An Undergraduate Course in Data Warehousing

Jeff Pittges, Radford University

Abstract

The data warehousing field has matured significantly over the past twenty years and data warehouses, which provide the foundation for business intelligence, have become ingrained in corporate America and throughout the scientific community. Data warehousing courses are beginning to emerge in academe, but universities must do more to prepare students for exciting career opportunities not only as producers of data warehouses, but also as competent consumers. This paper describes an undergraduate data warehousing course at Radford University to share our course objectives, provide a model to encourage others to develop similar courses, and to further shape the requirements for data warehousing courses at the undergraduate level.

1 Introduction

As a former member of the Query Processing team at Red Brick Systems, the pioneering data warehouse company founded by Ralph Kimball, the author has closely followed the data warehousing field for the past twenty years. During that time corporate America's perception of data warehousing has evolved from an esoteric pursuit of a few top companies to a competitive advantage and finally to a competitive necessity supporting a wide range of business intelligence applications.

Slazinski, who describes an undergraduate data warehousing course at Purdue [26], claims the data warehousing market has reached 28 billion dollars. While developing a graduate data warehousing and data mining course, Fang and Tuladhar [8] found roughly 900 data warehousing job openings posted on a popular IT job-search website. Murray and Guimaraes [23] acknowledge the pressure to expand the database curriculum and recognize the need to incorporate data warehousing while Dietrich, Urban, and Haag [7] have responded to rapid advances in the field by developing techniques to assess advanced database courses at the undergraduate level. The University of Winnipeg offers four courses in databases including a data warehousing course described by McFadyen [21].

The field of data warehousing has matured sufficiently to support rigorous study at the undergraduate level and the demand for data warehousing professionals is growing. However, it is not enough for our universities to prepare students to develop and maintain data warehouses, we must also prepare students to be competent consumers. Further discussion and additional research is needed to shape the requirements for teaching data warehousing at the undergraduate level. This paper contributes to the requirements definition process by describing various aspects of an undergraduate data warehousing course at Radford University including the course objectives, resources, lecture topics, and assignments. The paper discusses the lessons that have been learned thus far by providing student feedback, recommendations, and future directions for the course. Finally, this paper presents a model to encourage others to develop similar courses.

2 Course Context

The Information Technology department at Radford University offers Bachelor of Science degrees in Computer Science (CS) and Information Systems (IS). Students must choose from one

of seven concentrations; four in the CS program and three in the IS program. Each concentration includes three courses in the area of specialization. The database concentration includes an introductory course (Database I), an advanced course (Database II), and a course on Data Warehousing, Data Mining, and Reporting. This paper primarily describes the data warehousing portion of the third course. Database I is the only prerequisite for the data warehousing course which makes the course attractive to IS majors looking for a technical elective that is not based on software engineering. Consequently, the content of the course must balance the business focus of the IS students with the more theoretical interests of the CS students.

The purpose of a data warehouse is to help organizations make better decisions and one of the primary themes throughout the course is the business value of data. Within this context, all of the course topics are presented from three perspectives: (1) an engineer designing and developing a warehouse, (2) a business or data analyst, and (3) an executive decision maker. The success of a data warehouse depends on how well the warehouse supports the end user. Consequently, the CS students are encouraged to focus on the needs of the business to improve their effectiveness as engineers and the IS students are encouraged to focus on the technical details to become more effective consumers and stronger advocates of the warehouse.

The course is primarily based on resources from Ralph Kimball and his associates at the Kimball Group [20]. However, Bill Inmon [10] is considered the father of the data warehouse while Kimball is considered the father of the data mart. Although Kimball's view of the data warehouse as a conglomerate of data marts across the enterprise built on a foundation of conformed dimensions is more practical for large organizations than a single, centralized data warehouse, Inmon and Kimball are largely in agreement [11]. Another key distinction between the two is Kimball's insistance on the dimensional model, a main focus of the course, whereas Inmon favors storing data in 3rd normal form.

