

Managing the Burst: Optimizing Headcount in a Company with Highly Cyclical Demand

By

Tee Welton

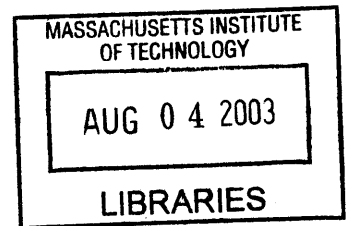
Bachelor of Science in Computer Engineering (1995), University of New Mexico
Master of Business Administration (2000), University of New Mexico

Submitted to the Sloan School of Management and the Department of Electrical Engineering and Computer Science in Partial Fulfillment of the Requirements for the Degrees of

Master of Science in Electrical Engineering and Computer Science
and
Master of Science in Management

In Conjunction with the Leaders for Manufacturing Program at the
Massachusetts Institute of Technology
June 2003

©2003 Massachusetts Institute of Technology. All rights reserved.



Signature of Author _____
Department of Electrical Engineering and Computer Science
Sloan School of Management
May 2003

Certified by _____
Thomas Roemer
Assistant Professor of Operations Management

Certified by _____
Roy E. Welsch
Professor Of Statistics and Management Science, Director CCREMS

Approved by _____
Arthur C. Smith
Chairman, Committee on Graduate Students
Department of Electrical Engineering and Computer Science

Accepted by _____
Margaret Andrews
Executive Director of Masters Program
Sloan School of Management

BARKER

Managing the Burst: Manufacturing Headcount Optimization in a Market with Cyclical Demand

By

Tee Welton

Submitted to the Department of Electrical Engineering and Computer Science and the Sloan School of Management on May 18, 2003 in partial fulfillment of the Requirements for the Degrees of Master of Science in Electrical Engineering and Computer Science and Master of Science in Management

ABSTRACT

Dell is in an industry that has very high variability and cyclical demand bursts. Dell's business model, Build to Order and Just-In-Time manufacturing dictates that they will always have variability in their production facilities. The purpose of this thesis is to analyze the production system and come up with a headcount strategy to help Dell manage these bursts of variability better within the Topfer Manufacturing Center.

Through the analysis completed at TMC, three objectives were accomplished. First, an analysis of the constraints within the factory was completed to help Dell better understand its manufacturing floor. The thesis identified the hard constraints in the factory, as well as modeled the factory area capacities with respect to Headcount. Some key learning from this analysis was discussed for each area as well.

After the capacities were determined for each area, the factory was looked at from a macro-level. The constraint of the factory was identified through use of Block and Starve time for each area. Further, the design of the factory was discussed and the best area for the constraint to be located in the factory was identified. An optimization was then built to statistically minimize the amount of Headcount needed on the production floor in order to keep the constraint in the Build area and meet the demand of the factory.

Additionally, an analysis was completed to try to correlate non-production outcomes, such as retention rate, quality, or injury rates to production factors. There was high correlation between Quality, measured in defects, and production factors such as Labor Hours, Day of the Week, Hour of the Day, and the Workweek. Finally, recommendations were given with every step of the analysis that would enable Dell to get better correlations using the data they have already, and methods that might be explored to collect new information that might reduce variability of the models as well.

Thesis Supervisors:

Thomas Roemer
Assistant Professor of Operations Management

Roy E. Welsch
Professor of Statistics and Management Science
Director CCREMS

ACKNOWLEDGEMENTS

This thesis is dedicated to my lovely wife, Ronda, and my newly born daughter, Faith. Thank you for your support, love, and partnership over the last six years, Ronda. Eight moves in five years is a lot to ask from anyone. Thank you for being with me through the best years of my life. I thank God for placing you in my life. God is great, and has spared my life for some reason. May our lives please Him and reflect His glory.

I would like to acknowledge the Leaders for Manufacturing Program for its support of this work. LFM provides a unique opportunity to merge engineering and management academia as well as industry experience. To Don Rosenfield, the LFM staff, and the LFM founders and supporters, I am appreciative.

I would like to extend my thanks and gratitude to Richard Komm and Dell Computer Corporation for making this internship experience one of the greatest work experiences of my life. Additionally, I would like to thank Maribel Sierra for her internship guidance. It was nice to have an LFM colleague available at Dell to help me through my internship. I would like to add special thanks to Jason Durst for his technical project support and inspiration. His help was crucial. Additional thanks goes to John Egan, Steve Cook, and Dick Hunter for their support of the LFM program.

I would like to acknowledge and thank my thesis advisors Roy Welsch and Thomas Roemer for their visits and extensive help throughout this internship. Their expertise enriched the learning opportunity at Dell, and helped make this internship experience a great one.

I would like to thank my corporate sponsor, Intel Corporation, as well as LFM program advocates Frank See and Bob Baker for giving me the opportunity to continue my education at MIT through the LFM program. Additionally, I would like to thank Morgan Burke and Brian Harrison for their active support of the Intel LFM program and their interaction with the LFM Class of 2003.

I would like to thank my entire family for all their love and support. Without family, all else is meaningless. Special thanks to my loving parents Ted and Brenda, for raising me right, with a competitive spirit, but compassion for others. Thanks to my brother and good friend Bruce. Thanks for your help through the process and for teaching me a little about writing. Thanks to my sister and business guru Kristi and her family Frank and Brett. Here's hoping many interesting conversations will come from this experience.

Additionally, I would like to send special thanks to my wife's parents Dave and Roberta Glasgow, and my wife's sister and family, Charlene, Mark, C.J. and Tiffany Redd. They accepted me as family, have been supportive throughout the years, and allowed me to steal Ronda.

Table of Contents

ABSTRACT.....	3
ACKNOWLEDGEMENTS	5
1. THESIS DESCRIPTION	11
1.1 THESIS CONSIDERATIONS	12
1.2 CURRENT METHODOLOGY FOR HEADCOUNT FORECASTING.....	12
1.3 THESIS STRATEGY	16
1.4 SUCCESS CRITERIA	16
2 DELL INTRODUCTION AND OVERVIEW	17
2.1 THE HISTORY OF DELL CORPORATION.....	17
2.2 DELL’S MISSION	18
2.3 DELL’S COMPANY OBJECTIVES	18
2.4 DELL PRODUCTS AND SERVICES	19
2.5 DELL’S “DIRECT” BUSINESS MODEL	21
2.6 TMC MANUFACTURING ENVIRONMENT	25
3 ANALYZING DELL’S TMC MANUFACTURING PROCESS	26
3.1 PROCESS DESCRIPTION	26
3.2 FINDING THE TRUE FACTORY CONSTRAINT	31
3.3 FINDING THE FACTORY HARD CONSTRAINTS.....	32
3.4 ANALYZE PRODUCTION CAPACITY IN EACH AREA	40
3.5 EXAMPLE - PREDICTING AREA OUTPUT BY HOUR	63
4 BUILDING A NEW HEADCOUNT OPTIMIZATION MODEL.....	66
4.1 DETERMINING MEAN OUTPUT AND STANDARD DEVIATION FOR A SHIFT	67
4.2 CHOOSING A CONSTRAINT	73
4.3 FACTORS USED TO CREATE THE HEADCOUNT OPTIMIZATION	76
4.4 CREATING THE BUILD HEADCOUNT OPTIMIZATION.....	79
4.5 KITTING HEADCOUNT OPTIMIZATION.....	79
4.6 BOXING HEADCOUNT OPTIMIZATION	82
4.7 OPTIMIZATION MODEL SUMMARY	84
5 HEADCOUNT MODEL RESULTS AND RECOMMENDATIONS.....	84
5.1 TEST RESULTS	86
5.2 HARD CONSTRAINT ANALYSIS RECOMMENDATIONS	87
5.3 CAPACITY DATA COLLECTION RECOMMENDATIONS	89
6 INCORPORATE NON-PRODUCTION FACTORS.....	97
6.1 QUALITY.....	97
6.2 RETENTION RATE.....	105
6.3 INJURIES.....	105
7 SUMMARY AND CONCLUSIONS	110

APPENDIX..... 112
REFERENCES..... 122

List of Figures

FIGURE 1: OLD MODEL OUTPUT BY AREA.....	15
FIGURE 2: MANUFACTURING PROCESS	26
FIGURE 3: TOTAL BURN TIME AVERAGE PER UNIT.....	37
FIGURE 4: DISTRIBUTION OF BURN UNITS PER HOUR	38
FIGURE 5: BURN EXCESS CAPACITY OVER BUILD BY SHIFT AND BY HOUR.....	38
FIGURE 6: KITTING REGRESSION ANALYSIS	49
FIGURE 7: KITTING RESIDUAL ANALYSIS	50
FIGURE 8: KITTING RESIDUAL DISTRIBUTION AND GOODNESS-OF -FIT	51
FIGURE 9: BUILD REGRESSION ANALYSIS.....	55
FIGURE 10: BUILD RESIDUAL ANALYSIS	56
FIGURE 11: BUILD RESIDUAL DISTRIBUTION AND GOODNESS-OF -FIT	56
FIGURE 12: BOXING REGRESSION EQUATION ESTIMATE	58
FIGURE 13: BOXING REGRESSION ANALYSIS.....	61
FIGURE 14: BOXING RESIDUAL ANALYSIS.....	62
FIGURE 15: BOXING RESIDUAL DISTRIBUTION AND GOODNESS-OF-FIT	62
FIGURE 16: PREDICTED OUTPUT WITH NORMAL DISTRIBUTION OF RESIDUALS.....	66
FIGURE 17 - DISTRIBUTION OF PREDICTED SHIFT OUTPUT - ACTUAL SHIFT OUTPUT.....	73
FIGURE 18 – OLD MODEL OUTPUT BY AREA.....	74
FIGURE 19 - THE NEW FACTORY HEADCOUNT MODEL	76
FIGURE 20 - PROBABILITY OF AT LEAST “A” UNITS BEING PRODUCED.....	78
FIGURE 21 - ADJUSTED PRODUCTION VOLUMES OF KITTING AND BOXING OUTPUT WITH BUILD OUTPUT	80
FIGURE 22 - REAL TIME BURN MONITOR SCREEN.....	88
FIGURE 23 - PLACEMENT OF BLOCK AND STARVE MONITORS IN THE FACTORY.....	91
FIGURE 24 - KITTING LINE 4 REAL TIME BLOCK MONITOR	92
FIGURE 25 - IMPACT OF HEADCOUNT ON QUALITY	101
FIGURE 26 - RESIDUAL PLOT FOR QUALITY DATA.....	102
FIGURE 27 - QUALITY REGRESSION ANALYSIS WITH PERCENT TEMPORARY WORKER INCLUDED	103
FIGURE 28 - MEAN INJURY WITH DIFFERENT OUTPUT LEVELS	107
FIGURE 29 - INJURIES VS. OVERTIME PIE CHARTS.....	108
FIGURE 30 - EXPECTED VERSUS ACTUAL INJURY AND NEAR MISS RATE FOR DIFFERENT OVERTIME RATES	108
FIGURE 31 - INJURY AND NEAR MISS RATE BY DAY OF THE WEEK.....	109

List of Tables

TABLE 1 - DELL'S COMPUTER PRODUCTS	20
TABLE 2 - DELL'S OTHER PRODUCTS AND SERVICES	21
TABLE 3: BLOCK STARVE ANALYSIS	41
TABLE 4: KITTING REGRESSION EQUATION ESTIMATE	45
TABLE 5: BUILD REGRESSION EQUATION ESTIMATE	52
TABLE 6 - EXAMPLE FACTORS FOR DETERMINING OUTPUT	64
TABLE 7 - TMC FIRST SHIFT WORK HOUR RULES	68
TABLE 8 – TMC SECOND SHIFT WORK HOUR RULES	69
TABLE 9 - SHIFT OUTPUT FOR AN 8-HOUR SHIFT ON MONDAY WITH 343 WORKERS	70
TABLE 10 - LIST OF VARIABLES FOR BUILD OPTIMIZATION	77
TABLE 11 - CASH SAVINGS OF OPTIMIZED VERSUS ACTUAL HEADCOUNT LEVELS.....	86
TABLE 12 - PRODUCTION FACTORS USED TO PREDICT QUALITY DEFECTS.....	98
TABLE 13-PREDICTION EQUATION FOR QUALITY VERSUS PRODUCTION FACTORS	99
TABLE 14 - TOTAL OUTPUT/LABOR HOURS EXAMPLE	100
TABLE 15 - PREDICTOR EQUATION FOR QUALITY VERSUS.....	103

