

Utah State University

DigitalCommons@USU

---

Undergraduate Honors Capstone Projects

Honors Program

---

12-1-2011

## Business Intelligence Tools in a Developmental Environment: An Academic Module

Landon M. Hemsley  
*Utah State University*

Follow this and additional works at: <https://digitalcommons.usu.edu/honors>



Part of the [Management Information Systems Commons](#)

---

### Recommended Citation

Hemsley, Landon M., "Business Intelligence Tools in a Developmental Environment: An Academic Module" (2011). *Undergraduate Honors Capstone Projects*. 113.

<https://digitalcommons.usu.edu/honors/113>

This Thesis is brought to you for free and open access by the Honors Program at DigitalCommons@USU. It has been accepted for inclusion in Undergraduate Honors Capstone Projects by an authorized administrator of DigitalCommons@USU. For more information, please contact [digitalcommons@usu.edu](mailto:digitalcommons@usu.edu).



**BUSINESS INTELLIGENCE TOOLS IN A DEVELOPMENTAL  
ENVIRONMENT: AN ACADEMIC MODULE**

by

**Landon M. Hemsley**

**Thesis submitted in partial fulfillment  
of the requirements for the degree**

of

**DEPARTMENTAL HONORS**

in

**Management Information Systems  
in the Department of Management Information Systems**

**Approved:**

---

**Thesis/Project Advisor**  
Dr. David Olsen

---

**Departmental Honors Advisor**  
Dr. Jeffrey T. Doyle

---

**Director of Honors Program**  
Dr. Christie Fox

**UTAH STATE UNIVERSITY**  
**Logan, UT**

**Fall 2011**

**TABLE OF CONTENTS**

<b>Abstract</b>	<b>3</b>
<b>Introduction</b>	<b>3</b>
<b>Literature Review</b>	<b>4</b>
<b>Value of Business Intelligence to Industry</b>	<b>4</b>
<b>Methods of Business Intelligence: Data Warehousing</b>	<b>8</b>
<b>Methods of Business Intelligence: Clustering and MapReduce</b>	<b>11</b>
<b>Importance of BI in Education</b>	<b>12</b>
<b>Classroom Module: Using Reporting Services for BI</b>	<b>14</b>
<b>Tutorial: Reporting Services</b>	<b>16</b>
<b>Review</b>	<b>22</b>
<b>Conclusion</b>	<b>22</b>
<b>Bibliography</b>	<b>24</b>
<b>Appendix A – Relational Database Theory</b>	<b>26</b>
<b>Author’s Biography</b>	<b>30</b>

## **I. ABSTRACT**

Business Intelligence (BI), defined here as the creation and use of meaningful business information for management, has over the last 20 years shown over and over again its value to professional industry. The prevalence of BI has also demonstrated that a new brand of worker is needed to take advantage of the sophisticated tools available to modern industry. Evidence suggests modern universities have struggled to educate professionals on the tremendous impact BI can have on an organization if used properly. This work is a step in an endeavor to bridge that educational gap and generate an educational strategic advantage by providing a module to be used in current information systems courses.

## **II. INTRODUCTION**

BI is the advantageous implementation of “decision support technologies for the enterprise aimed at enabling knowledge workers such as executives, managers, and analysts to make better and faster decisions” (Davenport, 2006). Different organizations will define success differently, but BI success can generally be measured by the effectiveness of data analytics in facilitating decision-making.

BI tools can facilitate the recognition of company or industrial trends from patterns in sales, manufacturing, or other operations and make decisions based on those trends for the overall advantage of the enterprise. These tools use computational power and analytical algorithms to convert thousands or millions of records into meaningful information to assist leadership.

Anecdotal evidence suggests BI tools have only seen wide use in business during the last two decades, and as such, their value to organizations has not yet been fully realized. As these tools are continually refined, and as computational machinery continues to be developed, the potential value of these tools to business organizations will only continue to increase.

This thesis reviews the value of BI in both business and an educational environment and outlines a method by which students can be taught to use some of the BI tools offered in the Microsoft SQL Server Development Environment (2008 R2) to analyze stock market data gathered from public sources. The theme of this thesis is centered on the added value an education in BI can give students, which in turn creates value to industry and the world at large.

### **III. LITERATURE REVIEW**

#### **Value of Business Intelligence to Industry**

Chaudhuri, Dayal, and Narasayya summarize the value of the BI industry over the last two decades. They suggest that BI has enjoyed explosive growth during this time. BI applications allow businesses to analyze business metrics, for example, to “report total sales by region for this year and compare with sales from last year.” These tools allow for common BI operations, such as “filtering, aggregation, drill-down and pivoting” among others (Chaudhuri, Dayal, & Narasayya, 2011). Other more advanced tools bring more in-depth information. For example, data mining engines are designed specifically to provide analysis “that goes well beyond what is offered by OLAP or reporting servers, and provides the ability to build predictive models to a greater extent than the common BI operations would allow” (Chaudhuri, Dayal, & Narasayya, 2011).

Use of BI can be directly correlated with increased profits. Continental Airlines realized a return of investment of more than 1000 percent, investing roughly \$30 million into its BI infrastructure and creating additional revenue streams of more than \$500 million (Anderson-Lehman, Watson, Wixom, & Hoffer, 2004). Lonnqvist and Pirrtimaki put the average return on resources for all BI projects at 310 percent (2006).

Despite the vast potential for increased profitability, "...upper management—especially at larger firms—is beginning to realize that information is worthless if it isn't used to maximize profits" (Williams, 2011). The world annually outlays \$60 billion in BI development, a sizeable investment. "Applying BI takes resources, and the benefits actually occurring in practice are not always clear" (Lonnqvist & Pirrtimaki, 2006). Research performed at the University of Cambridge in the UK suggests many industries may not have adequately prepared themselves for the acquisition and use of BI. "More than 50 percent of BI projects fail to deliver the expected benefit," and "seven of 10 executives say they do not get the right information to make business decisions" (Giniat, 2011).

Such executives who do not get the right information to make good decisions are unable to realize the full potential of BI in their business. The creation of new value, either by generating new revenue or by decreasing costs, is a main purpose of BI (Williams, 2011), but what that value is and how it's generated will vary. For a CFO, BI success might mean having the ability to more closely analyze how operational performance affects the bottom line, but for a CMO, BI success might mean "better customer segmentation, more precise campaign targeting, (and) improved customer service and customer retention" (Williams, 2011).

By creating new value, companies set themselves apart from competition and substitutes, creating competitive advantage. Davenport's work illustrates that when companies can no

longer effectively differentiate their products, they create competitive advantage by improving business processes; Capital One, Amazon, and the Oakland Athletics attest to that (2006).

The effects of competitive advantage translate directly to the bottom line. The degree to which BI directly translates to competitive advantage, however, will depend on the type of business, purpose of the business, and interests of the stakeholders. For some, BI can be used to offer a large competitive advantage, but for others, BI can only offer industry parity (Williams, 2011).

As an example of this idea, let us return to Continental Airlines. Brenneman outlines how this company saved itself through reconstruction and use of BI (1998). Continental designed, built, and deployed a massive data warehouse as part of a “First to Favorite” initiative. Because of this initiative, Continental has achieved near real-time BI. Continental uses its warehouse to quickly identify their most valuable customers, identify high passenger-volume trends on certain routes and can adjust flights to quickly accommodate a heightened passenger load, etc. Interactive dashboards allow company leaders to “quickly identify issues in the Continental flight network and then manage flights in ways to improve customer satisfaction and airline profitability.”