2.1 Course Objectives

The data warehousing portion of the course is organized around the following three objectives:

- Ability to design, evaluate, and develop a dimensional data warehouse:
  - Dimensional modeling and star schemas
  - Aggregation
  - Physical schema
  - Extract, Transform, and Load (ETL) process
- Ability to effectively utilize and employ a data warehouse to solve problems
  - Use OLAP applications effectively
  - Understand the limitations of SQL
  - Understand the relationship between data mining, data warehousing, and business intelligence
- Ability to identify and appreciate the complexities of data warehousing:
  - Scale of a data warehousing project
  - Importance of data quality and traceability
  - Political sensitivities and other practical considerations

These objectives will be discussed when the main topics and the assignments are presented in section 3.

2.2 Resources

Although several books have been written on data warehousing, most of the books are directed at practitioners. Kimball's *Data Warehousing Toolkit* [18], the classic text on dimensional data

warehousing, describes data models for 14 applications from retail, inventory, and procurement to order management and customer relationship management. While studying example schemas is an effective way to learn dimensional modeling, and Kimball masterfully chooses applications that highlight key concepts, such as slowly changing dimensions and the additivity of measures, the book is organized around the applications, rather than the concepts, which limits the book's usefulness within the context of a university course.

Consequently, while several text books are recommended, including Kimball's *Data Warehousing Toolkit* [18], *Lifecycle Toolkit* [19], and *ETL Toolkit* [17], along with Olson's book on data quality [24] and Adamson's book on mastering aggregates [1], the primary resource for the course is the Kimball Group website [20] which provides over 120 articles, design tips, webinars, and other online materials describing every aspect of data warehousing. Once the students grasp the big picture, Kimball's articles and design tips provide in-depth coverage of specific topics and these materials are easily referenced throughout the course.

Additional online materials are incorporated into the course including articles by Greenfield [9] and Demarest [6] regarding the political considerations surrounding data warehouses and research papers on aggregate computation [5] and maintenance [3].

2.2.1 Tools

The data warehousing portion of the courses uses two tools: Oracle Warehouse Builder (OWB) [25], the ETL and Metadata Management tool from Oracle, and the Mondrian OLAP server [22]. OWB is an expensive, enterprise class tool, but Oracle's educational license makes the package affordable. The Mondrian OLAP server is open source and freely available.

2.2.2 Server Infrastructure

All three courses in the database concentration are based on the Oracle database. The data warehousing course uses two databases on separate physical servers. The first database supports the Database I and Database II courses and serves as the operational system for the data warehousing course. The second database is used exclusively by the third course and serves as the data warehouse repository. Having two databases on two physically distinct servers gives the students a better feel for an enterprise environment. However, the course could certainly use two database instances running on a single server or even one database instance using separate schemas and/or accounts.

3 Course Organization

This section describes the main topics covered in the course along with an overview of the assignments and exams.

3.1 Dimensional Modeling

The primary objective for the course is to develop proficiency with dimensional modeling and the implementation of Star schemas in a relational database. The dimensional model is a logical design technique that separates facts (measures) into a separate table surrounded by dimensions containing rich attributes that provide the context for evaluating the facts.

According to Kimball [18] [14], "The goal of a data warehouse is to publish the 'right' data and make it easily accessible to decision-makers." A well-designed dimensional model captures the performance measures and descriptive attributes of a business which makes the model intuitive to business users because the database is structured based on the way users think about the business. "Dimensional models are critical to enabling the business to leverage the organization's valuable information assets."

The dimensional model also affords students valuable experience with de-normalization. Fact tables are normalized to the 3rd normal form, but dimension tables are typically *de-normalized* in 2nd normal form. After one or two database classes the student's urge to normalize data models has been ingrained and automatized. Consequently, students struggle with de-normalization and they must fight their compulsion to snowflake their dimension tables. This same effect was reported by Slazinski [26]. However, once the students have mastered the Star schema and they are comfortable with de-normalization they have acquired another tool in their arsenal of data modeling techniques and they have gained a deeper understanding of how and when to apply normalization.

The course begins with an overview of the fundamentals of dimensional modeling and spends the first two or three weeks examining data models from various applications, much like Kimball's *Data Warehouse Toolkit* [18]. Once the student's are comfortable with the Star schema they are introduced to more advanced dimensional modeling techniques like managing slowly changing dimensions.