1. THESIS DESCRIPTION

Dell Corporation is a leading computer manufacturer in an industry with a very cyclical demand cycle. During a given quarter, 15% of the period's sales volume occurs during the first month, 35% during the second, and about half of the quarter's sales volume occurs during the third month. Dell's build-to-order business model introduces additional variability into fulfillment projections. With a build-to-order business model, Dell manufactures exactly what each customer orders on a given day, which makes forecasting daily demand tenuous, and daily production variability high. With this variability affecting production, managing headcount at Dell's build-to-order manufacturing plant can be tricky. This thesis provides an analysis of the manufacturing system at Dell's highest volume manufacturing factory, the Morton L. Topfer Manufacturing Center, TMC, and creates a headcount-staffing optimization model for predicting optimized staffing levels under conditions of high demand volatility. Little in-depth analysis has been done on how headcount, or the number of people needed to work within each production area on the floor, impacts production for the Topfer factory. Several key factors affect a factory's ability to maximize production. Some of those factors include, headcount levels, downtime, shift lengths, and even the time and day of the week. This tool will help TMC to understand how to manage their headcount more effectively and see what impact headcount has on quality, safety and retention rate of employees in the factory.

The strategic objective of this thesis is to optimize the headcount-staffing level of the TMC factory by evaluating its productivity, retention rate, safety, and quality indicators. The first step in this process is to identify and characterize the different constraints on the factory floor. By understanding and controlling where the constraint is in the factory, we can identify the boundaries for productivity that can be achieved in the factory, which will limit our productivity in our headcount optimization. Additionally, this thesis will perform a productivity analysis of each area in the factory. By knowing the constraints and productivity in each area, factory area burst capacities will be defined. Finally, some analysis will be completed on how headcount and work schedules (regular, overtime and weekend schedules included) impact productivity, quality, safety, and retention of headcount. More strategically, this thesis will present a framework and tools for use by Dell to optimize the headcount in each area of the TMC in order to maintain an

efficient factory, with maximum throughput with minimal headcount. Ultimately, Dell will be able to use this headcount optimization model to minimize costs in their production facility through headcount reduction, without compromising meeting production output targets.

1.1 Thesis Considerations

While headcount modeling is usually done on a global level, the scope of this thesis is limited only to the impact of headcount at the TMC only. As mentioned, this analysis will include modeling for Quality, Safety, Retention Rate, and Productivity trends. To create such a model, a study on the factory will need to be completed to identify operational bottlenecks and productivity factors. This optimization model will help Dell to understand and predict headcount within the factory, and to better manage the constraints within the factory through changing staffing levels in the different areas of the factory. Additional analysis should be done to understand how work schedules impact productivity. For example, differences in productivity between the normal, scheduled work hours, and the overtime or weekend work hours affect productivity consistency. It will be important to recognize these differences to ensure work schedules are developed to ensure highest productivity of the employees. Finally, this study should present how a new headcount management system should be implemented, and also identify any other areas within the TMC where Dell should improve its headcount management.

1.2 Current Methodology for Headcount Forecasting

In order to fully understand the TMC headcount optimization problem, it is necessary to review TMC's current headcount methodology and the shortfalls that are inherent in it. Following is a summary of the headcount planning procedure used to predict headcount in TMC over the three years prior to the headcount optimization project.

1.2.1 Strategic Headcount Forecasting

Strategic headcount forecasting is a complex procedure that requires much work and coordination between departments. Strategic headcount is forecasted on a quarterly basis. A headcount-forecasting group meets midway through the fourth quarter to forecast headcount for the next full year. These projections are then adjusted once a quarter, until the fourth quarter of the ensuing year, when the cycle is repeated.

The forecasting model is first created as a generic “current” state headcount model that measures productivity levels as simply boxes built per person for each area in the plant at the time of the origination. In the current forecasting model, to calculate productivity, engineering standards based on time studies taken on the production floor are used to estimate production capability in each area. As improvement projects occur in the factory, new time studies must be performed and the new productivity standards are rolled into the headcount model. Many other factors that may be hard to predict, such as downtime, fatigue, employee vacation time, training time which impacts floor headcount, and personal absence time are approximated and factored into the headcount forecast model as well. Using a standard spreadsheet, these factors are applied to determine the long-range forecasting model for the factory. Finally, the headcount forecasting group looks at all of the projects that are coming in the future quarter. These projects could include productivity enhancements, new equipment or automation system installs, changes to factory products, or floor layout changes. For each of these projects, an estimate for productivity after the change is made is calculated. Then, an optimum headcount level is defined for every different combination of changes that could occur over the forecasted quarter that could be implemented concurrently on the floor. Each of these model iterations must also have predicted output volumes and productivity estimates for each week in the quarter.

While this system produces an adequate estimate of headcount, there are a few shortfalls to this planning approach. The biggest shortfall is that some of the factors used in the headcount model, such as factory downtime and engineering standards are goals that are passed down from corporate headquarters and are not based strongly on historical trends, but on idealized estimates of what the target should be over the forecasting period. Another shortfall in this method is that the currently used headcount model and spreadsheet have not been accurately adjusted over the past three years for productivity enhancements, and are strongly based on approximate goals instead of historical trending. While a headcount forecasting process is needed, the present approach can be better optimized.

1.2.2 Tactical Headcount Forecasting

While strategic headcount forecasting planned headcount for the whole quarter, tactical headcount is forecasting of headcount in each area for the upcoming week. To do

this, the staffing group in TMC utilizes the quarterly strategic headcount model that matches the manufacturing center's current unique scenario of workweek and implemented projects. Then, headcount is adjusted to account for demand differences between the week's actual expected demand and the expected demand as predicted by the strategic headcount model. The headcounts in each work area are in turn adjusted as well. Dell relies heavily on staffing temporary employees to survive large swings in demand. If Dell's full-time headcount cannot meet the demand, Dell relies on temporary employees from an external company to supply sufficient headcount to meet Dell's headcount needs. While temporary employee staff levels are projected through the baseline headcount model, it is up to the staffing group to ensure that accurate headcount resources are available each week to meet the demand volumes and compensate for leave of absences or permanent staffing level fluctuations.

While the tactical headcount forecasting model projects headcount needed from week to week, it too has some areas that could be improved. As with the strategic model, one area needing improvement is that many of the factors (such as downtime, etc) are based on corporate goals instead of actual data. Additionally, this model fails to adequately track where temporary employees end up working. At the beginning of each week, a temporary headcount level is requested for each work area, but as workers show up, they may be redirected to other areas. By not accurately tracking where temporary employees are working, it is hard to know how many workers are actually working in each area. Consequently, this practice makes it very difficult to calculate "real" output per worker data for each area for use in future model forecasting. Historical reporting and headcount levels are not fully accurate.

One other theoretical problem with the tactical staffing model lies in the fact that staffing is determined for each area in order to meet the demand of the factory. In itself, this is not a problem. However, each factory area has some amount of variability inherent in its production output per person, caused by downtime, breaks, or productivity lags that exist from time to time. While the output per person within a given an area hovers around a mean, it is really normally distributed around that mean, and on rare occurrences, there could be large deviations from the expected output. This variability can be seen in the following Figure of box plots for each area.

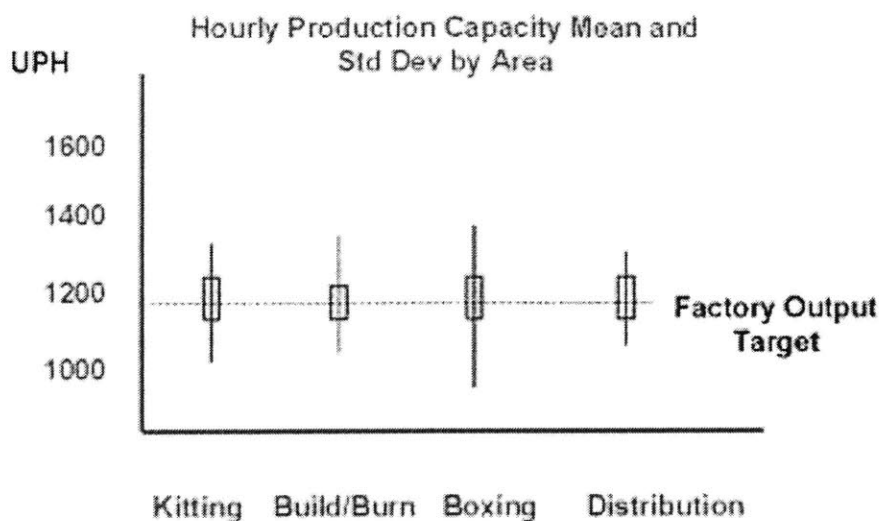


Figure 1: Old Model Output by Area

The above graph presents the output in Units Per Hour for each area in the factory line. As the product moves through the factory, it will proceed from Kitting through Build/Burn, Boxing and finally Distribution in that order. Each of these areas will be introduced more thoroughly in later sections in the paper. If the output goal for the factory was to produce 1200 Units Per Hour, and the factory were to employ the same amount of headcount every day over multiple days, the actual output per hour would vary. The average output per hour may be 1200 units, as the factory would like, but the range of output level would actually vary greatly by area, as demonstrated in the above box plot. Hypothetically, for Kitting, if enough employees were hired to produce the average 1200 units per hour, as Figure 1 asserts, over time the range of outcomes in the Kitting area is generally between 1020 and 1300 Units Per Hour with that same headcount. Figure 1 also illustrates that 50% of the points will fall within the boxed area, or between 1150 and 1220 for the Kitting area in the above plot, while the remainder of the output level with that same headcount will be outside of that range (between 1020 and 1150 and between 1220 and 1300 Units per Hour). Because the output for each of the areas has some amount of variability, if the output target for each area is set to the output target for the factory, the factory will most likely miss its output target. The theory of constraints asserts that a factory will only produce goods as fast as its slowest area, so if a single area misses its target, and each area depends on the input or output of the other areas, the factory as a whole will miss its target more times than not. A new headcount

model will need to consider this variability in its formulation and be created to ensure that the factory can meet its output objective, despite the variability inherent in each area.

1.3 Thesis Strategy

The goals of the study at Dell were to better understand the factory floor and implement a headcount model that will help the TMC factory better predict headcount through swings in the demand that occur in this cyclical industry. Some shortcomings existed in TMC's headcount planning, as already mentioned, that needed to be addressed. To improve the headcount planning I implemented the following process.

The first step was to identify and understand key non-negotiable parameters that are important in understanding the factory, which would influence the headcount optimization model. These parameters include items such as the maximum capacities in each work area, and factory-designed hard constraints. Once hard constraints had been identified and quantified in the factory, productivity information was collected for each area in the factory. The information was used to determine how each area's capacity related to that area's headcount level.

After the constraints and productivity information were determined, a model for each production area was built to predict the output expected given varying headcount levels. Once each area was analyzed, the overall factory was analyzed to determine the best location for the constraint to be located as well. With that information, an overall model was created to determine the appropriate headcount necessary in each area to meet the output goals in the factory while keeping the constraint where it was designed.

After analyzing the results of the optimization model additional analysis was performed to suggest ways that Dell could better control its manufacturing productivity, and make its headcount modeling more accurate.