Besides the airline industry, the effects of BI have been heavily in health care. Recently implemented federal regulations to doctors in the United States are meant “to create a nationwide care delivery system driven by data-supported best practices that deliver the highest quality care for the lowest possible costs” (Ericson, 2010). Current benefits of BI tool use in health care appear to be limited to the rate of industry acceptance and culture conversion (Ericson, 2010).

Studies on the effectiveness of BI tools in health care industry reveal an increase in efficiency of health care business practices and a corresponding increase in doctor and patient

satisfaction. Tools used in a clinic in the UK allowed the clinic to meet regulation compliance requirements, address key quality control indicators, easily distribute information across the hospital, make data available to hospital consultants, and save time through streamlining data management (Case Study II: South Tyneside Healthcare NHS Trust/Cognos, 2004). Black, et.al, demonstrate that BI has affected private and public sectors many ways, all of them positive (Black, Snow, Olsen, & Cooney, 2010). “For companies working in highly competitive and information-intensive industries, BI is critical for understanding business operations, increasing customer satisfaction, and taking advantage of new opportunities” (Lukmana, Hackney, Popovič, Jaklič, & Irani, 2011).

The effects of BI are not limited to western companies only. BI has facilitated economic growth in the developing world in a variety of industries. Lukmana, et al., examine the impact BI software has had on the emerging economy of Slovenia. While this small European nation cannot be considered representative of all developing economies, it does provide an example of how BI can influence the developing world.

Slovenia faces cultural and operational barriers to the adoption of BI technology. Slovenia is “mainly focused on the development of information systems for operative business processes,” lack(s) adequate knowledge, resources, a culture of change, and does not adopt state-of-the-art technologies at an adequate pace (Lukmana, Hackney, Popovič, Jaklič, & Irani, 2011).

“For companies working in highly competitive and information-intensive industries, BI is critical for understanding business operations, increasing customer satisfaction, and taking advantage of new opportunities” (Lukmana, Hackney, Popovič, Jaklič, & Irani, 2011). BI enjoys widespread use in Slovenia’s manufacturing, financial, and service industries, both because of its value to business and the requirements of European financial regulations (Lukmana, Hackney,



Popovič, Jaklič, & Irani, 2011). However, Slovenia “lags slightly behind the international community in BI adoption” largely because of a lagging economy and the purchasing power of Slovenia as a nation (Lukmana, Hackney, Popovič, Jaklič, & Irani, 2011).

BI brings new power of choice to those in the business of decision making. Successful implementation of BI architecture will give decision-makers important information they can use to increase the value of their business. This affects individual communities, individual companies, and individual nations to a different degree, but its effect can be observed at all levels of business. BI has been and will more greatly be instrumental in the evolution of the business environment.

### **Methods of Business Intelligence: Data Warehousing**

Multiple methods have been developed to leverage the decision-making power BI has to offer. The methods by which information has been gleaned from data have varied tremendously. One such method is data warehousing, which in and of itself has become a staple of business intelligence and a focus of the teaching module in this thesis. The other, clustering, makes use of the computing power of several small systems to perform the calculations a super computer or massive data computing center could accomplish.

A data warehouse is “a subject oriented, integrated, non-volatile, and time variant collection of data in support of management’s decisions” (Inmon, 1996, p. 33). Data warehouses operate on the premise that more data, more complete data, translated into meaningful information quickly and given to management will result in wiser decisions being made more quickly. The standard operational databases of business are the foundation from which data warehouses are derived.

Data warehouses are different than operational databases, and are often kept separate (Chaudhuri & Dayal, 1997). Operational databases exist to facilitate business tasks, tasks that “are structured and repetitive, and consist of short, atomic, isolated transactions. The transactions require detailed, up-to-date data, and read or update a few (tens of) records accessed typically on their primary keys” (Chaudhuri & Dayal, 1997). These types of databases are designed with maximization of speed for transaction execution in mind (insert, update, delete) and are not designed for maximization of query efficiency.

The requirements of a data warehouse are different. Warehouses are designed for online analytical processing (OLAP), which require an alteration in data structure and normalization conventions in order to house historical data for extended periods of time. Warehouses will also be much larger than operational databases. “Since data warehouses contain consolidated data, perhaps from several operational databases, over potentially long periods of time, they tend to be orders of magnitude larger than operational databases; enterprise data warehouses are projected to be hundreds of gigabytes to terabytes in size... The workloads are query intensive ... Query throughput and response times are more important than transaction throughput” (Chaudhuri & Dayal, 1997).

Because requirements for data warehouses are different than for operational relational databases, the structure of warehouses differs. Ralph Kimball, a leader in data warehousing, engineered a process for structuring data warehouses known as *Dimensional Modeling*.

“...dimensional modeling is the most viable technique to deliver data for business intelligence because it addresses the twin non-negotiable goals of business understandability and fast query performance” (Kimball, Ross, Thornthwaite, Mundy, & Becker, 2008, p. 233).

Dimensional modeling is centered on facts and dimensions. Facts are records of transactions. Dimensions are the measures by which a business can make sense of the business environment. The fact table keeps a record of the raw transactional data without context, thus normalized to the third normal form (3NF) (Kimball, Ross, Becker, Mundy, & Thornwaite, 2010, p. 181). Many of the records in the fact table will be foreign keys to several dimensional tables, keys through which BI applications will be able to derive context or an analytical frame of reference.

Because dimensional tables provide context to the heart of this model, the fact table, it is very important that any query that runs against

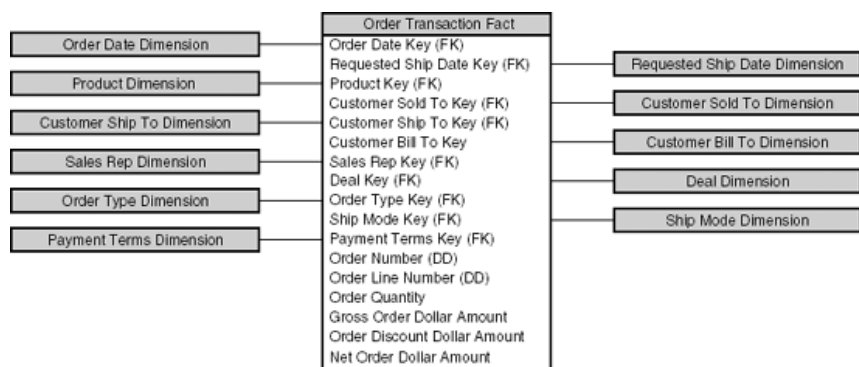


Figure 1 – A sample dimensional model. (Kimball, Ross, Thornthwaite, Mundy, & Becker, 2008, p. 235)

them not need to perform needless joins. Figure 1 portrays some dimensions that could exist in a dimensional model. If this dimensional information began to occupy several different tables and spread into what Kimball calls a “snowflake,” multi-table joins would become necessary to add the desired context, and those multi-table joins defeat the purpose of the system. For the dimensional tables, *denormalization* is the way to go (Kimball, Ross, Becker, Mundy, & Thornwaite, 2010, p. 135). Data in these tables should be normalized to the second normal form. This allows the system to get necessary context while optimizing processing capability (Kimball, Ross, Becker, Mundy, & Thornwaite, 2010, p. 181).