3.2 Aggregation

"The single most dramatic way to affect performance in a large data warehouse is to provide a proper set of aggregate (summary) records that coexist with the primary base records" [13]. Materialized aggregate views precompute data at various levels of summarization. For example, a retail warehouse typically records each order line in a Sales fact table. The Sales table is a base table that contains the most atomic level of detail. A series of aggregate views might be created to summarize the sales data by month, quarter, and year. When data is rolled up into a summary table the size of the summary table shrinks vertically and horizontally with respect to the lower level table. Consequently, when an aggregate view may be used to answer a query the performance of the query increases considerably.

Aggregates must be chosen wisely because they consume disk space and often require maintenance, even recomputation, each time new data is loaded into the warehouse. The DBAs who are responsible for selecting the aggregates to be materialized must be aware of *sparcity failure* whereby the aggregate table is actually larger than the base table [12]. Two tools make aggregates easier to use and easier to manage. The first is the Red Brick Vista Advisor [5] which helps warehouse administrators manage aggregates by monitoring query usage patterns and recommending the most useful aggregates.

The second is the Aggregate Navigator [12] which allows users to write their queries against the base tables and the Aggregate Navigator rewrites the queries to use the best aggregate view. The Aggregate Navigator makes the warehouse more accessible because end users do not have to keep up with all the views. Aggregate views may be added, removed, or modified and the end user receives the full benefit of the aggregates without writing a single query to access a view. In [13], Kimball describes four design requirements for developing an aggregate environment capable of exploiting the full power of aggregate navigation.

3.3 ETL Process

The Extract, Transform, and Load (ETL) process consumes over seventy percent of the time and effort spent developing and maintaining a data warehouse. Consequently, it is important for students to appreciate the complexities of the process. Kimball decomposes the ETL process into 38 subsystems and he and Joe Caserta have written an entire book on the ETL process [17]. By studying the ETL process, which encompasses a subset of Universal Data Integration (UDI), students gain insight into the bigger issues of data and application integration.

The ETL process, which extracts data from the operational systems, transforms the data from the source systems into a consistent and coherent data set that is ready to present to the end user, and loads the data into the warehouse, is responsible for two critical properties of the warehouse: quality and traceability. When organizations are basing strategic business decisions on the information contained in the data warehouse, the quality of the data is of upmost importance. Olson's *Data Quality* book [24] provides extensive and thorough coverage of how to assess the quality of data and improve its accuracy.

The Audit dimension [16], which records where each piece of data comes from and exactly how it was transformed, is becoming increasingly important as organizations are forced to meet emerging compliance standards. Students must appreciate the importance of auditing the data and the ETL processes involved in migrating the data into the warehouse.

Students should be aware of the load window and reporting requirements that constrain the nightly ETL process, especially for global organizations that demand 24 x 7 access to the warehouse. It is also important to study techniques to improve load performance such as staging the load, disabling constraints during the load, and dropping and recreating indexes, since these techniques are largely irrelevant in an OLTP system.

Data warehouses are complex systems that cost many organizations miilions of dollars annually to build and maintain. This has created an entire industry that produces expensive tools to support the data warehouse. Consequently, every organization is forced to choose between building and buying the tools needed to implement the ETL process. Instructors must also face this dilema. Should the students use tools, like Oracle Warehouse Builder (OWB) [25], to implement an ETL process or do they benefit more from developing a hand-coded system? As discussed in section 5, there are pros and cons both ways. The primary benefit to using commercial tools is that all of the metadata from the ETL process is captured and managed. ETL tools also hide many details from the user such as which operations are being performed on data files and which are performed within the staging database. While this is helpful to practitioners, this layer of abstraction obscures the student's view and hinders their understanding.

This course has required students to use OWB to complete an ETL assignment. However, OWB is a complex application with a steep learning curve that has caused many problems for the students and the instructor. Consequently, the author plans to revise the assignment in favor of hand-coding a simple ETL process. Students who want to learn and use Oracle Warehouse Builder will be allowed to implement some aspect of their project with OWB.