Finally, while productivity is the key ingredient in determining headcount, there are other non-production factors, such as quality, safety, and employee retention rate, that impact headcount strategy as well. After productivity was measured, some analysis was needed to attempt to correlate quality, safety, and employee retention rate to the production factors included in the headcount model.

1.4 Success Criteria

There are three major criteria for measuring the effectiveness of this thesis. First, upon completion of this study, Dell should be able to better understand how headcount

impacts its production facility by looking at the outcome of our analysis within each area. Using this data, Dell should be able to understand where its bottlenecks are within the TMC and should have an understanding of how to maintain designed constraints within the factory. The second success criteria is that Dell should be able to apply a headcount optimization model in its management schema to minimize headcount in the factory, while still maintaining the factory production rates needed to meet customer demand each day. This model should be validated against real production data. Finally, an analysis should be completed that will try to correlate non-production factors, such as retention rate, quality, or injury rates to production factors. With good correlations, Dell will better understand how changing headcount levels in the factory will impact these non-production factors.

2 DELL INTRODUCTION AND OVERVIEW

In order to better understand the environment for this thesis study, this section gives a broad overview of Dell, its origins, products, goals, and the manufacturing environment at the Morton L. Topfer Manufacturing Center (TMC) at the time of the study. By understanding the TMC factory environment and Dell's corporate culture and strategy, the importance and impact of the thesis conclusions will be better understood.

2.1 The History of Dell Corporation

In 1983, as a freshman at the University of Texas, Michael Dell began his computer building business. By buying surplus computers at cost from local computer dealers and adding features such as extra memory or disk drives he was able to resell them for a modest profit (10-15%). It became obvious to Dell that as computer manufacturer would strive to build computers in volume, they would forget about the special needs of the customers. This presented Michael with an opportunity to enter the market, making computers configured for individual customers. This idea was the start of the Dell Direct model, in which Dell would sell to the customer exactly what they wanted. By Spring Break of his freshman year, Michael was selling \$50,000 a month of personal computers. Having hired his first employees and opening his first office, on May 3, 1984, Dell Computer Corporation was founded (originally under the name of "PC's Limited"). In its first month, Dell's sales were above \$180,000. Today, Dell Corporation is among the 500 largest American Companies, making Michael Dell the youngest CEO to have reached the FORTUNE 500. Dell Corporation and Michael Dell have won many

prestigious awards for excellence. Michael Dell has been named Young Entrepreneur of the year 6 times (1986-1991) by the Association of Collegiate Entrepreneurs, “CEO of the Year” from Financial World Magazine, and Entrepreneur of the Year by INC. Magazine (1991), Dell Corporation’s manufacturing system has been recognized as a world-class leader as well. The production facility that was the subject of this headcount optimization, the Topfer Morton Manufacturing Center (TMC), also stands out as an exemplary production facility, winning Industry Week’s Best Factory in 2001. Dell Computer Corporation today employs 36,500 people in 34 countries.

2.2 Dell’s Mission

Dell’s mission is to be the most successful computer company in the world by delivering the best customer experience in the markets that they serve. Dell aims to meet its internal and external customer expectations by providing the highest quality products, leading technology, competitive pricing, individual and company accountability, best-in-class service and support, flexible customization capability, superior corporate citizenship, and financial stability.

2.3 Dell’s Company Objectives

Dell evaluates its current performance as a company by focusing on the following four key areas: Customer Experience, Globalization, Product Leadership, and Winning Culture. Success metrics in these four areas are rolled into company incentives as bonus targets to focus the company on meeting these goals. Each of these areas is briefly described below.

2.3.1 Customer Experience

From a Dell perspective, customer experience embodies more than just the quality of the product that the customer receives. At Dell, Customer Experience embodies the whole customer experience with Dell, from shopping and finding the right product and the purchasing experience, to customer service after the computer is in their home or office. Dell believes that some of the areas to measure success at Dell are through sales and services, price, performance of the systems, and through tracking revenue generated for services beyond the computer sale itself.

2.3.2 Globalization

Globalization is the expansion of Dell throughout the world. With a saturated American market for personal computers, much of the future growth of this industry will be outside the United States border. Dell wants to achieve some scaled growth in the faster growing world markets such as China, Germany, France, and Japan. Beyond simply meaning expansion into new regions throughout the world, globalization also covers the spread of standards and best practices throughout corporate Dell as well.

2.3.3 Product Leadership

Product Leadership comes in three different forms. First, striving for Product Leadership is striving to become the top personal computer company by creating extra value through innovation, while maintaining price and cost leadership for all products. Additionally, Product Leadership is growth for Dell's younger products such as servers, storage, and services. Finally, Product Leadership is the spread of Dell into other businesses that offer high margins and would fit strategically into Dell's business model, such as printers, overhead projectors, and handheld computers.

2.3.4 Winning Culture

Winning Culture encompasses Dell's vision to maintain global values across its organization and to raise potential leaders for the future. Winning Culture also aims to expand workforce diversity, implement more quality and cost-savings initiatives at all levels of the company, and cultivate a performance-based culture. Additionally, a good amount of Winning Culture focus is on making Dell a great place to work.

2.4 Dell Products and Services

Dell offers a wide array of technical products and services to its customers. While its main focus has been on personal computers, over the last six years its product portfolio has broadened considerably.

2.4.1 Computer Systems

Dell offers a variety of computer systems. Dell's products and the product descriptions are listed in Table 1 below.

Product	Description
Dell Optiplex Desktops®	The Optiplex Desktops are Dell's solution for corporate and institutional customers. They focus on reliability and low cost.
Dell Dimension Desktops®	These desktops are generally higher end technology computers for businesses requiring higher end performance.
Dell Latitude Notebooks®	These laptops are high-end laptops that focus on specific business needs such as performance, size, and flexibility.
Dell Inspiron Notebooks®	Targeted at small to medium sized companies, these notebooks provide good performance at lower prices.
Dell Precision Workstations®	These one- or two- processor workstations are designed to run the highest performing software applications for corporate customers, including graphic, digital content creation, and computer-aided design.
Dell PowerEdge Servers®	Since September of 1996, Dell has become the fastest growing company in high-end performance servers. Generally, these are high price servers for shared access and powerful performance.

Table 1 - Dell's Computer Products

2.4.2 Dell's Other Products and Services

Beyond selling personal computers, Dell has ventured into new markets that are similar to that of the personal computer assembly business. Each of the products and services are considered complements to their personal computer business. These products and services are listed below.

Product	Description
Dell PowerVault® and Dell/EMC Storage Products®	These are high-end storage systems that are designed to meet a wide range of specific customer storage needs.
Dell PowerConnect® Switches	These are standards-based network switches that connect computers and servers in small to medium size organizations.
Dell Monitors	These monitors span all levels of performance, from traditional tube monitors, to flat panel display.
Software and Peripherals	Dell also can be used as an outlet to purchase software and peripheral systems directly without going to a large distributor. Additionally, cameras, monitors, projectors, software, memory, notebooks and accessories are some of the products offered.
Dell Printers	This is an emerging market for Dell that has high margins. This is a market in which Dell can be successful by applying its direct order business model. By entering this market, Dell can increase their profitability as well as minimize the financial advantage of their competitors through sales of these high-margin printers. Dell will target the small to medium-sized customers with personal printing needs.
Dell Handheld	Like printers above, this market is an emerging market with high margins that Dell hopes to capture. Dell will begin assembly of the portable devices within the year as well. These handheld devices will target all levels of the handheld market.
Dell Services	Dell offers consulting services, enterprise services, and Dell computer management services to corporations.

Table 2 - Dell's Other Products and Services

The Morton L. Topfer Center manufactures two different products. For the most part, TMC is responsible for the manufacture of all of the Dell Optiplex Desktops® for the company. Additionally, during the peak season just before Christmas, TMC also builds Dell Dimension Desktops® as well. The products are quite different, but manufacturing of the two products is similar.

2.5 Dell's "Direct" Business Model

Also referred to as "Dell's Direct Model," Dell sells all of its products straight to the customer, and each exactly as the customer wants it to be built. Unlike most of Dell's

competitors, there are no middlemen between Dell and its customers, such as retailers or distributors. This strategy has catapulted Dell to the top of the PC market. Dell's Business Model can be broken down into 5 major components. They include eliminating resellers markup, time to volume, build to order, service and support, and low inventor and capital investment. Each of these business model components is explained in more detail below.

2.5.1 Eliminating Reseller Markup

Because Dell sells directly to the customer, there is no retailer between Dell and its customers to artificially "mark up" the price. Most of Dell's competitors rely on retailers to market, distribute, and sell their products. These retailers require some portion of profit. Avoiding resellers removes an extra layer of cost that Dell has to pay to sell a product. The customer pre-orders its product from the Dell website or phone line, and usually times the order starts on the assembly line within 24 hours. Within a week or so the product arrives at the customer's doorstep ready to be used. By ordering direct, the customer can bypass the distributor and retailer markups, making the Dell PC cheaper to its customers.

2.5.2 Build to Order

By pre-ordering Dell products, customers are assured to get exactly what they want, instead of settling for a pre-configured system that has been pre-made and is being sold in bulk rate at a store (for a higher price). Customers order customized personal computers via the Dell.com website or through a toll-free phone number.

Build to Order ensures that the customers have the latest technologies and the lowest costs available. Additionally, by not stocking shelves with pre-built computers, Dell keeps its inventories of built computers and raw materials low.

2.5.3 Low Inventory/Capital Investment

As mentioned, "Dell's Direct Model" has huge implications on the company's inventory amount and capital investment strategy. Dell only builds the exact product that its customers order and Dell doesn't start building it until after it has been ordered. Because of this, Dell can implement "just-in-time" manufacturing. Dell can order only the parts from its suppliers that it needs to fill its orders. Dell has made arrangements with most of its suppliers for raw materials to be available for Dell to pull and use to fill

orders at all times. However, Dell does not own the raw material inventory until it is pulled to fill an order. Because of this arrangement and business model, Dell, at any given time, has extremely low inventory compared to its competitors. Since 1994, Dell has reported that it reduced its inventory from 55 days worth of material to around six days worth of material. Compared to competitors such as Hewlett Packard, which holds about 50.7 days of inventory, it is clear that Dell holds a huge advantage. Two financial impacts of this low volume strategy become apparent. First, by holding nearly 50.7 days¹ of inventory, Dell's competitors have to tie up excessive cash in raw material and credit to finance that inventory. In the case of Hewlett-Packard, inventory cost equated to \$5.2 billion dollars² tied up in inventory in 2001. If Hewlett-Packard held only six days of inventory, it would be able to free up \$4.6 billion dollars for other investments. Dell has a great advantage by limiting the holding costs associated with inventory.

A second major advantage Dell has through its low volume strategy is avoiding the devaluation of its inventory over time. In the technology industry, depreciation of assets (and thus, raw materials) occurs quickly. According to some accounting standards, depreciation of technological assets takes five years. Translating this into a realized value today, an asset loses nearly 1.7% of its value every month. This means that Dell's competitors like Hewlett-Packard, which holds \$5.2 billion dollars of inventory each year, are essentially losing roughly 20% of their inventory values each year. That loss equates to over \$1 billion per year in inventory value. By holding only six days of inventory a company that size would only lose \$120 million. That is \$880 million lost due to the devaluation of the inventory value.

By utilizing a low inventory strategy, Dell is able to see a huge savings that can be directly added to the profitability of the company. At this point, none of Dell's major competitors can match these savings, due to their different business models.

