Teaching Introduction to Database in an Analytics Program

Douglas M Kline
klined@uncw.edu

Jeffrey Cummings
cummingsj@uncw.edu

Emily Lyons
Egl9282@uncw.edu

Information Systems
University of North Carolina Wilmington
Wilmington NC 28403

Abstract

This paper examines how database management has been taught in traditional technology curricula, and how to adapt a database course for a business analytics curriculum. We compare and contrast common established educational scenarios such as formal data modeling for controlled transaction processing systems versus high volume data capture from various internal and external sources for later analysis. Traditional database topics that are useful to keep are identified, along with those that may not be useful, and new topics that are very useful for analytics. A manufacturing process analogy is applied to data to help students better understand data forms, operations, and processes.

Keywords: curriculum, database, analytics, structured query language, data processing

1. INTRODUCTION

Database management is a common core course for many technology curricula. Database is an important stand-alone course, but many curricula revolve around analyzing and building archetypal Transaction Processing Systems (TPS), where a centralized relational database is foundational. Of course, TPSes are still core systems for most organizations, but databases and related technologies are useful in far more situations than just core TPSes.

The topics covered in traditional database courses focus on teaching students how to design and construct TPS-style software. Important topics in this context include data modeling, normalization, transactions and ACID requirements, data definition language (DDL) portions of structured query language (SQL), full data manipulation language (DML) portions of SQL, and database programming constructs like stored procedures. The underlying context is that *we* are building the system, we are heavily analyzing our needs in advance to define the data model, and we have control over the creation of the data.

New curricula in Data Science and Data Analytics are including database as a core topic. (Mitri & Palocsay, Toward a Model Undergraduate Curriculum for Emerging Business Intelligence and Analytics Discipline, 2015) place database topics in the Business Information Intelligence Knowledge Area, along with programming, implying the context of design and construction

of an information system. Others (Mamonov, Misra, & Jain, 2015)place database in the Technical Data Management Skills and emphasize that practical business analytics is best served with SQL facility rather than relational modeling.

The context in data analytics is not construction of a TPS. A more common scenario is that an analyst is given access to vast amounts of raw data in many forms from heterogeneous systems. Analysts must use the various tools to wrangle the data into a usable form and produce useful information. The findings of (Mills, Chudoba, & Olsen, IS Programs Responding to Industry Demands for Data Scientists: A Comparison between 2011-2016, 2016)indicate that IS programs are responding to industry demands by offering advanced database classes.

The need to change database curriculum is being recognized in information systems education literature. Some authors are including case studies that are not strictly TPS-oriented in nature, and include analysis techniques (Mitri, Active Learning via a Sample Database: The Case of Microsoft's Adventure Works, 2015). Other authors (Mills, Dupin-Bryant, & Olsen, Designing Database Modules using Correlation Coefficients and Linear Regression: A Content-Centered Approach, 2015) start with an analytics problem, and work backward to writing the necessary SQL statements to get the needed data. The teaching module presented by (Wang & Wang, 2019) does an admirable job of covering minimal traditional relational terminology before diving business intelligence and analytics.

A traditional relational database management course designed for a traditional information systems (IS) curriculum does not fully address the needs of future data analytics professionals. Some topics adapt directly from IS to Analytics, while some are less relevant. New topics need to be added for analytics. The good news is that database can be a topic that is attractive to all business majors (Wang & Wang, Renewal of Classics: Database Technology for All Business Majors, 2016), not only business analytics and information systems.

This paper relates the concerns, reasoning, and result of creating a "Database for Analytics" course. The course is introductory in nature, with no prerequisites other than basic computer skills. The discussion should be useful for those creating similar courses, and also for faculty whose course may need to serve traditional IS programs as well as analytics programs.

Section 2 describes traditional scenarios based on TPS design and construction and contrasts this with analytics scenarios. Section 3 details the many advantages of relational database technologies that make them desirable in an analytics context. Section 4 lists specific traditional database topics and whether they were included in the analytics course. Further, new relevant database topics are identified. Section 5 presents a "manufacturing" analogy that may be helpful in relating the many data technologies, stages, forms, and processes in today's systems. The final section summarizes key concepts and offers advice to faculty making this transition.

2. SCENARIOS

Traditional

Many traditional IS (and other) curriculum prepare students to design and construct a transaction processing system (TPS). This is represented in the IS 2010.2 Data and Information Management learning objectives from **(Topi, et al., 2010)**, which includes "design high-quality relational database" and "normalizing". This set of skills is extremely valuable, as nearly every organization must record the basic transactions of its day-to-day operations. However, this is a very specific type of system with very specific requirements and many assumptions. Non-IS students can find normalization and modeling to be too advanced and decreases the accessibility of a database class to non-IS students. **(Wang & Wang, Renewal of Classics: Database Technology for All Business Majors, 2016)**

In general, the type of system that curricula focus on has these characteristics:

- N-tier with at least these layers:
  - Data layer
  - Presentation layer (user interface)
- Full systems analysis with data model
- All operations: INSERT, UPDATE, DELETE, SELECT
- Transactions with ACID requirements
- Full ownership of the system and data
- Primary concern is accurate capture of transactions
- Volume of transactions low enough to make data quality and consistency checks feasible

This scenario is the perfect use case for a relational database management system. Relational Database technologies were purpose-built to solve these problems.