## **Methods of Business Intelligence: Clustering and MapReduce**

Data warehousing offers powerful data processing ability and methods, but it is not the only tool that can be used to process and analyze inordinate amounts of data, and may not be the most powerful (Dean & Ghemawat, 2010). Researchers at Google developed a different method for storing and analyzing such data, a method that does not use tables and does not utilize SQL for data query and analysis and distributes storage and processing power over many machines. This method, known as MapReduce, utilizes “large clusters of commodity PCs connected together with switched Gigabit Ethernet” to process the large amounts of data used in BI (Dean & Ghemawat, MapReduce: Simplified Data Processing on Large Clusters, 2008).

In some cases, a distributive computing strategy such as MapReduce may be more efficient and powerful than a dimensional modeling strategy. Because data warehousing solutions require an extraction-transformation-load cycle to construct a dimensionally-modeled data structure, data warehousing may be useless for certain applications. Dean and Ghemawat explain that in Google’s case, data warehousing is “inconvenient and unacceptably slow” for their web-crawling-and-indexing system in their search algorithms. Data warehouse tables would be used only once and then disposed of (2010). Additionally, Dean and Ghemawat explain that SQL simply does not suit the complexity of certain mapping functions, such as those present in Google Maps and Google Earth (2010).

As Ranger, et al., also point out, MapReduce also holds an advantage over multi-thread processing. It allows programmers to take advantage of multi-core, distributed processing systems without needing to manage for parallel programming. Implementations of MapReduce can lead to “scalable performance for both multi-core chips and conventional symmetric multiprocessors” (Ranger, Raghuraman, Penmetta, Bradski, & Kozyrakis, 2007)

Constructing a distributed computing system and implementing a MapReduce program in an educational setting may not be practical. The implementation of MapReduce programs requires several dedicated computers and very, very large datasets. As such, this work focuses on the BI advantages that can be gained through data warehousing and use of reporting and analysis services in the Microsoft Business Intelligence Development Studio. The use of open-source MapReduce programs, such as Hadoop, may in the future become critical to BI education in institutions of higher education.

Dimensional modeling, data warehousing, and distributed computing serve to create a system through which transactional data can be queried, processed, and analyzed quickly. The value of such systems for business intelligence is that its schema is designed to maximize processing ability on extraordinarily large sets of data, often to the millions of records. The end result from that processing is information that allows business leaders to make better decisions.

### **Importance of BI in Higher Education**

“... The ability for the average user to extract relevant data without the aid of specialized applications is limited,” despite the interactivity of BI databases. BI tools, while commercially available, can be very difficult to understand (Elbashir, Collier, & Sutton, 2011). If the potential operator of any tool fails to recognize its value or simply doesn't know how to use it, the tool is useless. This applies for BI tools as well. As previously described, BI tools make use of sophisticated computational systems to translate very big data into meaningful information, but if no educated worker is there to interpret that information or derive meaning from it, then the BI tool doesn't fulfill its mission very well.

Elbashir, Collier, and Sutton postulate that if top-level management and operation-level management have a high level of “absorptive capacity” (i.e. the inclination to understand and utilize BI tools), those tools will be far more likely to be assimilated into the organization. They also conclude that the IT infrastructure sophistication “enhances BI assimilation. ... This supports the belief that underlying enterprise systems-related IT infrastructure plays an important role in BI assimilation” (Elbashir, Collier, & Sutton, 2011).

Thus, education in BI theory and tool application is critical in modern industry. As those with knowledge of the usefulness of BI tools enter the workforce, the value of those tools increases, as does the likelihood that the tools will be used. It is the responsibility of institutions of higher education to equip top-level and operation-level managers with the knowledge and expertise necessary to adopt useful BI tools and to use them for the good of industry.

Some institutions of higher education have attempted to implement some form of BI education, but very few. As a notable example, Purdue University implemented a one semester course on data warehousing, a fundamental in BI (Slazinski, 2003). Students learn dimensional modeling, data mining, Extraction-Transformation-Load (ETL), etc. They design and implement a data warehousing solution, which they leave at the university upon graduation. Purdue’s implementation has received positive feedback from students, and “90 percent of students who take these electives plan on getting a job in ... database” (Slazinski, 2003).

The University of Arizona is another notable example of excellence in BI education. They offer two courses in BI, one focusing on the impact BI has had on the social media universe, and another for graduate students in which students learn dimensional modeling, ETL, data staging, and many other important BI techniques. Beyond construction of a BI data warehouse and working with big data, students at Arizona, “learn how to gather strategic

decision making requirements from businesses, develop key performance indicators (KPIs) and corporate performance management metrics using the Balanced Scorecard, and design and implement business dashboards” (University of Arizona MIS Department, 2011).

Purdue and Arizona are the exception. Innovation in business curricula has “failed to recognize the urgent need to introduce courses and specializations in the emerging field of BI” (Sirkar, 2009). Of the Top 50 business schools in America as set forth by LaVelle (2007), 12 include a BI course in their offering. None offer a BI major course of study, and only one offers a minor course of study (Sirkar, 2009).

#### **IV. CLASSROOM MODULE: USING REPORTING SERVICES FOR BI**

The classroom module should take a real-life application for BI tools and allow students to gain experience with them. In this section, we explore a basic tool in the Microsoft BI development toolset: the dynamic report.

The report can be used to review data both over a period of time in the past and in real-time. It stands to reason that the quicker decision-makers get information, the quicker they can take action that will enable them to gain an advantage against the competition.

The question in this report is to determine whether or not a given stock should be bought or sold. That decision is made using information derived from a report generated in Microsoft’s Business Intelligence Development Studio reporting services. Daily closing stock values are compared to a multiple day moving average price value of that stock. Moving averages of stock prices are used as indicators for the purchase or sale of stocks on the market (Simple and Exponential - Moving Averages).

Purchase and sale signals are generated by market movement. The closing price of the stock must be compared to yesterday's closing price and the moving average. A buy signal is generated when the difference between the closing price and moving average is positive and when the previous day's difference is negative. Contrariwise, a sell signal is generated when the difference between the closing price and the moving average is negative and when the previous day's price is positive. Ideally, this data should be measured at the present.

Brock, et al., contains a full theoretical discussion to the merits of using moving averages for stock purchasing and sale decisions; such will not be debated here (1992). Furthermore, the length of the moving average is less important than the application of the moving average itself. In this application, a 35-day moving average is used, but any moving average of reasonable length could take its place. While the data used may not be readily available and easily accessible to all, any comparable data to that which we use here is acceptable for the purpose of the lesson.

To begin, the student should have access to a database constructed in a Microsoft SQL Server. Direct Access to the database through the SQL Server Management Studio is not important. We will not be modifying the data; we will simply be analyzing it. Access to the Business Intelligence Development Studio, part of Microsoft Visual Studio, is necessary.

The tutorial covering the next few pages will allow interested BI developers to generate a report to assist decision-makers decide whether to buy or to sell a given stock.

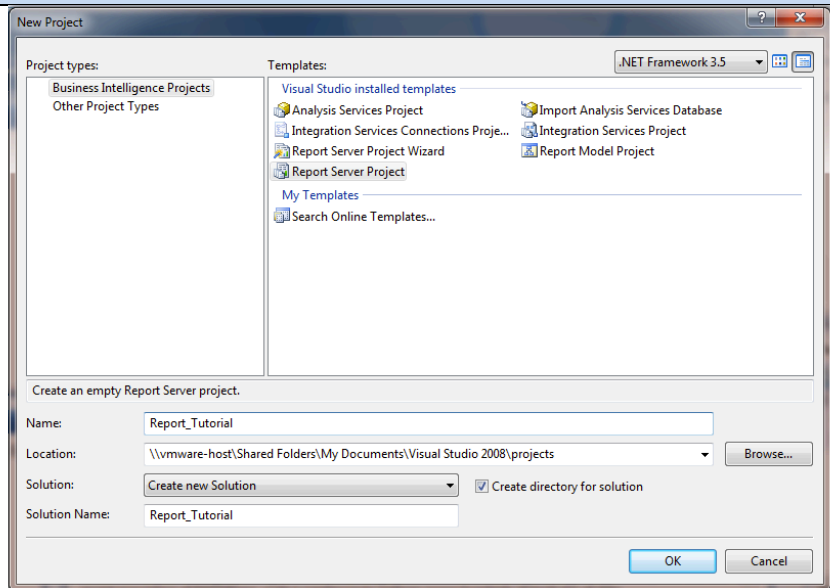


# Tutorial: Reporting Services

## Create a New Project

To begin, we must access the report designer portion of the Microsoft Business Intelligence Studio (BIDS program).