2.5.4 Service and Support

Unlike most major computer manufacturers, Dell's strategy is not "service at any cost". Dell provides a wide range of service levels- from individual personalized service

¹ HP Financial Statements 2001. Inventory Turnover = Inventory/Total Net Revenues
http://www.hp.com/hpinfo/investor/financials/annual/2001/text_only_10k.pdf

² HP Financial Statements 2001.
http://www.hp.com/hpinfo/investor/financials/annual/2001/text_only_10k.pdf

and onsite consulting to assistance in disposal of old computer systems. However, unlike their competitors that provide “full service” to their customers for no additional costs, Dell offers customized and cost effective service levels that are selected by the customers. With larger company contracts in which a high volume of computers are purchased. Dell provides onsite customer support for free. Unlike for the individual computer purchaser, the high volume of computers being purchased from Dell offsets the cost of that onsite service. Companies are also more prone to require support for their computer systems with their contracts.

However, unlike the money made on a high volume company’s purchase, an individual computer buyer’s purchase cannot offset the cost for providing customer service to that single person around the clock. Purchasers with little computer experience tend to call frequently, and the cost of supporting these users is high. Dell would actually lose money if such service were freely provided. Instead, Dell requires its single computer purchasers to buy customer support for their systems as well. This ensures that those that need service have access to it, and those that do not need or wish to purchase customer service will not be required to pay for it. This strategy actually transforms service for individual purchasers from being a cost center to a profit center for Dell.

2.5.5 Time to Volume

Within every Dell product family line, new designs that include chassis changes with new component assemblies are released every six months to a year. Time to Volume, as defined at Dell, is the time it takes Dell to get its new products ramped up to high volume production. This time to volume varies by the product itself, and according to the availability of parts. Dell has explicitly made reduction of time to volume part of its strategy. The quicker Dell can release new products into production, the higher in value these new products are to Dell. Given the short lifecycles of technology products, the sooner new products are released, the longer that product’s life can be profitable. This is significant, because Dell internally estimates that nearly 60% of the profit that a product makes is within its first six months after being released.

To accomplish this rapid time to volume, Dell partners with leading suppliers to ensure quick release of its products. To reduce research and development costs, Dell does not invent components, but relies on suppliers to create them. In return, Dell

provides open access to customer feedback on products and service issues Dell's suppliers receive real time feedback of issues if they occur.

2.6 TMC Manufacturing Environment

Dell's main strategy is that each computer is custom-made for the customer. Once an order is placed, Dell downloads the order and ships the built computer within three to five business days. Generally, when orders reach the manufacturing floor, they can be pushed out in less than eight hours time. New orders come into the manufacturing facility each day and can be started on the floor within one day of when the orders are taken. For this reason, forecasting of production volumes from day to day is extremely difficult. Seasonality can be predicted, but short bursts in volumes can cause problems on the production floor. Factory planners have less than one shift to predict production volumes or adjust for any anticipated volume bursts for the next shift.

One impact of daily demand shifts is that headcount management is fairly difficult. Enough headcount needs to be present to meet unanticipated higher volumes, but if too many employees are scheduled to work and demand volume is low, a lot of capital can be wasted on salaries. To handle this problem, Dell uses temporary employees to staff much of their production line. If the headcount need changes substantially from day to day, Dell can adjust the amount of people that they bring in each day to meet the staffing level needed. Additionally, adjustments to shift length can be made to ensure that production volumes are met. While these solutions allow Dell's headcount to be flexible, these options have negative impacts as well. By using temporary headcount that potentially changes each day, continuity and predictability in each line suffers. Also, as extra hours are required of employees to meet demand, there is some impact to worker morale, which has implications on quality, safety, or retention of employees.

Another issue with reactive business conditions is that engineers and managers spend much of their time reacting to problems, instead of planning and creating innovative solutions. The daily computations required to predict headcount needs for the next day or shift take time. Instead, that time could be better spent solving problems on the factory floor, such as mechanical and IT downtime that will help keep the manufacturing process in control.

3 ANALYZING DELL'S TMC MANUFACTURING PROCESS

The manufacturing process can be broken down into four main areas – Kitting, Build/Burn, Boxing and Distribution. This process is graphically depicted below.

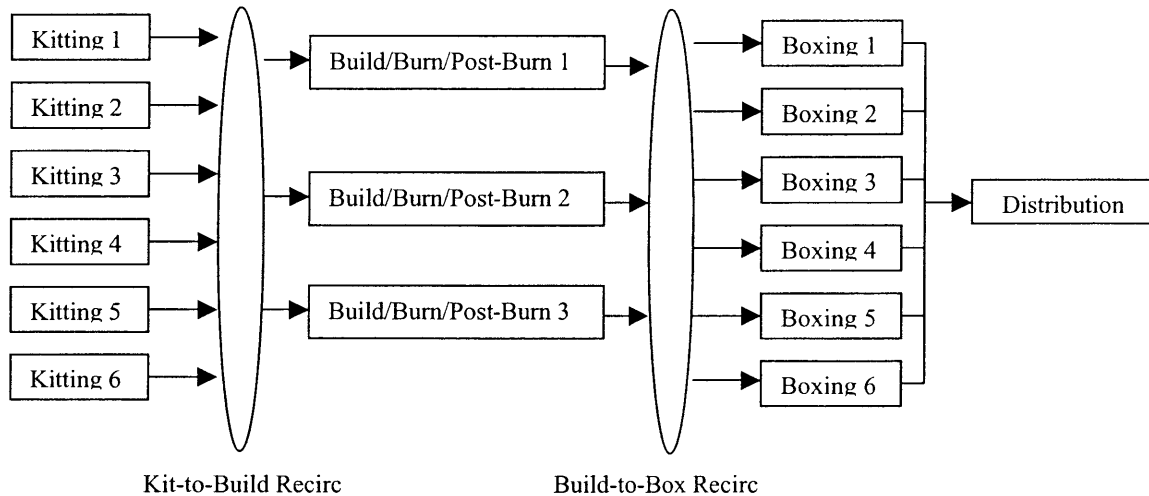


Figure 2: Manufacturing Process

Figure 2 shows the manufacturing process flow for the TMC factory at the time of the author's internship. As product moves through the factory, it travels sequentially through four main areas within the factory – Kitting, Build, Boxing, and Distribution. Each of these areas is described in detail below.

3.1 Process Description

3.1.1 Kitting

There are six kitting lines in the TMC factory. Every kitting line is nearly identical, and the product running down each line is similar as well. When an order comes in, the following process steps are completed within the kitting area.

The first step within the Kitting area is some verification of the order itself, which is done to make sure that the build is not an incompatible combination of parts. The associate on the floor scans the orders by barcode. The inventory management system, at that time, verifies the order and if there is an incompatible combination of parts, signaling the computer can not be built, the system shows a red screen. At this point, the order is held until the planning group dispositions the order correctly. After the order has been

verified, bar code stickers identifying the computer system are placed in totes, which are boxes that are used to transfer the computer and its parts throughout the factory. The bar code stickers are used to track the computer and its parts through the factory.

Once coded, the tote is sent down the kitting line and floor associates at five subsequent stations pick the parts that will be used for assembly in the computer. At each station, lights above the part bins signal what parts are needed to build that specific computer are automatically lit to let the associate know what parts they should pick and put into the tote. After the parts are pulled at each station, it is moved down the kitting line to the next station where the process is repeated until all parts needed for that computer to be assembled have been picked. At the end of the line, the tote containing the picked hardware is paired up with a chassis before being sent to the Build area. There are additional associates that are responsible for resupplying the raw materials for the kitters on the line to pick.

As seen in Figure 2, after going through the Kitting area, the tote (paired with a chassis) is sent along an overhead conveyer, through the Kitting-to-Build Recirculation conveyer, referred to as a Recirc. The tote will rotate on the Recirc until there is an opening in one of the three Build line queues, where the tote will rest queued until assembly. Normally, the floor is configured to send material from Kitting lines 1 and 2 to Build line 1, Kitting lines 3 and 4 to Build line 2, and Kitting lines 5 and 6 to Build line 3. However, the floor can be reconfigured to allow product from any of the Kitting lines to go to any of the Build lines. All products must move through the Kitting-to-Build Recirc, so it is a crucial mechanism in the factory.

3.1.2 Build

Build is where the actual product is assembled. Once the product has been kitted, its tote has all of the parts needed for final assembly. The next step in the process is that the unassembled computer systems are sent down one of the three Build lines. Each Build line has approximately 50 build cells along it, in which associates assemble the computer systems, the software for the computer is downloaded, and it is electronically tested for assembly and software configuration bugs.

Build currently uses one- and two-person build cells. At the time of the study, the majority of the lines were still using two-person build cells, though, Dell would be moving to solely one-person build cells in the future. In two-person build cells, a pair of

associates build a computer together, each with different roles. One of the associates installs the motherboard, memory, processor and heat sink, while the other associate is in charge of all of the drives (i.e. ZIP drives, floppy drives, disk drives, and hard drives). Both people then install all of the cables. The output target for a two-person build cell is 18 units per hour per cell.

In a one-person build line, each associate is responsible for building the entire computer system. The main benefit of moving to the one-person build cells is more build capacity. Two one-person build cells, each with a goal of 12 units built per hour per cell, use the same amount of floor space as a single two-person build cell. This means the average output would rise from 18 to 24 units built per hour with the same floor area. A secondary benefit of using the one-person build cell is that each individual is wholly accountable for both the quality and speed of the build. With two people building a computer, it is hard to identify associates that might be building computers incorrectly. With individual tracking, personal accountability can be enforced. However, the major downside of the one-person system is that there are not two sets of eyes looking at the system. Two-person build cells better helps spot inaccuracies in the build. If there are extra defects due to fewer eyes checking the system after it is built, it could slow up the process, because computers built incorrectly need to be stopped, debugged and possibly rebuilt. Another concern with a one-person build cell is that floor associates might get fatigued quicker because they do more individual labor. These will be concerns as Dell moves into one person build cells in the short future.

Reflecting on the two-person cell system, the process for building a computer is as follows. Once the tote with the individual parts arrives at a build cell, the associates in the build cell begin by verifying the parts provided by Kitting are accurate by scanning each part's bar code. Missing or extra parts are identified or replaced within the tote from kitting. Once the parts have been verified, the associates place a disk in the floppy drive, which is used to populate information about what is installed on the computer. This will be used later in the process by the burn and test area. After the information is loaded, the associates build the system and then put it on a conveyer to be moved over to burn area. In the burn area, communication cables from the burn servers are connected to the newly assembled computer to allow the Burn servers to download software and test the new computer. The first burn step is a five to six minute diagnostic test, which tests for

assembly and component errors. If there are problems, a quick fix can be attempted. For substantial problems, the computer must be sent back to the build cell and rebuilt. If there are three rebuilds on the same system, the system is scrapped and started over back in the Kitting area. After the diagnostic check, all good computers have the system software burned onto their hard drive. This process can take between one hour and 1 hour and 45 minutes. Once complete, an indicator will tell the associate when to disconnect the cables and then the associate will perform a quick wipe down and final verification that the components are all attached. After that, a clean label that shows what type of processor and operating system resides in the system is put on the outside of the box. Once complete, the computer is sent to the Boxing area, via the Build-to-Box Recirc.

Like the Kitting-to-Build Recirc, every product that goes down the production line uses the Build-to-Box Recirc. Also, similar to Kitting-to-Build Recirc, product can be delivered between any of the three build lines to any of the six Boxing lines, however, generally Build line 1 delivers to Boxing line 1 or 2, Build line 2 delivers to Boxing line 3 or 4, and Build line 3 delivers to Boxing line 5 or 6.

