermore, additional analyses of the worksite dimension can easily be performed after normalization.

The data vault model was not built for user accessibility. Data vault was built for scalability, flexibility, and accountability on the back-end for the life of the data warehouse. It is more complex to build and requires more time to establish. It is more difficult to translate hubs and satellites into data marts and cubes. Also, it is not ideal to adopt data vault for this data, as we do not require real-time analysis or trend analysis of all the data points.

**Data Warehouse Design**

The Model



Explanations

Specifically, in our starflake scheme we decided to use an Accumulating Snapshot Fact table as the main fact table in our model design. As described in the analysis above, the primary business process in this data set is the current status and status changes of an H-2A visa application case. Therefore, we have designed our fact table to describe these Visa Cases, with their respective attributes, such as the attorney agent and the job for which the visa is being applied. The measure in this fact table is the current visa status and the time it takes to process each visa case.

The most important dimension in our model is the *Employer* dimension. This dimension includes attributes such as the name and contact details of specific employers applying for an H2A visa. Due to the fact that many employers are part of one case, but at the same time one employer could potentially be part of more than one visa case throughout the years, the nature of this relationship is N:M. To represent this we have linked the employer dimension and visa case fact table with employer IDs and visa case IDs as Foreign Keys in a *Bridge Dimension*.

There are other attributes in our model that are only fit to describe a specific visa case/employer combination. Therefore, we have included these attributes in this employer-visa case bridge dimension. These employer-visa case attributes include number of workers required, number of workers certified, and the organizational flag or type of employer application.

Directly related to our fact table are a number of dimension tables. The first one is the *Job* dimension. Each visa case is directly related to 1 job description, therefore we represent this relationship with a 1:N relationship and a foreign *Job ID key* in the fact table. The job table itself includes all of the attributes that describe a specific job, such as the title, the primary crop for which the job is intended, and the work schedule, amongst others. Other job attributes, such as the job worksite location ID have been separated into a different dimension called the *Worksite Location* dimension, linked by a foreign worksite ID, with descriptors such as worksite city, state and postal code.

Another dimension directly linked to the fact table is the *Agent* dimension, because like the job, this is related to the visa cases by 1:N relationship and each visa case. The agent dimension has different attributes such as name, city, and law firm for which they work.

Finally, we have *Date* dimension which is linked to the Visa case fact table as well as other dimensions that reference dates. This dimension serves as a way to standardize date formats throughout the model and also facilitates queries for specific time periods like months or years.

Other additional dimensions in our model were put in place to reduce redundancy of data in the data warehouse. We have created these additional dimensions for attribute values across the model that are always part of a limited set of options and can be easily standardized:

1. **Case status**: Each case is in 1 of 4 statuses, as described above in workflow process, and we do not expect statuses to be changed frequently. Therefore, we have decided to include these in a separate dimension.
2. **Employer Country**: Every employer is described with a country attribute but only 2 countries exist in the data. Since the business case for this data is US H-2A cases, we do not expect countries to be changed frequently and have therefore created a separate dimension for this data.
3. **Crops**: Every job description has a single crop or crop/group. To reduce redundancy, we have separated the existing crop list into a separate dimension. As more crops are inserted into the database, they can be added into this table, however we do not believe this will happen frequently.
4. **Organizational Flag, Nature of Temporary Need, Units of pay**: For these attributes/dimensions we have decided to separate them into a dimension table because they come directly from the Form ETA-9142A. Having these in separate dimension tables will make it easier for the stakeholders to handle the options as the form format changes.

References

United States Department of Labor. (2015). *Small Business Guide to the H-2 Visa Program: Herding or Production of Livestock on the Range*. [PDF File]. Retrieved from <https://www.foreignlaborcert.doleta.gov/pdf/H-2A_Small-Business-Compliance-Guide_Herder_Final_Rule.pdf>

United States Department of Labor. (2016). *H-2A Application for Temporary Employment Certification*. [PDF File]. Retrieved from <https://www.foreignlaborcert.doleta.gov/pdf/ETA_Form_9142A.pdf>

**Appendix 1: H-2 Visa Application Form**