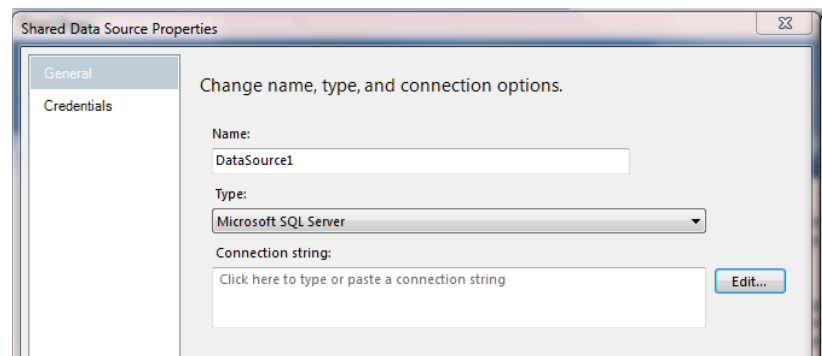
- Open the BIDS program
- Open the New Project Dialog Box
  - File > New > Project
- Under the BI projects subheading, select “Report Server”
- Give your project a name and a directory
- Make sure “Create Directory for Solution” is checked
- Click OK when ready



## Connect to a Data Source

We must now connect to a database containing the data we need to analyze.

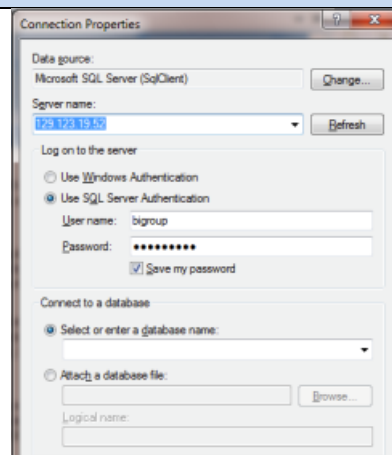
- In the solution explorer, right click the “Shared Data Sources” folder, and select “Add New Shared Data Source”
- Select the “Edit String Option”



## Set Connection Properties

The connection to the database will vary depending on where the database containing your stock data is located. The entries here are specific for the USU MIS department.

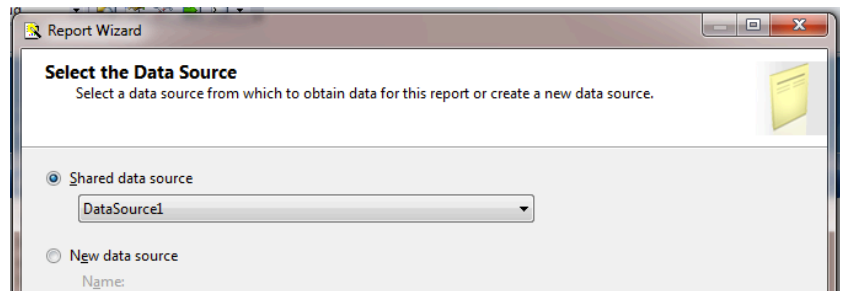
- Server Name
  - User Name
  - Password
- Database Name



## Start the Report Wizard and select your data source

Now that you have accessed your database, you can start generating the report.

- In the solution explorer, right-click the “Reports Folder”
- Select “Add New Report”
- Pass the Introduction Window (if it appears)
- In the “Select Data Source” Dialog, your Data Source should appear in the “Shared Data Source” Subsection
- Make sure your data source is selected and continue



## Enter the parameter query

In the next dialog box, we’re going to write our own query rather than rely on a query builder. This query selects the Ticker, the closing price, the 35-day moving average, the difference between the two, and the month, year, and quarter for the data presented. If your data structure is different, modify the query at your leisure. The report will be a visual representation of this query.

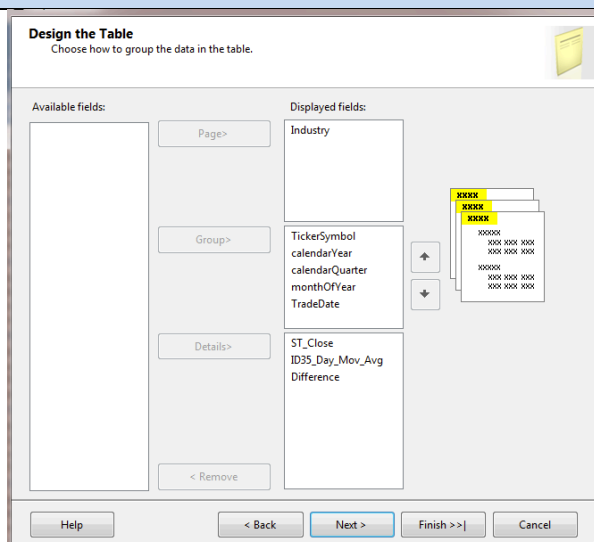
```
SELECT s1.Industry, s1.TickerSymbol, s1.TradeDate, s1.ST_Close, AVG(s2.ST_Close) AS
"35 Day Mov Avg", s1.ST_Close - AVG(s2.ST_Close) AS "Difference", cb.monthOfYear,
cb.calendarQuarter, cb.calendarYear
FROM StockData AS s1 JOIN StockData AS s2 ON
(s1.TradeDate >= s2.TradeDate) JOIN calendarBasic AS cb ON
(s1.TradeDate = cb.FullDate)
WHERE s2.TradeDate >= DATEADD(dd, -35, s1.TradeDate) AND
s1.TickerSymbol = s2.Tickersymbol
GROUP BY s1.Industry, s1.TickerSymbol, s1.TradeDate,
s1.ST_Close, cb.monthOfYear, cb.calendarQuarter, cb.calendarYear
ORDER BY s1.Industry, s1.TickerSymbol, s1.TradeDate
```

## Design the Table

After entering the query, we now begin to show what our report will look like. The options here are largely a matter of preference for design and functionality.

- Select “Tabular” for the Report Type
- Choose how you want to group the data in the table.

What I’ve entered here is what I recommend. Each page of the report will show multiple stocks, divided by Ticker Symbol at the highest level.