3.1.3 Boxing

In the Boxing area, the product is packaged for delivery. The first step in the Boxing process is the computer is scanned and verified. At the same time, a box is folded and taped along the bottom of the box, preparing for the computer for packaging. Once the computer is verified, the computer is picked up with vacuum hoist and foam ends are placed on each side of the computer. The computer with the foam ends is then put into a box and put on a conveyer line similar to the Kitting line. Here, the box moves down the line from station to station, and associates pick parts, such as a keyboard, a mouse, documentation packets, and extra parts that were specially requested by the customer, and put those parts in the box with the computer. At each station, the picked parts are scanned as they are put in the box to confirm correctness. At the end of the conveyer line, the boxes are automatically closed and taped. If there is a part missing that prevents the Boxing from being shipped, the computer is pulled aside and tagged with an indicator of the shortage. These boxes are held until the missing part is found. One of the lines was fully automated at the time of the study, requiring no people, and one line required no human interaction to put foam on the computer and place it in the box. Once the

computer systems are complete, they are put on a conveyer that takes them to distribution.

3.1.4 Distribution

After boxing, the packaged systems are scanned and the material tracking system shows that the box is traveling into the distribution area. A second scan point is used to check the weight of the box and also verify that the computer has reached the distribution conveyer system. Once the box with the computer has been weighed it is matched up with the order's Speakers, Printers and Monitors, or SPAM, as it is known at Dell. All the computers and SPAM are routed through the Shipping Mezzanine where orders are automatically sorted, shipping labels are attached, and package slips such as invoice notice are attached as well.

The sorted orders can go to one of three areas – the “Less than a Truck Load” (LTL) area, the Parcel area, and the “Reject” area. The “Less than Truck Load” (LTL) is where bulk order of product can be loaded by pallette onto the trucks for shipping. For orders greater than three computers, the product initially is stored in large storage racks until the last computer in the order has cleared boxing, at which time the order is released from storage (automatically) and are moved from storage to conveyer to LTL docks where they assembled onto pallettes and are loaded on trucks for delivery. For delivery for orders of one to three computers, the computers will be sent to a different distribution area, call Parcel. Here, small orders are sent by conveyer system to docks for UPS or FedEx to load on their trucks and deliver. This area requires individual manual loading of SPAM. Orders greater than one are sent to storage racks until the full order has been built and assembled. Once assembled, the order gets sent to the docks for shipping. For orders of one computer, the computer and SPAM can be sent to Parcel docks directly. The third area is the “Reject Lanes.” This area is used for individual goods that are going to international sites or require special processing, and each product here must all be manually handled. The speed of the distribution area is relatively flat given the process is almost completely automated. Its throughput is not tracked like the other areas in the factory for that reason.

3.2 Finding the True Factory Constraint

Familiar now with the manufacturing process, the first key step in building an optimization is identifying the constraint (or bottleneck) of the factory. A bottleneck is defined as any process that impedes the flow of work. The slowest area in a factory is considered the bottleneck, or the capacity constraint of the factory. This could be a hard constraint, such as conveyer equipment, or could be a constraint due to insufficient capacity within one of the work areas. The Theory of Constraints states that the factory will produce at a speed equal to that of the slowest area in the factory. Areas that move quicker than the constraint and lie before the constraint in the process, will build queues of material for processing in front of the constraint area. If an area lies down the production line from the constraint, they will only be fed material to work on at a rate equivalent to the constraints production rate. Understanding where the bottleneck is in the factory will be useful in determining how to load each of the areas in order to maintain the constraint in the factory. Therefore, when looking at optimizing headcount for the factory, it is important for the model to fit into the current factory design. It should staff to a level in which the constraint area will meet the demand, and all other areas ensure that the constraint area is always working.

3.2.1 How do you find a Constraint?

Generally, there are some easy ways to try to identify the constraint in a factory. Ultimately, an associate on the floor could identify a constraint by looking at where material is built up in the factory. If material is built up and waiting to be completed by a specific area, that area might be the factory bottleneck. In the case of this factory, there is usually inventory built up before the build area. However, there may be more than one bottleneck in the factory.

Another approach used to find a constraint is by talking with the supervisors that monitor the pulse of the factory every day. In the case of TMC, supervisors were asked to identify the constraint in the factory, but there was some disagreement about where the bottleneck was. The Kitting area supervisor thought the bottleneck was in Build, the Build area supervisor thought the bottleneck was in Boxing, the Boxing area supervisor pointed at Build or Kitting and the Distribution supervisor pointed at Boxing or Build. Because there was some inconsistency in the perception of where the constraint was located, it would imply that the bottleneck moves or is not known in the factory.

Another approach to determine where the bottleneck is would be to talk to the factory process designers about where they designed the constraint to be in the factory. When talking to the manager of the Process Control group, it was found that the bottleneck was originally designed to be in Build. However, over time, process improvement projects that have occurred in each area and the general feel was that the bottleneck of the factory may have shifted into another area. As productivity enhancements were made, old headcount forecasts may have become obsolete. Now, some areas may be overstaffed, while others are understaffed. This may cause the bottleneck in the factory to move from where it was designed. Talking with the different groups may give an overall look at the factory and help to determine where the constraint lies. However, despite these observations, to really identify the constraint, production data is key.

In order to analyze the manufacturing process at TMC, the constraints and capacities of the different areas in the factory were determined. Initially, the idea was to find out if the factory was hitting a bottleneck in the factory that could not be increased by adding headcount to it. For example, the Recircs are conveyer systems with a maximum capacity. If the capacity of the factory matches the Recirc capacity, the Recirc is likely the bottleneck of the factory. Maximum capacities that are not headcount related are referred to in this thesis as hard constraints. Hard constraints tell the maximum capacity that can be achieved in the factory. At the time of the analysis, production output for the factory had a maximum output of just over 2000 units per hour. After determining if the hard constraints impact throughput for the factory, focus can be shifted to determining the capacity of each area as it relates to headcount.

3.3 Finding the Factory Hard Constraints

As mentioned, the first step to building a headcount model is analysis of the factory. Each area was looked at and production capacity was calculated. The key areas that were considered hard constraints on our factory floor are the Kit-to-Build Recirc, the Build-to-Box Recirc, and the Burn process. While there are other hard constraints, these represented the four processes that were stand-alone entities with major impacts on factory performance. Their analysis follows.

3.3.1 Kitting-to-Build Recirc Capacity

To best estimate the Kitting-to-Build Recirc capacity, the factory was assumed to be implementing 100% “waterfalling,” meaning any kitting line can send material to any build line. Finding a time in which the Recirc was full and running at max capacity was hard to do, as some needed information was not accurately reported. So, the capacity of the Kitting-to-Build Recirc needed to be calculated another way. Talking with the engineer in the area, capacity could be calculated by knowing the speed of the conveyer, the size of the box, and the space between the boxes. This calculation follows:

$$(1) \quad \text{RecircMaxCapacity (RMC)} \quad = \text{ConveyerSpeed} / (\text{ToteWidth} + \text{SpaceBetweenTotes})$$

Using equation 1, above, and the following data we can calculate the number Recirc’s maximum capacity.

ConveyerSpeed	= 230 ft/min
ToteWidth	= 2 ft/unit
SpaceBetweenTotes (SBT)	= 2ft/unit
RecircMaxCapacity (RMC)	= 230/(2 ft/unit tote + 2 ft/unit SBT)
	= 57.4 units/min
	= 3450 units/hour

Initial reaction to the above analysis is that there is sufficient capacity in the conveyer system to meet the current 2000 unit demand that the factory could potentially see. However, there are two other factors to determine maximum Recirc capacity. First, the amount of time that the Recirc is stopped (due to downtime, etc.) must be accounted for. The impact of this can be estimated using the following formula:

$$(2) \quad \text{Average Downtime Impact} \quad = \text{DowntimePercentage} * \text{RMC}$$

Additionally, you would need to know the amount of capacity in totes that is lost because the totes are not scanned properly and sent to a build line, so they need to be

circulated and rescanned again. The Rescan Impact is two times the count of output that does not move to the Build area. The impact includes one for the tote itself because it could be not scanned properly or could not be pushed to the Build area for a reason other than the Build area being full of totes. The second tote impact is due to a new tote not being able to be pushed onto the Recirc, and therefore maximum capacity is reduced. With these estimates known, the max capacity of the Recirc can be calculated.

$$(3) \quad \text{RescanImpact (RI)} \quad = \text{TotesRescanned} * 2$$

With estimates for the downtime and rescan impact the following formula could be used to estimate the capacity of the Kitting-to-Build Recirc.

$$(4) \quad \text{TheoreticalMaxCapacity} \quad = 3450 - \text{DowntimePercentage} * 3450 \\ - 2 * \text{TotesRescanned}$$

However, TMC at the time of this analysis did not have good downtime and rescan data to approximate the data from. Instead, to collect this data, the process engineer in this area did a time study in the factory in which the Recirc was filled to capacity and units were counted manually. Downtime and Recircs were manually tabulated. While a good approximation, the challenge was keeping the Recirc fed to its max capacity without being starved or blocked. In that time study, the Estimated Max Capacity actually was estimated to be 2600 Units per Hour. The combined downtime and rescan impact could be estimated as below:

$$(5) \quad \text{Downtime/Rescan Impact} \quad = 1 - \text{ActualOutput/RecircMaxCapacity} \\ = 1 - 2600/3450 \\ = 0.246 \\ = 24.6\%$$

This means that downtime and rescan reduced the Recirc capacity by 24.6%. Finally, it is worth noting that in the last months, an upgrade allowed the units to be

spaced only six inches apart. Using Equation 1, and the following information, the new theoretical maximum throughput is reflected below:

$$\begin{aligned}
 \text{SpaceBetweenTotes (SBT)} &= 0.5 \text{ ft/unit} \\
 \text{RecircMaxCapacity (RMC)} &= 230 / (2 \text{ ft/unit tote} + 0.5 \text{ ft/unit SBK}) \\
 &= 92 \text{ units/min} \\
 &= 5520 \text{ units/hour}
 \end{aligned}$$

A new theoretical maximum capacity could be calculated if the downtime and number of tote rescans were known. If so, the equation would be shown as follows:

$$\begin{aligned}
 (6) \quad \text{TheoreticalMaxCapacity} &= 5520 - \text{DowntimePercentage} * 5520 \\
 &\quad - 2 * \text{TotesRescanned}
 \end{aligned}$$

Unfortunately, because the rescans and downtime are not known, we could approximate the capacity of the Kitting-to-Build Recirc, by applying the same approximate downtime and rescan impact as with our old system. If the theoretical max capacity was reduced by 24.6%, the capacity for the Kitting-to-Build Recirc will be 4160 units per hour. That is well above the 2000 Units per Hour of our current manufacturing system. Therefore, it is not a bottleneck for our factory.

3.3.2 Build-to-Boxing Recirc Capacity

The capacity of the Recirc could not be accurately estimated given production data, however, the technique used to estimate the Kitting-to-Build Recirc capacity could once again be used for this area. Again, the assumption is that the factory is implementing 100% “waterfalling,” or that any build line can send material to any boxing line. As in the Kitting-to-Build Recirc calculations, the capacity of the Build-to-Boxing Recirc could be calculated. Using engineering estimates of conveyer speeds, tote size and the spacing of the totes on the conveyer system, capacity can be calculated using Equation 1 below:

ConveyerSpeed	= 150 ft/min
ToteWidth	= 2 ft/unit
SpaceBetweenTotes (SBT)	= 2ft/unit
RecircMaxCapacity (RMC)	= 150/(2 ft/unit tote + 2 ft/unit SBK)
RecircMaxCapacity (RMC)	= 37.5 units/min
	= 2250 units/hour

With that as the baseline Recirc maximum capacity, the downtime and number of totes rescanned needed to be computed to find the Kitting-to-Build Recirc total capacity. The equation to calculate the maximum capacity is shown below:

$$(7) \quad \text{TheoreticalMaxCapacity} = 2250 - \text{DowntimePercentage} * 2250 - \text{Totes Rescanned} * 2$$

For the above calculation, the downtime and number of totes rescanned are not known. Engineering studies have estimated the average capacity of the Recirc at 2100. While this is above the maximum output within the TMC factory at the time of the study, it is close to it, which means it may become a constraint as the factory grows and ramps up production volumes.