3.4 Physical Schema

The physical schema of a data warehouse presents many challenging design decisions. The use of partitions is very important for rolling off data to conserve disk space. A partition is a logical structure used to segment data. A partition is defined by specifying the range of data that the partition will hold. Data files are associated with a partition such that all of the data within

the range of the partition is stored in the data files allocated to the partition. A partition may be dropped with a single DDL statement thereby deleting all of the data in the partition from the warehouse. Fact tables are often partitioned by date such that data may be rolled off after a certain period of time. An organization may want to store 13 months of data to enable yearly comparisons. To support this requirement the warehouse administrator might partition the fact table by months. A new monthly partition is created at the start of each month and at the end of the 13th month the first partition is dropped before the new partition is added thereby maintaining a 13-month window of data.

Some databases allow partitions to be loaded offline which allows users to access the online partitions while the offline partition is being loaded. Oracle allows partitions to be merged and split so a daily partition may be loaded offline and then merged with the partition for the current month.

As discussed in section 3.2, aggregate views play an important role in the warehouse, especially as the primary mechanism to improve query performance. However, aggregates typically consume a large percentage of the disk space allocated to the warehouse. Consequently, aggregates must be chosen wisely. Data striping, whereby aggregates are materialized at every other level of a hierarchy, allows the warehouse administrator to balance the tradeoff between computation and storage. If a query requests data from a materialized layer the data is returned with a single disk read, but if a query requests data from a layer that is not materialized the requested data is computed from the data at the next lower materialized layer. Consider a hierarchy along the time dimension consisting of day, week, month, quarter, and year. The most detailed data at the daily level should always be stored in a base table. The summary data may be striped by precomputing aggregates for month and year. If a query requests quarterly totals or averages the data is computed by summing or averaging the data in the monthly aggregate.

Although most commercial query optimizers will apply multiple indexes when performing a star join it is valuable to study multi-dimensional indexes. When considering high performance data warehouses that contain terabytes, or even petabytes of data, the physical schema can present some of the most interesting challenges.

3.5 OLAP

On-Line Analytical Processing (OALP) is synonymous with drilling up, down, and across to slice and dice data. The concept of slicing and dicing is typically presented in an introductory lecture that illustrates a data set as a three-dimensional cube. The students are quick to grasp the concept as the visual model is sliced and diced. OLAP tools and capabilities should be presented within the context of business intelligence. Several vendors have online demos that show how their OLAP tools may be used to analyze a data set. The author often uses the Desktop Intelligence demo from Business Objects [2] to motivate how data warehouses support Decision Support Systems (DSS) and other business intelligence applications. The Desktop Intelligence demo includes screen shots of an executive dashboard with key performance indicators (PKI).

This course has also used the open source Mondrian OLAP Server [22] to demonstrate an OLAP tool accessing one of the example models. The author is considering an OLAP lab and one or two assignments at the beginning of the course to help students better understand the Star schema.

Students should be clear on the tradeoffs between multi-dimensional OLAP (MOLAP), which stores data cubes as multi-dimensional arrays in main memory, versus Relational OLAP, which stores data cubes in a relational database, versus Hybrid OLAP (HOLAP), which stores data

cubes in main memory with the ability to drill through to the database if additional detail is needed.

3.6 Warehouse Lifecycle

The last lecture on data warehousing discusses the overall lifecycle of the warehouse from marshalling support to managing the project and selecting the applications to be delivered first. A fair amount of time is spent on the politics of the data warehouse to increase awareness and help the students understand the political issues along with the risks and dangers inherent in a data warehousing project. Greenfield [9] and Demarest [6] provide excellent discussions of the politics surrounding data warehousing and data warehouse projects. Practical guidelines are discussed such as starting small with a single data mart and building a business case based on return on investment. Kimball's *Data Warehouse Lifecycle Toolkit* [19] is the definitive guide to the data warehouse lifecycle

3.7 Assignments

The data warehousing portion of the course consists of five assignments plus a project described below. Each assignment focuses on a specific topic: dimensional modeling, aggregation, physical schema, ETL, and OLAP, but all of the assignments are based on the same application such that when combined the assignments constitute a complete data warehouse application. It is important for all of the students to be familiar with the application domain so they may focus exclusively on the learning objectives. An example application might analyze student performance based on a number of factors such as where the students live (dorm, apartment, home) and their study habits. The application could be motivated by suggesting that the Provost might use such a system to help students who are struggling and to proactively identify students at risk. It is helpful to describe the operational system that will be used to collect the data. A student performance application might provide a simple interface that allows each student to record how they spend their time in 15-minute increments (e.g., attending class, eating, sleeping, studying).