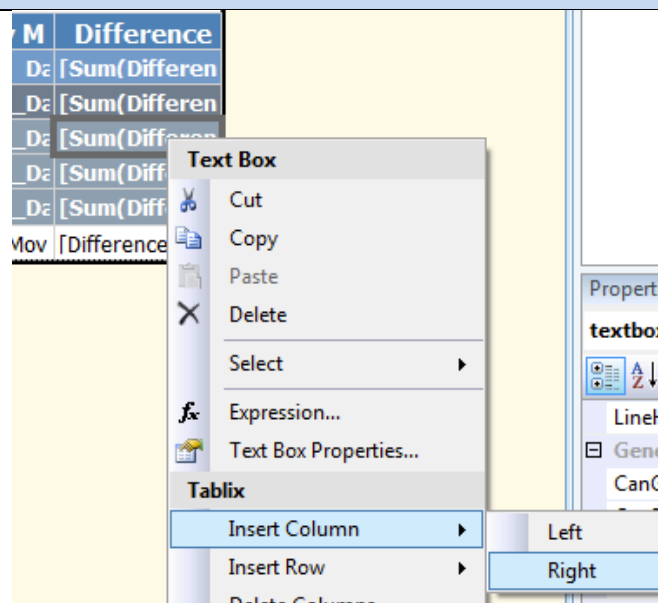


## Make Alterations to include graphics

The report is now in a tabular form, but there are no visual indicators to tell us to buy and sell. We can create them using the tools in the reporting services libraries.

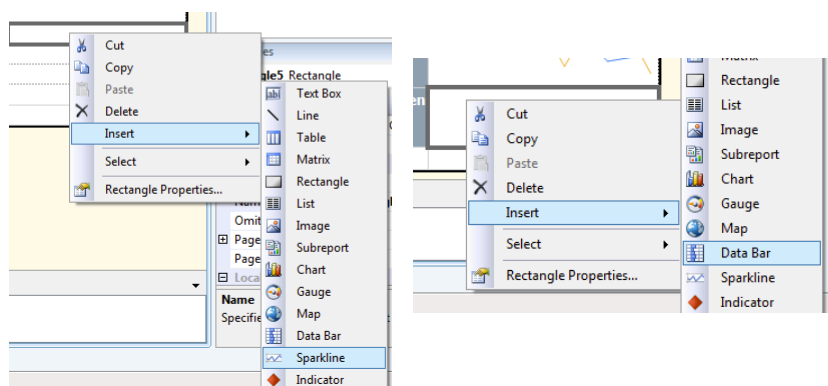
We start by inserting a new column on the right end of the table.

- Right-click on any cell in the right-most column
- Insert column > Right



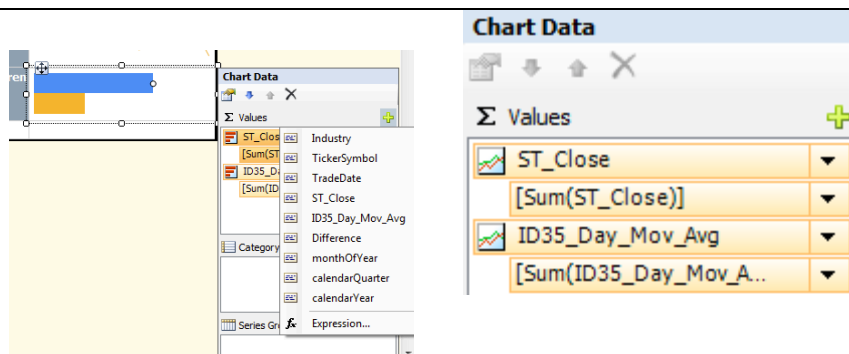
Now we can put graphs in the new column.

- Right click the cell where you want to insert a chart
- Insert > Sparkline
  - Alternatively, you can insert a rectangle first and then insert a sparkline. This allows for greater size and position control on the graph
- For the cell to the right of the [TradeDate] values, a horizontal data bar is much more useful



Now we need to define parameters for the graphs so that they read and interpret the data.

- Double-click any of your sparklines. A chart data box will appear
- Select the button in the box to add values. I have added "ST\_Close" and "35\_Day\_Moving\_Average"
- Order matters in this dialog box. The first entry will correspond with the blue line, and the second with yellow. The more value entries, the more lines in your graph.

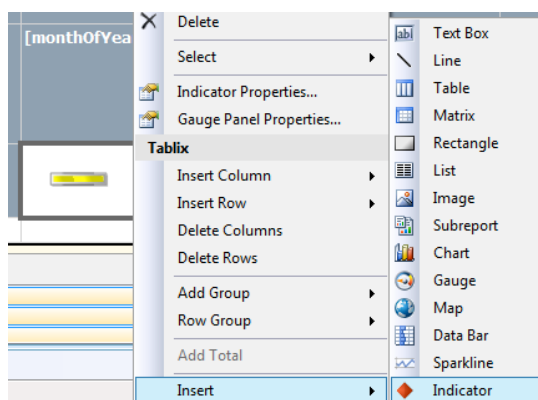
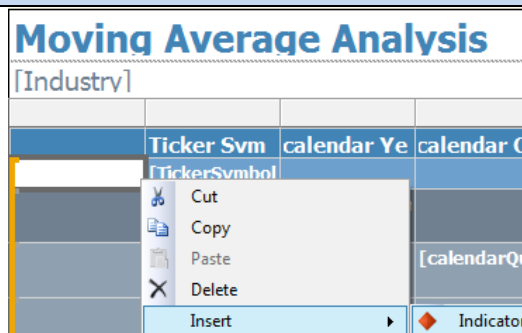


## Insert Indicators

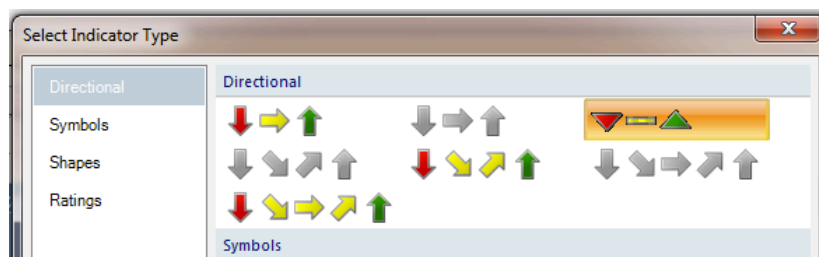
We have now created line and bar charts to be able to interpret the data in our database. Now we can add an extra bell by adding two indicators, one that will tell us if the stock is a good long-term investment, and the other that can tell us if the stock is a good investment day-to-day

- Insert a column to the left of the ticker symbol column
- Insert an indicator in the cell directly below the [monthOfYear] cell.

The location of the indicator is less important than that you know what it means when you see it. When you choose a location for the indicator, bear that in mind.

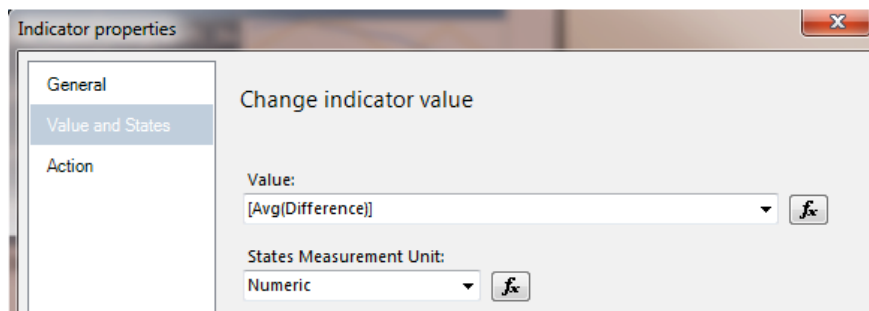



In the new Dialog Window, select the indicator type of your liking. The one I chose can be seen to the right.