3.3.3 Burn Capacity

Burn capacity, is the physical capacity of the Burn cells in the Build area. When looking at Burn Capacity, a direct capacity calculation could not be performed due to a minimal amount of data available. However, capacity could be estimated by looking at average Burn times and total capacity in the Build areas. Within Build, there are three lines as mentioned, and each line has 24 different quads where the computers are built and burned. Within each quad, there are 46 burn slots that individually burn software and test the new computer systems. Monitors within Burn record the pre-test dwell time, test time, and post-test dwell time of each computer. Pre-test dwell time is defined as the amount of time a computer sits in a burn station waiting to start testing. Post-test dwell time is defined as the amount of time the computer sits within a burn cell after the burn sequence has completed. Over the three months of the study, these times were looked at for each computer. It found that the average pre-test dwell, test time, and post-test dwell

per unit were 15 minutes, 51.5 minutes, and 28.2 minutes respectively. That was a total average Burn time of 94.7 minutes as shown in the following Figure.

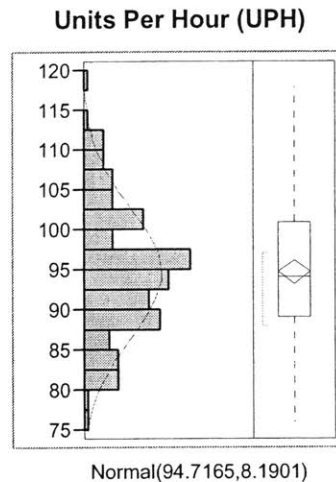


Figure 3: Total Burn Time Average per Unit

The Figure shows that the total Burn time is Normally distributed, with an average burn time of 95 minutes and a standard deviation of 8 minutes. The box plot shows that 50% of the total Burn time data falls between roughly 90 and 100 Units per Hour. Given the total time for each unit through Burn, we can calculate the total number of burn slots and the capacity measured in Units per Hour (UPH) for Burn, by using the following formulas.

$$(8) \quad \text{TotalBurnSlots} = \text{BurnSlots} * \text{BuildQuads} * \text{BuildLines}$$

$$\begin{aligned} \text{Total Burn Slots} &= 46 \text{ slots} * 24 \text{ quads} * 3 \text{ lines} \\ &= 3312 \text{ slots} \end{aligned}$$

$$(9) \quad \text{UPH Estimate} = \text{Total Burn Slots} * 60 / \text{Total Burn Time}$$

This estimate was used to calculate the UPH in Burn for each hour over the timeframe of the internship. The following Figure shows the output Units per Hour for the Burn area.

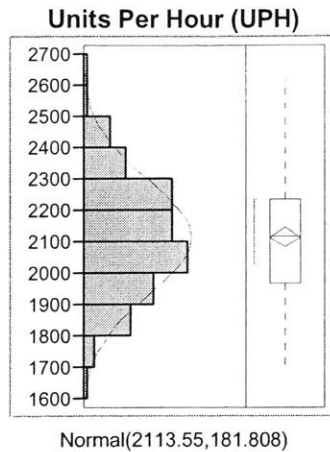


Figure 4: Distribution of Burn Units per Hour

The Burn UPH is a Normal distribution with average capacity of 2114 UPH and a standard deviation of 182 UPH, which leads us to believe that there is sufficient capacity at Burn.

To verify this, Burn capacity was compared to the capacity of the Build area for the same shift in two ways. The graph on the left shows the distribution of shift Burn capacity minus the Build area's total output for each shift. The graph on the right shows the average Burn hourly capacity minus Build hourly average output during the same shift.

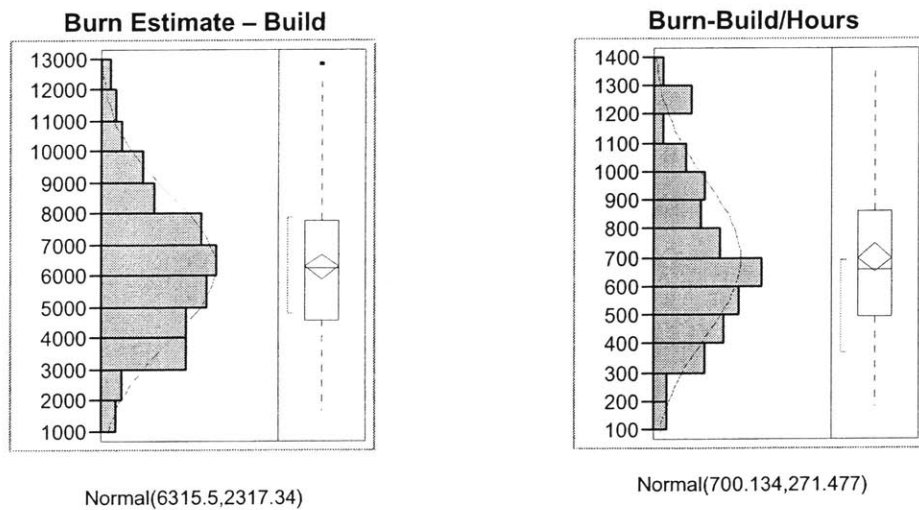


Figure 5: Burn Excess Capacity Over Build by Shift and by Hour

In Figure 5, the graph on the left shows that by totaling the capacity in Burn for each shift and subtracting the Build total from the Burn capacity there was an average shift surplus of burn capacity that averaged 6315.5 units with a standard deviation of 2317.34. This reiterates that we have sufficient capacity to not constrain Build. The graph on the right shows that same data as an hourly average, Burn capacity was greater than Build average hourly output in all cases tested, with a minimum excess capacity of nearly 181 units per hour worked. This Burn capacity data again shows that Burn would generally not be considered to be constraining Build or the factory for that matter.

3.3.4 Hard Constraints Analysis Results

The results of measuring the hard constraints show that none of the three hard constraints were low enough that they could be considered the constraint for the factory. However, Build-to-Boxing Recirc may be near the maximum capacity for the factory, and may become an issue as the factory grows and output volumes increase. With maximum factory output at the time of the study at roughly 2000 units per hour, because each of the hard capacities were sufficiently above this maximum historical output level, the constraint of the factory is likely not due to these hard constraints, but instead due to headcount capacity constraints found in one of the areas in the factory.

The hard constraint calculations were good estimates, however, better estimates could be found with a few extra data items. First, in the Estimated Max Capacity for the Recirc capacities, both downtime and tote rescans could be measured by writing reports that query the material tracking database currently in place in the factory. However, those reports have not been generated, and therefore the data is too cumbersome to calculate. For the Burn cell calculations, some data collection could improve capacity tracking as well. First, the cell total is based on the total amount of cells in the factory, not necessarily cells that were available each day. While close to the same number, the availability number will have some impact on final capacity calculations. Also, the capacity measurement in the Burn area is based on shift totals averaged to units per hour. While hourly numbers are tracked, there are no current reporting methods to access that data at this time. Better reports will allow better tracking to be done and allow capacity variation to be tracked as well.

3.4 Analyze Production Capacity in Each Area

Since the hard capacities were determined to not be the bottleneck within the factory, the production capacity can be looked at in each area to determine which area is the constraint. Within this thesis, production capacities are defined as those in the factory that can be controlled through changing the conditions inside the factory. To change the capacity, the area factors such as headcount could be changed which would in turn raise or lower the output in the area. To change the constraint in the factory, the factors such as headcount could be changed in each area simultaneously, which would in turn raise or lower the capacity in every area simultaneously. The constraint becomes the area with the lowest capacity overall, and the constraint could change with time. This was verified by looking at Block and Starve analysis in the factory. Here, the constraint can be found by looking at how often each area is blocked by an area down line, or starved for material from the area before it. An area is blocked when an area cannot move product out of their area into the next area because the area down the line is full and cannot accept new material. Likewise, an area is starved when that area is not able to work on product because it is not being sufficiently supplied with enough material to keep that area busy.

Using the data from internal inventory tracking databases, every minute of block and starve time in the factory had been recorded for the last year. With that data, I analyzed the amount of block and starve time in each of the areas. The Kitting measurements were taken pre-kitting for Starve time, and right after kitting for the Block time. Build was measured before and after each of the Build lines. Boxing Block and Starve was measured after push-out from Boxing and before the Boxing vacuum hoists respectively. It is important to say that for this discussion we have assumed that the hard constraints in the factory are not in themselves a bottleneck, but instead the bottleneck is caused by productivity of the headcount in the Kitting, Build or Boxing area. The results of the analysis on block and starved time for each area follow:

AREA	BLOCK TIME (Avg, Std Dev)	STARVE TIME (Avg, Std Dev)
<i>Kitting</i>	Not Reliable	(0.007, 0.031)
<i>Build</i>	(0.232,0.238)	(0.112,0.212)
<i>Boxing</i>	(0.039,0.045)	(0.321,0.171)

Table 3: Block Starve Analysis

While the Kitting block data is a bit unreliable, the amount of time that each of these areas is the constraint can be estimated. The block time in Table 3 above shows the mean percentage and standard deviation of each hour that each area is blocked by the next area in the production process. The starve time represents the amount of time (mean and standard deviation) that each area is starved of material from the preceding area. To estimate the amount of time that Kitting is the bottleneck the following equation can be used.

(10) $KittingConstraintPercent = BuildStarvedPercent - KittingStarvedPercent$

Looking at the data from Table 1, Kitting is starved for materials on average of 0.7% of each hour, and Kitting starves Build 11.2% of the time. Using this information, we can use the above equation to estimate the percentage of time that Kitting is the constraint.

$$\begin{aligned}
 KittingConstraintPercent &= 11.2\% - 0.7\% \\
 &= 10.5\%
 \end{aligned}$$

Likewise, the percent of time that Build is the constraint can be approximated, as well, using the following equation.

(11) $BuildConstraintPercent = BoxingStarvedPercent - BuildStarvedPercent$

From the data in Table 3, Boxing is starved of material an average of 32.1% of the day, and Build is starved of material an average of 11.2% of each day. With that

information the percentage of each day that the Build area is the constraint can be estimated.

$$\begin{aligned}\text{BuildConstraintPercent} &= 32.1\% - 11.2\% \\ &= 20.9\%\end{aligned}$$

Finally, the amount of time that the Boxing area is the constraint can be determined as well. The equation to estimate the amount of time that Boxing is the constraint is shown below.

$$(12) \quad \text{BoxingConstraintPercent} = \text{BuildBlockedPercent} - \text{DistribBlockedPercent}$$

From Table 3, it can be seen that the Distribution area blocks the Boxing area only 3.9% of the time. Table 3 also shows that the Build area is constrained by the Boxing area 23.2% of the time. Using this information, the percentage of time that the Boxing area is the constraint can be calculated.

$$\begin{aligned}\text{BoxingConstraintPercent} &= 23.2\% - 3.9\% \\ &= 19.3\%\end{aligned}$$

Noticing that each of the above areas are a constraint within the factory supports the theory that the bottleneck is at this point unpredictable and can move between areas throughout a day, however, most of the time, it resides in Build. This also supports the idea that the shift could be due to headcount fluctuations in the factory.

To understand how shifts in headcount can impact productivity, a model for production capacity was created for each area, based on various factors on the factory floor. There are three key benefits to creation of these production models. First, the production models should help determine which factors are important in predicting production volumes and which factors are not. For those factors that are important, the analysis should point out to what extent that factor impacts production and if there are trends that we should be aware of. Secondly, by running the analysis, it should help measure how closely output in each factory area can be predicted given the production

factors. Finally, the analysis should characterize the amount of variability that exists between the predicted output and the actual output of the factory. By measuring the variability, the range of possible output given set factors can be known, as well as the probability of meeting certain output level given the different factor settings. For example, given a set headcount, the range of output possible with the headcount can be determined, and the probability of achieving an output goal given that headcount level can be determined.