The assignments require each student to write a two to three page essay describing their designs (e.g., design a Star schema with one fact table and at least three dimensions other than Date and Time of Day). Because the course has been designed in part to attract IS students, hardcore programming is not required. However, the students must specify their designs at a sufficient level of detail to fully understand the concepts. Therefore, the assignments often require the students to include SQL and snippets of pseudo code in their essays.

Although we all want to encourage creativity and we would like our students to solve open ended design problems, the design assignments must be structured around a very specific set of questions to guide the students through the relevant details. Using a standard set of questions across the lectures, assignments, and exams helps to set expectations and increase retention.

3.7.1 Using the Warehouse

To evaluate the student's ability to utilize and employ a data warehouse in a problem solving situation, the OLAP assignment presents the students with a data warehouse and a set of questions to be answered based on the contents of the warehouse. For example, given a retail warehouse with sales data collected over a period of time (e.g., a week or month) the students might be asked to explain why total revenue fell below expectations. To complete the assignment

students must drill into the details to discover that the sales from a particular region or city were low which explains why actual revenue fell short. This assignment may be done with SQL or with an OLAP query tool. The assignment may be made more challenging by adding a second fact table that records the sales forecast. Students must drill across the two fact tables to compare actual revenue with forecasted revenue.

3.7.2 ETL Assignment

Slazinski [26] notes how the data warehousing course at Purdue generates source materials for other courses. We have taken the opposite approach. Students in the first database course typically develop a retail application, such as a shopping cart for an online store, complete with customers, products, and orders. The Database I course devotes one or two lectures to introducing data warehousing. The author has developed a PL/SQL package that extracts all of the orders, customers, and products from the student projects and loads the data into a data warehouse along with a dimension that records each of the student's stores. Each student executes the ETL package to populate a single data warehouse. In a previous assignment the students design a simple set of reports based on their OLTP schema. Once the data warehouse has been populated the students rewrite their reports for the data warehouse to experience the difference between a transactional data model and a dimensional model. The introductory students benefit from a deeper experience with data warehousing and the data warehousing students benefit from having a fully populated warehouse to study and explore. In addition, the PL/SQL package gives the students an example of a simplified ETL process.

3.7.3 Class Project

At the beginning of the semester the students are asked to pick a domain that interests them. Many students pick sports, especially college basketball and professional football because they want to predict the winners in the NCAA Men's basketball tournament or they want to improve their NFL fantasy football team. Other students pick domains, such as insurance or printing, that are related to their current employment. The students do not need to know a lot of details about the domain, but the domain must interest them enough that they are willing to go deep as they develop their project.

After each of the five homework assignments has been completed and discussed in the class the students must complete a similar project assignment for their chosen domain. The project requires the students to define a normalized schema for their source system, design a Star schema with at least two fact tables, define two aggregate views, define the physical schema, and define the ETL process between their operational system and the warehouse. Like the homework assignments described above, the students write a two to three page essay for each project assignment. However, the project may require some implementation or award extra credit for implementing some aspect of the system.

The final project requires a single document that pulls everything together. The final paper is weighted higher than the other project assignments and students are encouraged to apply everything they have learned to improve the sections from their earlier assignments. The final paper makes a nice addition to the student's portfolio and in some cases the students present their paper to their employers as a proposal.

Because the students choose domains that interest them they often get caught up in their domain and lose sight of the project objectives. Consequently, it is important to break the project into parts and provide feedback throughout the course to keep the students on track.

3.8 Exams

The course originally included two exams during the semester plus a final exam. The first exam covered dimensional modeling, aggregation, and the physical schema. The second exam included topics from the first exam but focused on ETL, OLAP, and Reporting. The final exam was cumulative but focused on data mining. Although data warehousing and data mining complement each other, it seems more logical to break the class into two distinct parts with equally weighted exams following each part. Eliminating the second exam has the additional benefit of saving a week of instruction that would otherwise be spent reviewing for the exam, taking the exam, and reviewing exam solutions. The next iteration of the course will be taught with two exams, a mid-term and a final exam covering: (1) Data Warehousing and Reporting, and (2) Data Mining.