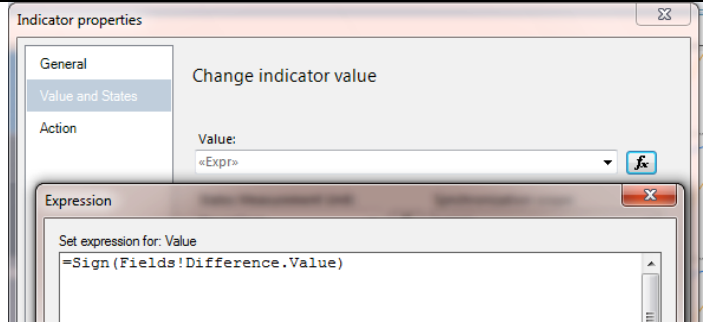


Now that the indicator has been inserted, we need to tell it what to indicate. The process is very similar to the graphs

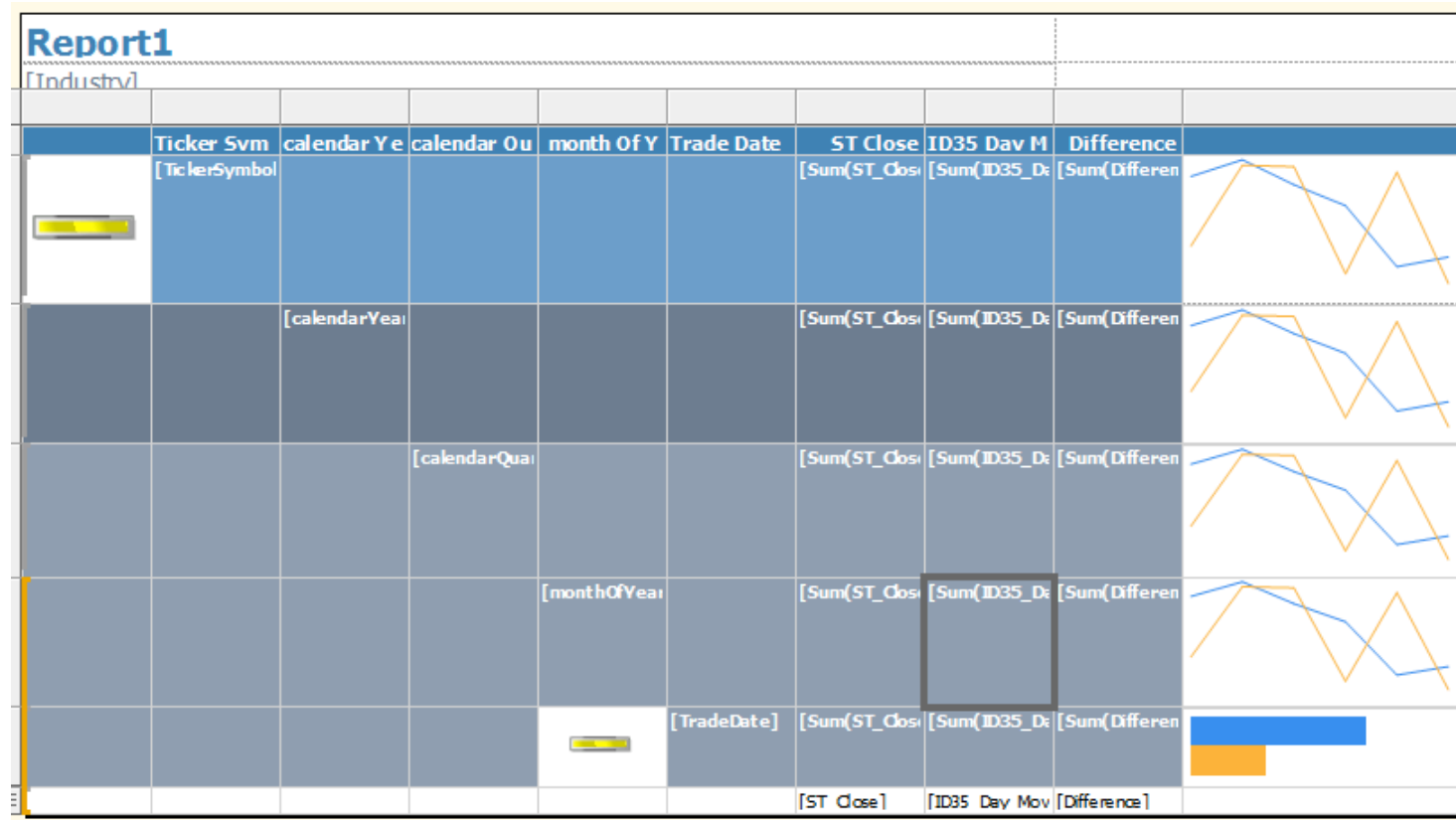
- Double-Click the new indicator
- In the Gauge Data box that appears, select the properties box. A new dialog box opens
- Select “Value and States” and enter the formula to the left to compute the average from the difference column over a year.
  - If the average is positive, it’s a good buy
  - Otherwise, it’s bad



For the other indicator to the left of the individual trade dates, the formula should depend on the sign of the difference. If the difference between the price and the moving average is negative, sell. Otherwise buy. The formula can be set by clicking the  button. Use the options to generate the formula, or you can enter the formula shown here.