A TPS designed with a thorough systems analysis and well-normalized data model is highly organized and exhibits tight control on the data. Huge efforts are put towards analyzing the entities, how they relate, and what operations are allowed. An entire language representing the system domain is created with nouns and verbs particular to the domain. The database schema is a rich, carefully crafted artifact representing detailed knowledge of the data and its nature. Data types, check constraints, entity integrity and referential integrity slow down basic operations, but ensure high quality data.

In summary, the traditional scenario refuses to admit any data that does not conform to pre-defined type, domain, format, and consistency checks. The work of organizing the data is done at capture. Data is "normalized", i.e., independent of any single use. This makes it flexible and easily adaptable to new purposes. The cost is the high amount of systems analysis required and the relatively slow data operations.

Curricula based on this traditional TPS scenario rightfully focus on data modeling, transactions, data integrity, and all the DML and DDL parts of SQL.

Analytics

Analytics scenarios are quite different than the above situation, with different requirements and concerns. Common characteristics include:

- Limited or no ownership of systems that create the data
- Heterogeneous internal and external data sources
- Variable degrees of systems analysis
- Variable data quality
- TPS data sources as well as other sources
- Some sources are INSERT-only with no checks
- High volume – quality checks are infeasible
- Deferred organization
- Data pipelines/processes/stages
- Multiple formats and technologies

Modern organizes tend to keep all data, without full knowledge of how or if they will use it. It is sometimes impractical or infeasible to check data as it enters, due to volume, velocity, and an insufficient knowledge of the data.

Curricula based on this scenario may do better to have less emphasis on systems analysis, data modeling, and transactional integrity. More emphasis might be placed on thorough knowledge of the SELECT statement, import/export, data cleansing, and work flows for data.

3. RDBMS BENEFITS

Relational databases are well-established, mature, rock-solid foundational platforms for today's information systems. Because of this, they have many important advantages for data analytics. These include:

- Tabular in nature – intuitive
- Wide variety of established, full-featured, stable products
- Integrations into and out of almost all formats
- Integrations into and out of almost all systems
- Highly performant, mature storage engine
- De-facto standard data manipulation language: SQL
- Many data types and facilities for manipulating them

Tables tend to be easy to understand and intuitive to most people. Even if data does not start out as a table (for example, a linguistic corpus), it is common to transform it into a table for analysis (for example, a term frequency matrix).

Modern enterprise database products provide easy storage and full manipulation of many types of data including:

- Dates and times
- Geolocation
- XML (and variants such as JSON, HTML, etc.)
- Binary large objects
- Files, documents
- Columnar data

In addition, relational database storage engines have very high performance. They are highly adjustable and adaptable to many different situations.

Because of these benefits, relational databases are commonly used in situations highly unlike

TPSes. In other words, a situation might not *require* a relational database, but a relational database is just very convenient for the situation.

4. DATABASE TOPICS

Here is a list of database topics that were included in the more traditional Information Systems relational database course that were retained in the database for analytics course:

- Data types
  - Numeric
  - Character
  - Date and Time
  - JSON
- Primary Keys, Foreign Keys, Relationships
- SELECT and its rich variations

Here are topics that were NOT included in the analytics course:

- INSERT, UPDATE, DELETE
- Modeling and Normalization
- Programming constructs, e.g., Stored Procedures
- Indexes

Here are topics that were added to the analytics course that were not in the traditional IS database course that were added in the analytics course:

- Forms of non-tabular data
  - Document databases
  - Column stores
  - Hadoop file systems
  - Object stores
  - Key-value stores
  - Etc.
- Other Scenarios (not TPS)
  - Internet of things, event hubs
  - Streaming data
  - Data lakes
- Moving data – Import/Export
- SQL Ranking functions (NTILE, RANK, etc.)
- Data wrangling tasks
  - Detecting and handling nulls
  - Detecting and handling widow and orphan records
  - SQL data type conversions
  - SQL CASE statement
  - Changing formats, e.g. tabular to JSON

5. MANUFACTURING ANALOGY

A manufacturing analogy is helpful in presenting the scenarios more common to analytics and non-TPS data processing. TPS processing is relatively simple: data doesn't get in unless it's organized, then it stays in place or gets updated, then it might get moved to a data warehouse or get archived. It's a fairly linear, consistent path.

Consider a manufacturing process that makes furniture with wood. Below are steps in the process with analogous data processing steps.