Fifty percent of the data warehousing exam is based on a design problem while the other half of the exam covers conceptual questions. The design problem describes an application and provides the students with a sketch of a Star schema. The design problem draws from the standard questions that have been used throughout the lectures and homework assignments to assess how well the students are able to apply what they have learned. The students are asked to fill in details of the schema, such as identifying the measures in the fact table and identifying and justifying the best fact table grain for the application. The students may also be asked to design an aggregate schema that satisfies a particular type of analysis. Given a basic description of an operational system the students might discuss some aspects of the ETL system. Questions like, "What additional views are needed to complete this schema and why?" allow the instructor to cover advanced topics, like the implementation of dimensions that play multiple roles, without being too obvious. However, the design questions must be specific and they should come from a standard set of questions that are familiar to the students.

One of the design problems was based on an article from Inc. Magazine describing how Zipcar [27], the world's largest car sharing service, increased annual revenue from 2 million to 100 million dollars [4]. The students were given a copy of the article a week prior to the exam. This particular article provided enough detail about various decisions to present a simple data warehouse and ask the students to describe how the data may have been mined and analyzed. The students gained confidence knowing they could take an article about a real company and identify what data was captured and what analysis was done to significantly improve the business. Basing design problems on profiles of real companies ties the course back to the business value of data and reminds the students how critical a data warehouse can be to the success of an organization.

4 Student Feedback

In an effort to continually improve the course the students are given an extra credit assignment in which they may earn additional points based on the value of their feedback. The assignment is structured around several specific questions along with general comments about the course.

The biggest challenges reported by the students were disassociating from OLTP systems, understanding the dimensional model, designing a Star schema from scratch, and becoming comfortable with de-normalization. On the surface a data warehouse looks like any other database and at the beginning of the course the students experienced a good deal of interference from the transactional systems they had studied previously. Students also had difficulty defining boundaries around dimensions because all of the data seemed related. Slazinski found the concept of

de-normalization to be the "single biggest challenge facing the students" [26]. The struggle to overcome the urge to normalize cannot be underestimated.

All of the students agreed that constructing models in class was the most helpful aspect of the course. Overall the students want more examples. Several students requested examples of data transformations to illustrate the differences between operations that should be performed on data files versus operations best performed in a relational database. One student even suggested exporting example data from a legacy system. Most students agreed that the standard questions, along with the examples discussed in class, helped them recognize patterns.

Some of the students did not care for Kimball's articles. They felt the articles were difficult to follow, the details were often sketchy, and some of the articles spent too much time promoting the Kimball Group. One student even claimed that the articles put too much pressure on him because he was driven to understand every last detail. This student felt that by trying so hard to understand the details he missed the big picture. He also reported that the class discussions helped him understand the articles and he often got more out of the articles when he reread them after class. It is important to clarify expectations around the articles and to manage the reading load carefully. Even when the expectations are well deinfed the better students will set high standards for themselves and read more than the assigned material. It is helpful to provide these students with a list of recommended articles in addition to the required reading.

Two articles emerged as clear favorites: Fundamental Grains [15] and the discussion on a "family of schemas" in Aggregate Navigation With (Almost) No Metadata [13]. Many students felt the articles would be a great resource in the future.

Several students mentioned the need to refresh general database techniques. The first project assignment is helpful in this respect as the students must describe the operational system(s) that will feed their data warehouse. Most students felt the project was essential to mastering the material.

A few students commented on the ETL process and OLAP. One student suggested that looking at system tables may help students better understand metadata. With respect to OLAP the students want less lecture and more hands-on experience. The next iteration of the course will include an OLAP lab.

Although most students felt the dimensional model was the most interesting aspect of the course, a few students who gravitated more towards administering a database than developing applications reported that the physical schema was the most interesting part of class and they felt they needed to work through the low level details before they fully understood the concepts.

5 Recommendations

When presenting design solutions questions are extremely important to frame and structure the material and to improve retention. Each modeling topic is introduced with a standard set of questions. For example, there are three fundamental fact table grains: transaction, periodic snapshot, and accumulating snapshot. When reviewing a particular model the questions about the grain of the fact table include:

- What is the grain of the fact table? Why?
- Why is this grain appropriate for this application?