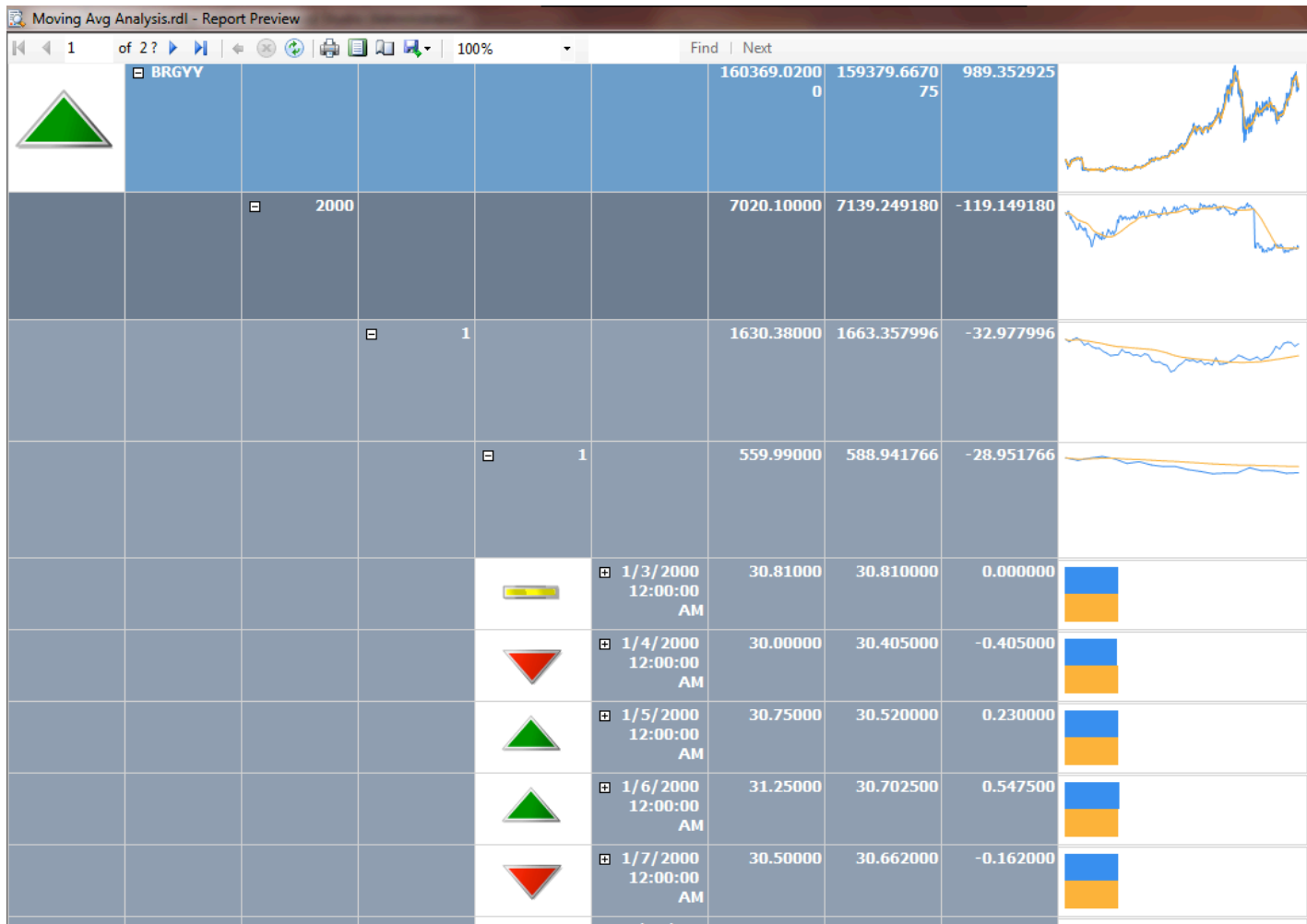


After you're finished entering the indicators and graphs, your report should look something like this in the design window.



Present your report

The report can now be previewed, and we're done! The indicator next to the ticker name is a good indicator of whether or not the stock is good in general. Its day-to-day performance against the moving average is indicated by the indicators next to individual dates. The report is dynamic; selecting the +/- icons at each level will allow the user to see the averages and values at a higher granularity.



## **V. REVIEW**

The goal of this example is to introduce students to the decision-making power that can be derived from raw data analysis in a real-world scenario. The presence of principles from another discipline not MIS but understandable to MIS students is crucial to the effectiveness of this tutorial. The tutorial is also meant to introduce students to BI application use. The more the students use the software to obtain BI, the more comfortable they will become with it. Teachers are encouraged to take liberties and become creative with how they use this tutorial, but I recommend two weeks of class time minimum be dedicated to the subject.

The limitation of this example is the static nature of the data. In a real-time BI scenario, data would be continuously feeding through the database, giving new indicators for stocks as time progresses. This example only tells us which stocks have historically been the best buys or sales. The report is limited in how much it can tell potential users what transactions to make today. In order to do that, we would need real-time, continuously updated data.

## **VI. CONCLUSION**

BI is the use of technology to provide decision-makers with the information they need to make informed decisions. BI success is determined by the effectiveness of data analytics in assisting those leaders with their job, whether it's taking advantage of patterns in industry sales, financing markets, customer feedback, manufacturing, or any other quantifiable activity.

This thesis has reviewed the value of BI in both business and an educational environment and has outlined a method by which students can be taught to use some of the BI tools offered in the Microsoft SQL Server Development Environment, namely Reporting Services, to analyze

stock market data gathered from public sources. The reports generated are meant to assist the students determine which stocks would be a good buy over time and can help them see at what points the stocks turned toward positive or negative profitability.

This thesis has explored the value that business intelligence has to both industry and education, and it has emphasized that the world of higher education is currently not fulfilling the demand of the business world for those who can derive meaning from massive amounts of data. Through use of BI tools such as the Reporting Services available in the Microsoft Business Intelligence Development Studio, we have been able to make a meaningful analysis of stock data and assist those who are interested determine which stocks have been historically profitable both on general and on day-to-day scales.



## BIBLIOGRAPHY

- Anderson-Lehman, R., Watson, H. J., Wixom, B. H., & Hoffer, J. A. (2004). Continental Airlines Flies High with Real-time Business Intelligence. *MIS Quarterly Executive* .
- Black, M., Snow, B., Olsen, D., & Cooney, V. (2010). *Towards a Relevant Business Intelligence: Enhancing the Database Knowledge of MIS graduates*. Logan: Utah State University.
- Brenneman, G. (1998). Right Away and All at Once: How We Saved Continental. *Harvard Business Review* , 76 (5), 162-179.
- Brock, W., Lakonishok, J., & LeBaron, B. (1992, December). Simple Technical Trading Rules and the Stochastic Properties of Stock Returns. *The Journal of Finance* , 47 (5), pp. 1731-1764.
- Case Study II: South Tyneside Healthcare NHS Trust/Cognos. (2004). *Journal of Database Marketing & Customer Strategy Management* , 12 (1), 58-62.
- Chaudhuri, S., & Dayal, U. (1997, March). An Overview of Data Warehousing and OLAP Technology. *ACM SIGMOD Record* , 26 (1), pp. 65-74.
- Chaudhuri, S., Dayal, U., & Narasayya, V. (2011). An overview of business intelligence technology. *Communications of the ACM* , 54 (8), 88-98.
- Codd, E. (1970, June). A relational model for large shared data banks. *Communications of the ACM* , 13 (6), pp. 377-387.
- Codd, E. (1990). *The Relational Model for Database Management*. New York City: Addison-Wesley, Inc.
- Davenport, T. H. (2006). Competing on Business Analytics. *Harvard Business Review* , 84 (1), 98-107.
- Dean, J., & Ghemawat, S. (2010, January). MapReduce: A Flexible Data Processing Tool. *Communications of the ACM* , 53 (1), pp. 72-77.
- Dean, J., & Ghemawat, S. (2008, January). MapReduce: Simplified Data Processing on Large Clusters. *Communications of the ACM* , 51 (1), pp. 107-113.
- Elbashir, M. Z., Collier, P. A., & Sutton, S. G. (2011). The Role of Organizational Absorptive Capacity in Strategic Use of Business Intelligence to Support Integrated Management Control Systems. *Accounting Review* , 86 (1), 155-184.
- Ericson, J. (2010). Old Priorities, New Urgency. *Information Management* , 20 (5), 29-32.
- Giniat, E. (2011). Using business intelligence for competitive analysis. *Healthcare Financial Management* , 65 (9), 142-146.

- Inmon, W. (1996). *Building the Data Warehouse* (2nd Edition ed.). New York City: Wiley Computer Publishing.
- Kent, W. (1983, February). A simple guide to five normal forms in relational database theory. *Communications of the ACM* , 26 (2), pp. 120-125.
- Kimball, R., Ross, M., Becker, B., Mundy, J., & Thornwaite, W. (2010). *The Kimball Group Reader: Relentlessly Practical Tools for Data Warehousing and Business Intelligence*. Indianapolis: Wiley Publishing, Inc.
- Kimball, R., Ross, M., Thornthwaite, W., Mundy, J., & Becker, B. (2008). *The Data Warehouse Lifecycle Toolkit*. Indianapolis, IN: John Wiley and Sons.
- LaVelle, L. (2007, March 19). Does your b-school make the grade? *BusinessWeek* , pp. 68-73.
- Lonnqvist, A., & Pirttimaki, V. (2006). The Measurement of Business Intelligence. *Information Systems Management* , 23 (1), 32-41.
- Lukmana, T., Hackney, R., Popovič, A., Jaklič, J., & Irani, Z. (2011). Business Intelligence Maturity: The Economic Transitional Context Within Slovenia. *Information Systems Management* , 28 (3), 211-222.
- Maier, D. (1983). *The Theory of Relational Databases*. Rockville, MD: Computer Science Press.
- Ranger, C., Raghuraman, R., Penmetsa, A., Bradski, G., & Kozyrakis, C. (2007). Evaluating MapReduce for Multi-core and Multiprocessor Systems. *Proceedings of the 13th International Symposium on High-Performance Computer Architecture* (pp. 13-24). Scottsdale, Arizona: IEEE.
- Simple and Exponential - Moving Averages*. (n.d.). Retrieved October 18, 2011, from StockCharts.com:  
[http://stockcharts.com/school/doku.php?id=chart\\_school:technical\\_indicators:moving\\_averages](http://stockcharts.com/school/doku.php?id=chart_school:technical_indicators:moving_averages)
- Sirkar, S. (2009). Business Intelligence in the Business Curriculum. *Communications of AIS* , 2009 (24), 289-302.
- Slazinski, E. D. (2003). Teaching Data Warehousing to Undergraduates: Tales from the Warehouse Floor. *Proceedings of the 4th conference on Information technology curriculum* (pp. 242-248). New York: CITC.
- University of Arizona MIS Department. (2011). *Courses : Master's in MIS : Department of Management Information Systems*. Retrieved November 30, 2011, from Eller College of Management : The University of Arizona: <http://mis.eller.arizona.edu/masters/courses.asp#587>
- Williams, S. (2011). Five barriers to BI success and how to overcome them. *Strategic Finance* , 93 (1), 27-33.

