Data Warehousing and NoSQL

Crystal Combs, Matt Luc, Jonathan Quilliams, Charles Wigle

April 26, 2017

CSCI 3020 – 001

Dr. Bennett

*When in the course of Beta University’s events, it becomes necessary for its Development Office to seek and obtain donations from a variety of donors.*

To help the Development Office in their annual fundraising, Suzanne Hayes of Beta University called upon us to create a database for tracking donors, events, employers, donations, and payments. These and other tables would replace the spreadsheet currently keeping track of pledges and donations. Along with creating a database and inserting mock data, we created procedures printing out reports including an annual report to donors, an internal monthly report, a Phonothon Contact List, and a report displaying information on employers which match donors’ donations. Shortly after completion of the reports, it was decided to discuss the future of the project: the addition of data warehousing and NoSQL.

**Data Warehouse**

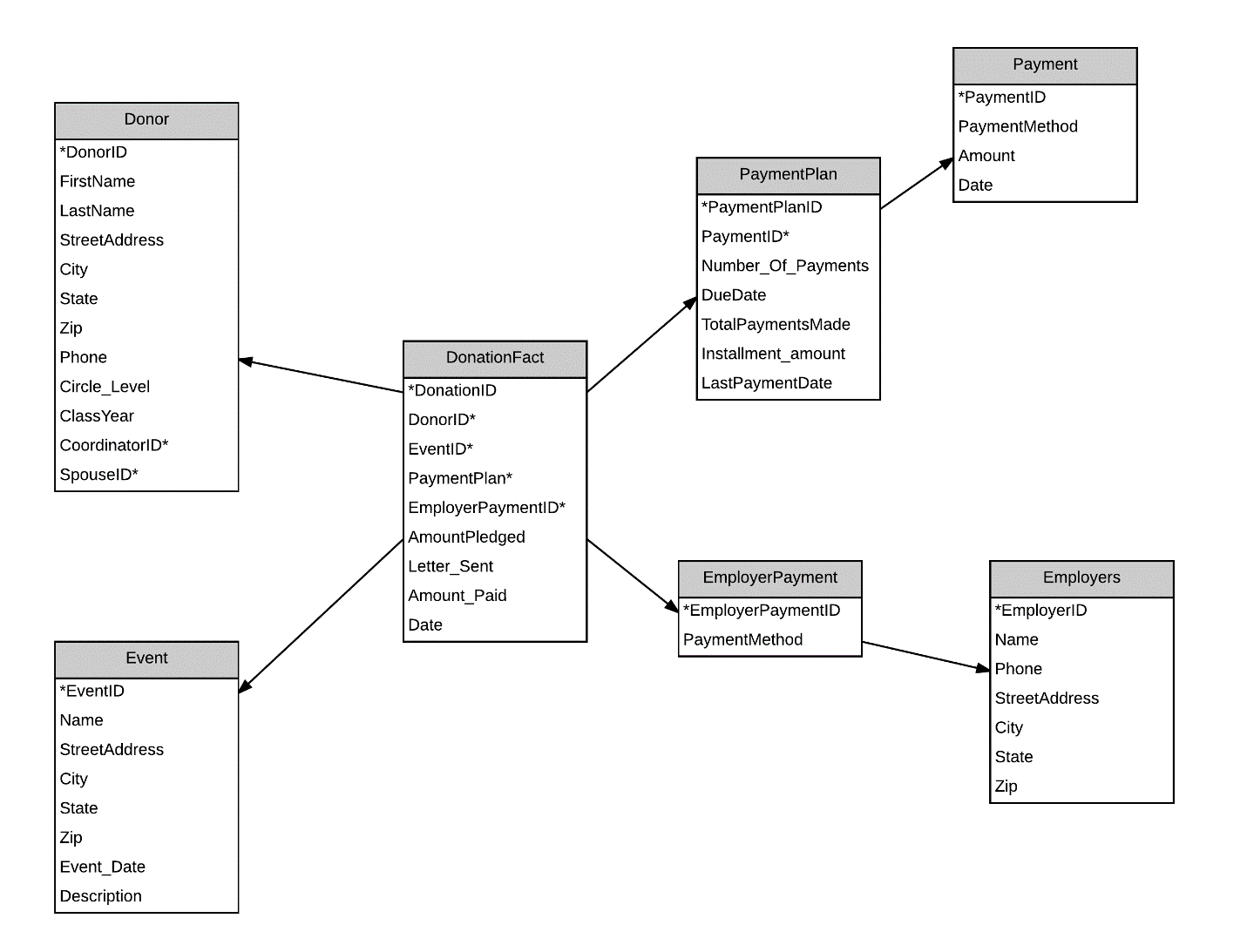


Fig. 1 – Data Warehouse Design

Holding the Phonothon Event is one reason for our need of a data warehouse. Due to the static number of volunteers and previous donors for event(s), having a specific data mart for volunteers would protect the data in two ways. Volunteers do not need access to the entire database, so a data mart would be extenuating. And unlike skimming through an irreplaceable spreadsheet, with a data warehouse there can be security concerning data retention and reducing possible data loss.

A data warehouse would also be appropriate for donor data analytics, as we also believe there are trends to be found within the donation historical data. For example, keeping track of Employers and Donors, we can determine which companies would be best suited for soliciting donations, through its employees. Furthermore, we can tell when to better hold an event so as to get a better amount of donations both numerically and fiscally.

When it comes to creating a data warehouse, there’s usually the process of Extracting Transforming, and Loading to be found. We will extract the data from the donor, donation, event, employerPayment, employers, paymentPlan, and payment tables, with the donor as the fact table. Address fields will go into their respective tables of donor, event, and employers. (See Fig. 1) During the process of ETL, we will need to disable the loading process and to then re-enable after. We also expect the ETL process to run on a weekly basis for maintenance from operational systems and should update after an event is held to secure a larger amount of information.

**NoSQL**

Now after deciding to use a data warehouse, this leaves the question as to whether decide on a SQL or NoSQL option of database. We have decided to go with a MongoDB, NoSQL database. Our reasoning: There are more advantages to MongoDB than disadvantages. Helpful when it comes to data retention through the use of Auto-sharding. It’s even more scalable than regular SQL. The hardest part is taking the time to learn it.

Our plan for this Data Warehouse: We expect the donor table to be nested up top being the collection. The next level will be the tables donor, event, employerPayment, and paymentPlan. Payment would then be nested under paymentPlan and employers under EmployerPayment. Under the employers, donor, and event tables would be nested addresses.

We expect these documents to contain the same data as the fields in Fig. 1, just layered differently. If we decided not to go with MongoDB in the near future but would decide to do so later, it would take a while to transfer the data over to the new DB.

**Conclusion**

Overall, we believe a data warehouse and NoSQL for our project to be good investments. A Data warehouse is good because we are using for both the Phonothon event and data analytics. The use of a NoSQL server as would mean better storage, data retention, and scalability. It also gives flexibility when it comes to adding new fields to records in the near future. In conclusion, the