The questions are used to evaluate the models discussed in class and the same questions are used on homework assignments and exams. The students felt these questions helped them to better understand the models and retain the key concepts.

The students unanimously felt the examples were the most helpful material covered in class. Planning out 3 - 5 examples and referring back to the examples throughout the semester increased retention because the students were able to recall how a concept was applied in an example. Database courses typically require substantial effort to develop course materials and data warehousing courses are no different.

Oracle Warehouse Builder offers many advantages to students. First, given the high price tag, this may be a student's only opportunity to use the tool. Second, the students gain valuable experience working with a graphical programming interface. Third, the complexity of the tool helps students appreciate the complexity and scope of the ETL process and data warehouses in general. Finally, developing some proficiency with OWB will most certainly increase the student's marketability.

However, OWB has considerable drawbacks. First, the learning curve is steep. Second, the tool is overkill for a class project. Third, OWB hides many details of the ETL process. Overall, students probably learn more from a hand-coded project. However, some students may want to use OWB as part of their project. If the students are going to use OWB they should start early and allow sufficient time to overcome the learning curve. OWB can be used throughout the course to explore existing warehouses, develop a new warehouse, and develop an ETL process. Most of the online help provided by the tool and the information available on the Internet focuses on the functionality provided by the tool. The instructor and the students have found few good examples of how to perform basic tasks (e.g., create an aggregate summary table). Consequently, labs and tutorials that that offer a cookbook of solutions would be especially helpful for those students who wish to tackle OWB.

Although the class was specifically designed to avoid hardcore programming, it is important for the students to specify their designs in SQL and pseudo code. This is especially true when defining aggregate summary tables and developing an ETL process. The devil is in the details. Even if the students do not produce correct queries, just the experience of thinking the problem through to that level is beneficial.

Many students expect to join the dimension tables to each other; a reasonable expectation given the students background and experience with transactional systems. Therefore, explicitly showing an ER diagram to illustrate that the dimension tables only join to the fact table and walking through query execution to illustrate how the dimensions act as indexes into the fact tables reinforces the relationships between the tables and helps students to visualize how the schema supports drilling up, down, and across.

Finally, although some of the students did not care for Kimball's articles, the articles provide a practical side of data warehousing that is not typically found in a text book. Providing discussion questions with the articles helps to set expectations and focus the students on the salient details.

6 Future Directions

The author is considering the following changes to the course: (1) reposition OLAP at the beginning of the class, and (2) drop Oracle Warehouse Builder.

OLAP is currently covered at the end of the data warehouse portion of the class because it is logical to discuss data warehouse applications after the students understand how to build the warehouse. In addition, OLAP provides a nice segue to reporting. In its current placement within the course the students are quick to pick up the concepts and the discussion quickly

shifts to the limitations of SQL to motivate the need for reporting writers and query tools. Given the difficulty that students have understanding the dimensional model the author is going to introduce OLAP early in the next iteration of the class along with one or two assignments that require the students to use the Mondrian OLAP tool to explore some of the example data warehouses. This should help the students understand the dimensional model and how it is used. This will also introduce the concept of business intelligence early in the course which should help to motivate the need for data warehouses and reinforce the main themes of the course.

Students have struggled with Oracle Warehouse Builder in other classes and our experience in the data warehouse class has been no different. OWB is a complex tool with a steep learning curve and it hides many relevant details that students need to see and experience. Therefore, the course will no longer require an assignment involving OWB. The ETL assignment will be revised to design, and possibly implement, a hand-coded solution. Students who are interested in OWB will be allowed to implement a portion of their class project with OWB.

7 Conclusion

The Internet enabled one-to-one marketing which drove the desire to capture every aspect of customer behavior. It is not enough to know what items are in a customer's shopping cart; marketing demands to know when items are placed in the cart and when they are removed to better understand the factors that influence customer behavior. New advances in electronic commerce and other online services will continue to drive the demand for data, information, and knowledge. Databases will continue to grow in importance as our demands for information increase and the pressure to expand the database curriculum will continue to mount as new database technologies emerge to meet the demands.

Data warehouses form the foundation for mission critical applications that guide strategic business decisions. Consequently, universities must offer more courses to prepare students for exciting career opportunities developing, maintaining, and consuming data warehouses. This paper has described an undergraduate data warehouse course at Radford University to continue the process of defining requirements for data warehousing courses and to provide a model that will hopefully encourage others to develop similar courses.