## **APPENDIX A - Relational Database Theory**

A robust discussion of the virtues of business intelligence will rely heavily on the understanding of basic relational database architecture principles. The relational model for electronic databases was conceived in 1970 by E.F. Codd, who was working for IBM at the time. By his own admission, the relational model is difficult to understand, and misunderstandings abound in database architecture and design (Codd, *The Relational Model for Database Management*, 1990, p. 3). Business intelligence relies heavily on relational database theory; a short review of basic principles in that theory is warranted.

Codd developed a model on the premise that the tree and file system of data organization in use at the time was ineffective and inefficient. Codd asserted it was “not practical to develop application programs which test for all tree structurings permitted by the system” (Codd, *A relational model for large shared data banks*, 1970). Codd’s model is based on the mathematical entity of relations to replace it, drawing attention to its structuring flexibility and ease of access to data while minimizing redundancy of records.

A relation is a “finite set of attribute names” in which the value of each attribute for each tuple in the relation has a finite value within that attributes domain, which is also definable. The attribute domains are “arbitrary, non-empty sets, finite or countably infinite” (Maier, 1983, p. 2). More simply, a relation is a table, the rows representing instances of an object entity of some type, and the columns defining qualities or attributes of each instance. It is assumed that those who are familiar with relational database design have a general understanding of the definition of a relation and that further discussion is not warranted.

Codd also proposed forms of normalization for data storage. A normal form in database architecture is “a restriction on the database scheme that presumably precludes certain

undesirable properties from the database” (Maier, 1983, p. 96). Normalization is simply the enforcement of that restriction. When a database adheres to a normal form, rules to the structuring of data within relations and how the relations themselves are organized are followed. Codd proposed a normalization method for data in 1970 in order to eliminate data that belonged to what he called “non-simple domains” (Codd, A relational model for large shared data banks, 1970). Since their inception, there have been several different normal forms proposed, with five being almost universally accepted as standard. These are aptly named ordinally, the first through the fifth normal forms.

When we speak of business intelligence architecture, we rely on relational database architecture. Business Intelligence systems use relational databases normalized to the third normal form, though the application of the database is unconventional. The third normal form requires that functional dependencies upon non-key attributes be removed to their own relational tables. It is required that the relational tables employed in a third normal form database adhere to first and second normal forms as well.

The first normal form is more matters of definition than they are actual restrictions, but their implementation is important because they are requisite for the third normal form. Per Maier’s definition, a relation is in first normal form if the values of any attribute domain are not composite, non-atomic values. That is, the values in attribute domains cannot be interpreted to hold more than one set of information (Maier, 1983, p. 96). This definition may be difficult to apply and interpreted loosely.

Consider the attribute domain for a birth date. Birth dates belong to a domain that can be interpreted to be singular, non-divisible, and atomic: days on the calendar. However, birth dates can be subdivided into birth years, birth months, and birth days. From this perspective, birth

dates can be considered non-atomic. “As a general guideline, a value is non-atomic if the application deals with only a part of the value” (Maier, 1983, p. 96).

Second normal form demands first normal form, but it is here that the concept of keys are introduced. Keys are unique identifiers of a given tuple that pertain solely to that tuple and that cannot pertain to another. They are pieces of data through which direct access to that tuple’s objective data can be achieved. In second normal form, “every nonprime attribute is fully dependent on every key” of the relation scheme. Whether those keys are candidate keys or primary keys is immaterial. There are no partial dependencies in second normal form (Maier, 1983, p. 99).

(a)

<u>FLIGHT</u>	<u>DAY</u>	<u>PILOT-ID</u>	<u>NAME</u>
112	6 June	31174	Bosley
112	7 June	30046	Brooks
203	9 June	31174	Bosley

(b)

<u>FLIGHT</u>	<u>DAY</u>	<u>PILOT-ID</u>	<u>PILOT-ID</u>	<u>NAME</u>
112	6 June	31174	31174	Bosley
112	7 June	30046	30046	Brooks
203	9 June	31174		

Figure (I) – An example of normalization to third normal form (Maier, 1983, pp. 98-99)

Third normal form requires elimination of transitive dependencies on the primary key in the relation. A transitive dependency on the primary exists when a non-key attribute is fully dependent on another attribute within the relation that is not the primary key. Maier employs an example of airline flights, as described in Figure (I). The pilot’s name is transitively dependent on FLIGHT and DAY in diagram (a) because it is wholly dependent on the PILOT-ID. To resolve the transitive dependency, PILOT-ID and NAME are moved to their own relation in

diagram (b), and the names of pilots can still be accessed with joins. (Joins and this normal form are critical in business intelligence modeling.)

Originally, Codd said the advantages to normalizing relations are threefold: 1) pointers become obsolete in databases, 2) hash addressing schemes in order to access data become obsolete, and 3) indices and ordering lists would become obsolete (Codd, A relational model for large shared data banks, 1970). But these points illustrate the advantages of a small portion of relational database management. Codd lists power, adaptability, safety, productivity, controllability, flexibility, and many other qualities as the strengths of the relational model (Codd, The Relational Model for Database Management, 1990, pp. 431-440). The model has seen more than 30 years of widespread use and has yet to be supplanted as the dominant theory upon which modern databases are constructed.

## **AUTHOR'S BIOGRAPHY**

Landon Mitchel Hemsley was born on November 11, 1985 to Mitchel and Janet Hemsley in Salt Lake City, Utah. He graduated from American Fork High School in 2004 and began attending Utah State University that same year. He left Logan to live and work in Lisbon, Portugal for two years, and upon returning, resumed his studies as an Aggie. Landon is part of the Phi Sigma Phi honor society, has been nominated three times for Tau Beta Sigma, the national business honor society, has been nominated three times for the Golden Key International Honor Society, and has received two USU "A"-Pin awards, one in the College of Humanities and Social Sciences and the other in the Jon M. Huntsman School of Business. Landon graduated from Utah State University in the Fall of 2011, receiving a BA in Journalism and Communications, Summa Cum Laude, and a BS in Management of Information Systems, Summa Cum Laude. Landon is also trilingual, has studied international business, and is a member of the Huntsman Scholars Program at USU. Landon was married on December 28, 2011, and values his lovely wife, Kelsey, and his family above all else.