To build the production model, a multivariate regression analysis that predicts the maximum output for the three main production areas, Kitting, Build and Boxing were run using a standard statistics package. The results for each area are discussed below. Additionally, some analysis was done in Distribution that will be addressed as well. Data was clustered by shift for the convenience of the managers in charge of each shift.

3.4.1 Kitting Capacity

The first area that was analyzed was the Kitting area. The factors that focused on in Kitting included Hour of Day Worked, Day of the Week, and a factor called Labor Hours. Labor Hours was calculated by multiplying the percentage of each hour worked times by the number of temporary and permanent employees that were working that hour. The equation for Labor Hours can be seen below.

$$(13) \quad \text{KittingLaborHours} = \text{PercentHoursWorked} * \text{KittingHeadcountTotal}$$

Initially, regression analysis was performed measuring hours worked and headcount separately, but the best regression analysis occurred when these factors were combined into the one factor, labor hours, as described above. For kitting, in order to account for economies of scale, two factors including Labor Hours Squared and Labor Hours Cubed were also included. The premise was that as the number of hours of labor increase, the increase in productivity actually increased at a polynomial rate instead of linearly. The results confirm this assumption as the model was a better predictor as the squared and cubed factors were included. Each of these factors were regressed against production output volume, which was defined as the number of filled totes that leave the area with parts ready for assembly in the build area. Production output was summarized by hour, so the hourly output rate for the Kitting area could be determined. It was

important to choose a time frame in which some intra-day variation could be measured, but not too granular that the data was meaningless. Shift totals would not reflect different output rates from the other areas because the factories each move as fast as the slowest area. For less than an hour, the regression would contain too many parameters and would not be meaningful.

One other factor that was looked into was the amount of block and starve time for the kitting area. Block and starve time did not prove to be a good factor to help predict production levels in the kitting area. To measure block time, there is a photo-eye sensor that sits directly after each Kitting Line. To be considered block time, the conveyers need to be filled completely, all the way to the photo-eye sensor. When the photo-eye sensor sees a tote sitting above it, the line is considered blocked, and the total blocked time will be recorded in the database. The reason block and starve might not be a good predictor of output in the Kitting area could be that the associates in the Kitting area actually slow down as it visually sees the buildup of material, and so the conveyers are not backed up to the photo-eye. The workers are working less efficiently than they should, however, the photo-eye does not show that the Kitting area is blocked. Regardless of block and starve time's usefulness in this model, it is something that could be used to create a good prediction model in the future.

3.4.1.1 Parameter Estimates for Kitting

The output of the multivariate regression analysis will give you a combination of each factor, referred to as a Term, a measure of its importance to the prediction model (given by a P-value), and a multiplier, called the Estimate, that is used to measure the impact of each Term. The following table contains the Parameter Estimates by factor for the Kitting area.

Parameter Estimates				
Term	Estimate	Std Error	t Ratio	P-Value
Intercept	-2477.419	1150.806	-2.15	0.0321
Hour[00]	209.97356	53.4979	3.92	0.0001
Hour[01]	65.359689	69.82185	0.94	0.3500
Hour[02]	-600.7254	128.3333	-4.68	<.0001
Hour[05]	-21.93978	59.73281	-0.37	0.7137
Hour[06]	269.59273	47.19971	5.71	<.0001
Hour[07]	8.1870055	43.55285	0.19	0.8510
Hour[08]	86.295521	44.87439	1.92	0.0554
Hour[09]	330.52775	47.38304	6.98	<.0001
Hour[10]	237.5415	44.83703	5.30	<.0001
Hour[11]	-613.4656	44.71202	-13.72	<.0001
Hour[12]	-253.5107	44.76367	-5.66	<.0001
Hour[13]	252.95222	47.19808	5.36	<.0001
Hour[14]	10.430631	54.59729	0.19	0.8486
Hour[15]	-378.224	187.7608	-2.01	0.0448
Hour[16]	-124.8497	44.73264	-2.79	0.0056
Hour[17]	375.5987	45.03964	8.34	<.0001
Hour[18]	-142.6235	45.03964	-3.17	0.0017
Hour[19]	-625.4569	45.03964	-13.89	<.0001
Hour[20]	408.5987	45.03964	9.07	<.0001
Hour[21]	365.32092	45.03964	8.11	<.0001
Hour[22]	-78.34575	45.03964	-1.74	0.0830
Day of the Week[FRI]	-39.0757	26.31688	-1.48	0.1386
Day of the Week[MON]	24.860837	24.04191	1.03	0.3019
Day of the Week[SAT]	-76.47103	61.24114	-1.25	0.2127
Day of the Week[THU]	9.8015916	25.54663	0.38	0.7015
Day of the Week[TUE]	28.396449	23.3257	1.22	0.2244
Labor Hours Squared	-1.57805	0.712395	-2.22	0.0275
Labor Hours Cubed	0.0067708	0.003132	2.16	0.0314
Labor Hours	129.68356	51.39143	2.52	0.0121

Table 4: Kitting Regression Equation Estimate

There are a couple of important pieces of information that can be focused on from this information. First, the terms listed along the left column are the factors that have been used to predict output in the factory. In the above parameter estimates, there are three types of Terms listed.

First, the Intercept Term is the value of where the prediction equation would cross the y-axis if all other factors had a value of zero. The second type of Term is a continuous variable term. These terms can be any numerical value. In the Parameter Estimates shown above, the terms Labor Hours, Labor Hours Cubed, and Labor Hours Squared are continuous terms, because the amount of Labor Hours worked could be any number, such as 323.25 hours. Finally, the third type of Term is called a categorical, or discrete term. This means that there are predetermined specific values that these Terms could be. In this case, the Hour and Day of the Week terms are both examples of categorical Terms. The values for Day of the Week can only be Monday, Tuesday, Wednesday, Thursday, Friday or Saturday because the factory never ran on Sunday. Likewise, the Hour Term can only be 0 through 23, or midnight through 11pm. Each of

the categories possible for categorical Terms is listed in the brackets next to the Term name. For example, the Hour[0] Term and Estimate displays the Estimate value for when the hour is zero, or midnight. Each hour of the day that TMC actually worked over the three months from which the data was collected is included in this model.

It is important to note that to determine the Estimate, one value is selected as the baseline case, and the effect that each other value has is measured. For the Day of the Week variable, the baseline case would be Wednesday. Its Estimate value is assumed to be zero in this model. Every other day's value is the Estimate of the difference between that day and Wednesday. For this model, Hour[23], or 11pm, is considered the baseline case for Hours. All other Estimates are compared to that hour. For continuous terms, the Estimate is the estimate of the effect of adding 1 more unit of the term to the model.

One other important piece of information given by the regression output is the P-Value column, which is used to measure significance of the factor. The lower the value the more significant that factor is. For the analysis run in the factory, a factor is considered to be significant when the P-Value column contains a value that is less than 0.15. The parameter estimates for kitting allow us to pull out some key information. This Kitting prediction equation shows that with a few exceptions most of the factors we have used are significant indicators of output in the factory. Below are key assumptions that we can make from our data. All the conclusions below are based on what the model shows as significant data points. Additionally, using all of the Terms and Term Estimates can be used to predict the output for the Kitting area given a known number of Labor Hours, Day of the Week, and Hour of the Day. An example of this is shown later in this chapter.

3.4.1.2 Kitting Prediction Equation

Knowing the Terms and Estimates, a Prediction Equation can be created that will estimate the average output given the Intercept, Day of the Week (DOW), Hour, and the number of Labor Hours worked in the Kitting area. To find the average output in the Kitting area, the following equation is derived.

$$\begin{aligned}
 (14) \text{ KittingOutput} &= \text{KittingInterceptEstimate} \\
 &+ \text{KittingDOWEstimate} \\
 &+ \text{KittingHourEstimate} + \text{KittingLaborEstimate}
 \end{aligned}$$

For the above Equation, the InterceptEstimate term can be found in the Parameter Estimates, and is equal to -2477.419 . Additionally, the DayofTheWeekEstimate and HourEstimate can be found by matching the day of the week and hour you are estimating with the Day of the Week and Hour terms that matches them. Finally, by plugging the number of labor hours (as computed in Equation 13) into the following equation, the LaborEstimate can be solved.

$$(15) \quad \text{KittingLaborEstimate} = 0.0067708 * \text{KittingLaborHours}^3 \\ - \text{KittingLaborHours}^2 * 1.57805 \\ + \text{KittingLaborHours} * 129.3568$$

Using the above equation, the Kitting area can be estimated. An example using numerical data is given in Section 3.5.

3.4.1.3 Observations from the Data

Looking at each hour in the Kitting parameter estimates, it can be determined those good production hours for the first shift includes the hours of 6am, 9am 10am and 1pm. By looking at the Estimates, we see that these hours generate at least 100 units per hour *greater* output than the model baseline. Poor production hours include the hours of 11am and 12pm. These hours generate at least 100 units per hour less output than the model baseline case. When excluding lunch hour and the first hour of the shift, the general productivity of the workers arguably *decreases* as the shift drags on by about seven units per hour.

A second observation is that good production hours for the second shift include the hours of 5pm, 8pm, 9pm and 12am. These hours generate at least 100 units per hour *greater* output than the model average. Poor production hours for second shift include the hours of 4pm, 6pm, 7pm and 2am. Each of these hours has output volumes that are over 100 units per hour *less* than base case for the overall production model. When excluding lunch hour and the first hour of the shift, the general productivity of the workers arguably *increases* as the shift drags on by about four units per hour.

A third observation is that the day of the week does not generally have a significant effect on the output level of the Kitting area. However, Friday performance is

significant and shows that productivity is generally poor, yielding 39 units less per hour than other our base case, Wednesday. Also, by looking at the predicted Estimates for the each Day of the Week, it can be seen that output reduces by an average of 40 units per day throughout the week.

Each day, there are scheduled breaks in the factory. One big impact that that this shows is that breaks have a severe impact on productivity. While some drop in output is expected during breaks, it can be seen that lunch breaks for each shift, occurring at 11am and 7pm, have additional impact to production output. The hour following lunch in the first shift, 12pm, produces 253 units less than average, and the hour before lunch during second shift, 6pm, produces 143 units less. There is a two-hour window of impact around lunch breaks for each shift.

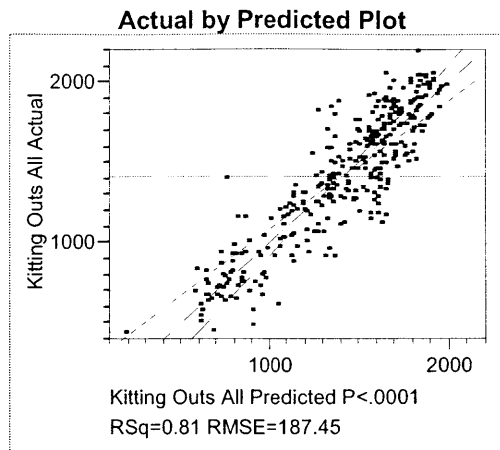
. Finally, by looking at the relationship between labor hours and productivity the model shows that productivity is a polynomial function. An increase in output due to labor hours can be approximated by the equation:

$$\begin{aligned} \text{KittingLaborEstimate} &= 0.0068 * \text{KittingLaborHours}^3 \\ &- 1.58 * \text{KittingLaborHours}^2 \\ &+ 129.68 * \text{KittingLaborHours} \end{aligned}$$

This shows that productivity is not a linear relationship, but one that grows with growth in labor hours invested. While not intuitive, this model suggests that there might be some economies of scale within the kitting area.

3.4.1.4 Measure of Predictability of Kitting

Given the factors and the equation above, the Parameter Estimates was found to be pretty accurate for predicting output against the actual output values. To measure the output, each hour that was analyzed had its actual output tracked against the predicted output given the hour, day, and labor hours that were used during the hour. Below is a plot of the predicted value versus the actual value plotted against each other for all hours analyzed. Additionally, a summary of the model's fit is included.