Acknowledgements

The author wishes to thank Bob Phillips for his help and support in developing this course and for his comments on earlier drafts of this paper.

References

[1] Christopher Adamson. *Mastering Data Warehouse Aggregates: Solutions for Star Schema Performance*. John Wiley & Sons, July 2006.

[2] Business Objects desktop intelligence demo. `http://www.businessobjects.com/product/catalog/desktop\_intelligence/`.

[3] Craig J. Bunger, Latha S. Colby, Richard L. Cole, William J. McKenna, Gopal Mulagund, and David Wilhite. Aggregate maintenance for data warehousing in informix red brick vista. *Proceedings of the 27th International Conference on Very Large Data Bases*, pages 659 – 662, September 2001.

[4] Stephanie Clifford. How fast can this thing go, anyway? *Inc.*, March 2008.

[5] Latha S. Colby, Richard L. Cole, Edward Haslam, Nasi Jazayeri, Galt Johnson, William J. McKenna, Lee Schumacher, and David Wilhite. Red brick vista: Aggregate computation and management. *Proceedings of the Fourteenth International Conference on Data Engineering*, pages 174 – 177, February 1998.

[6] Marc Demarest. The politics of data warehousing. `http://www.noumenal.com/marc/dwpoly.html`, 1997.

[7] Suzanne W. Dietrich, Susan D. Urban, and Susan Haag. Developing advanced courses for undergraduates: A case study in databases. *IEEE Transactions on Education*, 51:138 – 144, February 2008.

[8] Roger Fang and Sama Tuladhar. Teaching data warehousing and data mining in a graduate program in information technology. *Journal of Computing Sciences in Colleges*, 21:137 – 144, May 2006.

[9] Larry Greenfield. Data warehousing political issues. `http://www.dwinfocenter.org/politics.html`.

[10] Bill Inmon. Corporate information factory. `http://www.inmoncif.com/home/`.

[11] How would you characterize the difference between Bill Inmon's philosophy on data warehousing and Richard Kimball's?

[12] Ralph Kimball. The aggregate navigator. *DBMS*, November 1995.

[13] Ralph Kimball. Aggregate navigation with (almost) no metadata. *DBMS*, August 1996.

[14] Ralph Kimball. A dimensional manifesto. *DBMS*, August 1997.

[15] Ralph Kimball. Fundamental grains. *Intelligent Enterprise*, 2, March 1999.

[16] Ralph Kimball. Design tip #26: Audit dimensions to track lineage and confidence. `http://www.kimballgroup.com/html/designtips.html`, August 2001.

[17] Ralph Kimball and Joe Caserta. *The Data Warehouse ETL Toolkit*. John Wiley & Sons, September 2004.

[18] Ralph Kimball and Margy Ross. *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling*. John Wiley & Sons, second edition, April 2002.

[19] Ralph Kimball, Margy Ross, Warren Thornthwaite, Joy Mundy, and Bob Becker. *The Data Warehouse Lifecycle Toolkit: Practical Techniques for Building Data Warehouse and Business Intelligence Systems*. John Wiley & Sons, second edition, January 2008.

[20] Kimball group. `http://www.kimballgroup.com/`.

[21] Ron McFadyen. Data warehousing in an undergraduate curriculum. *Journal of Computing Sciences in Colleges*, 22:221 – 227, April 2007.

[22] Pentaho analysis services: Mondrian project. `http://mondrian.pentaho.org/`.

[23] Meg Murray and Mario Guimaraes. Expanding the database curriculum. *Journal of Computing Sciences in Colleges*, 23:69 – 75, January 2008.

[24] Jack E. Olson. *Data Quality: The Accuracy Dimension*. Morgan Kaufman, December 2002.

[25] 11g Oracle Warehouse Builder. `http://www.oracle.com/technology/products/warehouse/index.html`.

[26] Erick D. Salzinski. Teaching data warehousing to undergraduates: tales from the warehouse floor. *Proceedings of the 4th conference on Information technology curriculum*, pages 242 – 248, October 2003.

[27] Zipcar. `http://www.zipcar.com/`.