- Trees grow in the wild, all over the world
  - Consumers/users create data in the wild, e.g. facebook, twitter, amazon, google
- A lumber company purchases rights to harvest trees
  - Companies purchase rights to data
- Trees are cut down, stacked transported to a lumber mill
  - Data is captured in its original form, sent perhaps to a data lake
- Logs are milled into standard size lumber, dried, etc. Bad logs and wood are thrown out.
  - Data is cleansed and put into a standard format.
- Standard lumber can go to many different manufacturers for many different purposes. It is more valuable than logs.
  - Clean, standard format data can be used for many purposes and is more valuable than data in its original form.
- The furniture manufacturer cuts the lumber into standard lengths and sizes that are good for furniture. They may cut out additional defects. The lumber is still not specific to the use (table, chair, cabinet)
  - A company buys the data, then processes it into a more usable form for their purpose. The data may go into a relational database. It's still flexible and may be used in a number of different ways.
- The furniture manufacturer machines chair parts: legs, seats, etc. The wood is no longer easily usable for other purposes
  - The company puts the data into a data warehouse that supports specific reports. The data is no

longer easily usable for other reports.

6. SUCCESSES AND CONCERNS

This course has been delivered since the initial design. Several subjective successes and concerns were identified through the experience and are related here.

Due to the short format of the course, not all of the additional topics could be covered in depth. The various forms of non-tabular data were stated, but not covered in depth, and no hands-on projects were given. However, the context given by describing the various technologies, data flow scenarios, and their use cases was helpful. The manufacturing analogy was helpful in presenting the concept of data in motion, the goal of increasing value through the process, and the uncertain "final" use of the data.

Successful new topics included the SQL ranking functions, data types, data type conversions, and various data wrangling tasks. SQL Ranking functions were readily familiar to students as an analytics method, and tied to non-parametric statistical methods. Because of the various backgrounds of students, the concept of data types and converting between data types was new and valuable to many students. The data conversions and wrangling of dates and times was particularly useful, as these are prevalent in many forms of data.

While unstructured data is notable because of its huge volume, structured data is still very valuable to organizations, and can provide context and structure to unstructured data it relates to. In covering methods for handling unstructured, there is a concern that dirty, error-filled, disorganized data is "normal", and perhaps acceptable. Given the mainly tabular and SQL-oriented focus of the course, and the visceral experience of the work involved in data wrangling, students gained an appreciation for the increased value of clean, error-free, organized data.

7. CONCLUSION

This paper has described many of the concerns of adapting a traditional relational database management course to an analytics curriculum. The traditional TPS-building context was explored and contrasted with a data analytics scenario.

The basic features of Relational Database products were discussed along with how they were beneficial in non-TPS scenarios. Specific topics common to both IS and Analytics audiences were listed. Topics specific to IS versus Analytics were also identified.

Finally, a manufacturing process analogy to data processing was described as a possible mechanism to help students understand the complex stages that data can go through as it is processed.

7. REFERENCES

Mamonov, S., Misra, R., & Jain, R. (2015, January). Business Analytics in Practice and in Education: A Competency-based Perspective. Information Systems Education Journal, 13(1), 4-13.

Mills, R. J., Chudoba, K. M., & Olsen, D. H. (2016). IS Programs Responding to Industry Demands for Data Scientists: A Comparison between 2011-2016. *Journal of Information Systems Education, 27*(2), 131-140.

Mills, R. J., Dupin-Bryant, P. A., & Olsen, D. H. (2015). Designing Database Modules using Correlation Coefficients and Linear Regression: A Content-Centered Approach. *Performance Improvement, 54*(6), 20-31.

Mitri, M. (2015). Active Learning via a Sample Database: The Case of Microsoft's Adventure Works. *Journal of Information Systems Education, 26*(3), 177-185.

Mitri, M., & Palocsay, S. (2015). Toward a Model Undergraduate Curriculum for Emerging Business Intelligence and Analytics Discipline. *Communications of the Association for Information Systems, 37*(31), 651-669.

Shah, V., Kumar, A., & Smart, K. (2018). Moving Forward by Looking Backward: Embracing Pedagogical Principles to Develop an Innovative MSIS Program. *Journal of Information Systems Education, 29*(3), 139-156.

Topi, H., Valacich, J. S., Wright, R. T., Kaiser, K., Nunamaker, J. F., Sipior, J. C., & de Vreede, G. J. (2010). Curriculum Guidelines for Undergraduate Degree Programs in Information Systems. *Communications of the*

Association for Information Systems, 26(1), 18.

Wang, S., & Wang, H. (2016). Renewal of Classics: Database Technology for All Business Majors. *The Journal of Computer Information systems, 56*(3), 211-217.

Wang, S., & Wang, H. (2019). A Teaching Module of Database-Centric Online Analytical Process for MBA Business Analytics Programs. *Journal of Information Systems Education, 30*(1), 19-26.