Summary of Fit

Rsquare	0.8065
RSquare Adj	0.788042
Root Mean Square Error	187.4489
Mean of Response	1416.087
Observations (or Sum Wgts)	334

Figure 6: Kitting Regression Analysis

Ideally, for a perfect prediction, a straight line would be seen in which the actual output values always equal the predicted output values. Under the perfect scenario, the summary of fit would show an R-square and R-square Adjusted values of 1.00. For the Kitting area prediction, there is some variability that exists in the plot of actual values versus the predicted value, seen by scattered appearance of the points. However, a good general trend exists relating our prediction equation to actual values. The measure of fit for this model is summarized by the R-Square and R-Square Adjusted values. In this model, the prediction model accounts for nearly 80% of the variation in production output. That is a decent prediction given the number of factors that we are tracking.

3.4.1.5 Variation between Predicted and Actual Values

One final component that was looked at in the regression model is the residual plots. The residual is the measure of how far the predicted values are off from the actual values. There are two factors to look at. First, it is important to look at the residual values against the actual output volumes using a residual plot. That plot is shown in the Figure below.

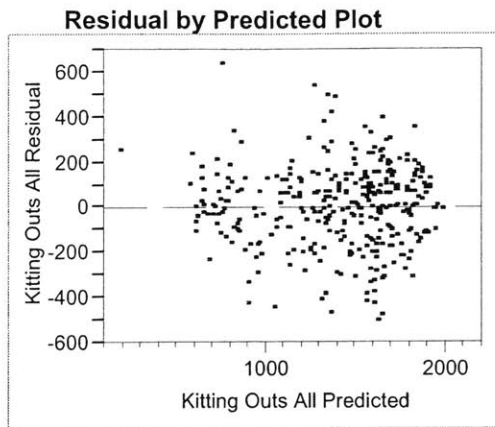
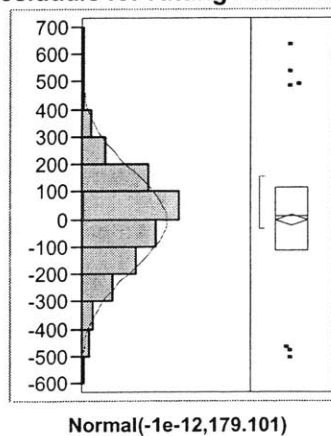


Figure 7: Kitting Residual Analysis

This plot is used to look for trends in the residual plot. If a trend exists, and the plot doesn't appear randomly distributed, the regression could probably be made to be more accurate by trying more advanced regression techniques with the data. With this data, there is slight convexity, however it is not a severe trend in the residual plots, such as U-shapes, etc. This implies that our data points are truly independent, and the regression model can be used.

The second trend to look for with the residuals is if they are well behaved normal probability plots. If the values are truly random, there should be a good normalized distribution for each of the areas. The plot of this data is shown in Figure 8 below.

Residuals for Kitting Distribution



Moments	
Mean	-1.49e-12
Std Dev	179.10086
Std Err Mean	9.7999631
upper 95% Mean	19.277639
Lower 95% Mean	-19.27764
N	334
Goodness-of-Fit Test	
Shapiro-Wilk W Test	
W	Prob<W
0.981398	0.2634

Figure 8: Kitting Residual Distribution and Goodness-of -Fit

These plots are important for showing that the residuals are normally distributed around the predicted output values. The Shapiro-Wilk test measures Normality and measures the percent of variation that is attributed to normal variation of the output. In this case, 98% of the variation of the residuals can be explained through normal variation. For kitting, by looking at the residuals, there is a mean difference from each hour of actual output of zero (mean of $-1e-12$). The measured standard deviation of the model is 179. In general, a model for the kitting area has been generated by using the prediction estimates from Table 2 to solve for a predicted output volume, it can be safely assumed that the model will be off by an amount that can be quantified by a normal distribution around that predicted value by a standard deviation of 179 units per hour. The measure of variability of the Kitting area and the Build area will be important in determining what level of output in the Kitting area is needed to ensure that the Build area is not starved. This will be addressed later in the paper.

3.4.2 Build Capacity

Hourly data from the time period of our study was available for the factors identified in the Area Production Headcount Factors section above. The factors that focused on in Build included Hour of Day Worked, Day of the Week, and a factor called Labor Hours. Labor Hours was again calculated as it was in Kitting found above, by multiplying the headcount by the percent of each hour worked, as shown in the following equation.

$$(16) \quad \text{BuildLaborHours} = \text{PercentHoursWorked} * \text{BuildHeadcountTotal}$$

Once again, while regression analysis was performed measuring hours worked and headcount separately, the best regression analysis occurred when these factors were combined into labor hours.

3.4.2.1 Parameter Estimates for Build

As in kitting above, the output of the multivariate regression analysis will give a combination of each factor, a measure of its importance to the prediction model, and a multiplier that is used to measure the impact of each factor. Below was the parameter estimates for each factor used to model the output level in the Build area.

Parameter Estimates				
Term	Estimate	Std Error	t Ratio	P-Value
Intercept	-2200.792	534.4663	-4.12	<.0001
Day of the Week[FRI]	-11.17126	11.70407	-0.95	0.3401
Day of the Week[MON]	-56.12757	11.61581	-4.83	<.0001
Day of the Week[SAT]	43.422362	20.20481	2.15	0.0319
Day of the Week[THU]	18.503633	11.9323	1.55	0.1213
Day of the Week[TUE]	-6.239416	11.38596	-0.55	0.5838
Hour[00]	130.79715	27.96144	4.68	<.0001
Hour[01]	243.99288	34.75841	7.02	<.0001
Hour[02]	147.13515	75.14358	1.96	0.0506
Hour[03]	-136.7332	108.3637	-1.26	0.2074
Hour[04]	-70.95223	86.61524	-0.82	0.4129
Hour[05]	26.669131	30.14686	0.88	0.3766
Hour[06]	41.312235	23.96619	1.72	0.0851
Hour[07]	-139.6062	23.305	-5.99	<.0001
Hour[08]	56.299763	24.25424	2.32	0.0205
Hour[09]	146.4533	24.49728	5.98	<.0001
Hour[10]	140.17673	24.02719	5.83	<.0001
Hour[11]	-586.5979	24.19114	-24.25	<.0001
Hour[12]	-208.173	23.9738	-8.68	<.0001
Hour[13]	132.79337	24.189	5.49	<.0001
Hour[14]	191.58747	27.55515	6.95	<.0001
Hour[15]	-92.63199	77.05813	-1.20	0.2297
Hour[16]	9.6097112	23.35953	0.41	0.6809
Hour[17]	125.30767	23.62446	5.30	<.0001
Hour[18]	-306.0731	23.21596	-13.18	<.0001
Hour[19]	-461.4484	23.41146	-19.71	<.0001
Hour[20]	192.97865	23.61812	8.17	<.0001
Hour[21]	261.2289	24.07136	10.85	<.0001
Hour[22]	-64.57066	23.99994	-2.69	0.0073
Labor Hours	26.405038	5.574089	4.74	<.0001
Labor Hours Squared	-0.071831	0.01818	-3.95	<.0001
Labor Hours Cubed	0.000072	0.000019	3.81	0.0002

Table 5: Build Regression Equation Estimate

There are again some key observations that can be seen on from the Parameter Estimates. As in Kitting, P-Value is the measure of significance of each factor and the lower the value the more significant that factor is. For the analysis run in the factory, a factor is considered to be significant when the P-Value column contains a value that is

less than 0.15. Also worth noting, the baseline case for the Hour parameter estimates was once again 11pm, and the baseline case for the Day of the Week variable was again Wednesday.

3.4.2.2 *Build Prediction Equation*

Like in the Kitting area, knowing the Terms and Estimates, a Prediction Equation can be created that will estimate the average output given the Intercept, Day of the Week (DOW), Hour, and the number of Labor Hours worked in the Build area. To find the average output in the Kitting area, the following equation is derived.

$$(17) \quad \text{BuildOutput} = \text{BuildInterceptEstimate} + \text{BuildDOWEstimate} + \text{BuildHourEstimate} + \text{BuildLaborEstimate}$$

The InterceptEstimate for the above equation is found to be -2200.792 in the Parameter Estimate table for the Build area. Additionally, as in Kitting, the Day of the Week estimate and the Hour estimate can be found as well, by matching the day and the hour being predicted with the Day of the Week and Hour estimates in the Parameter Estimates table. Finally the LaborHourEstimate term can be determined by calculating the Labor Hours for the Build area using the following equation.

$$(18) \quad \text{BuildLaborEstimate} = 0.000072 * \text{BuildLaborHours}^3 - 0.071831 * \text{BuildLaborHours}^2 + 26.405038 * \text{BuildLaborHours}$$

With the prediction equation calculated, the parameter estimates can be analyzed. Below are some observations that we could make from the Build area model.

3.4.2.3 *Observations from the Data*

For the first shift, good production hours include the hours of 9am, 10am, 1pm, and 2pm. These hours generate at least 100 units per hour *greater* output than the model baseline of 11pm. Poor production hours include the hours of 3am, 7am, 11am, and 12pm. These hours generate at least 100 units per hour *less* output than the model

baseline. When excluding lunch hours, 11 am, and the first hour of the shift, the general productivity of the workers arguably *increases* as the shift goes on by an average of nearly 26.5 units per hour.

For Shift 2, good production hours for Shift 2 include the hours of 5pm, 8pm, 9pm and 12 am. These hours generate at least 100 units per hour *greater* output than the model baseline of 11pm. Poor production hours for Shift 2 include the hours of 4pm, 6pm, 7pm and 2am. Each of these has output volumes that are over 100 units per hour *less* than baseline for the overall production model. When excluding lunch hour, 7pm, and the first hour of the shift, the general productivity of the workers also arguably *increases* as the shift goes on by an average of about 5.5 units per hour.

Looking at the productivity by day, Monday performance is significant, with a P-Value of less than 0.0001, and shows that productivity is generally poor, yielding 56 units less per hour than the baseline case of Wednesday. Saturday is a highly productive day yielding 43 extra units per hour worked. Also, by using these predicted estimates and observing the general trend for the week, it can be noted that output increases by an average of 14 units per day throughout the week.

One big indicator that this Build prediction equation shows is that breaks have a severe impact on productivity. While some drop in output is expected during breaks, the model shows that lunch breaks for each shift have a high impact on productivity. The hour following lunch, 12pm, is not productive, yielding an average of 208 *less* units than the baseline case, and the hour before lunch during shift 2, 6pm, produces an average of 306 *less* units than the baseline as well. This data shows that there is a 2-hour window of impact around lunch breaks for each shift.

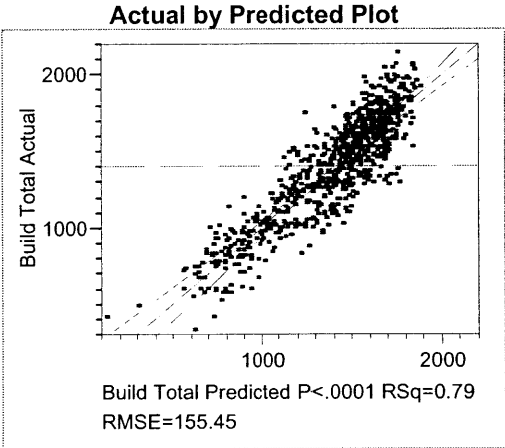
Finally, by looking at the relationship between labor hours and productivity, it can be seen that productivity is a polynomial function. An increase in output due to labor is approximated using Equation 18, as shown below.

$$\begin{aligned} \text{BuildLaborEstimate} &= 0.000072 * \text{BuildLaborHours}^3 \\ &\quad - 0.07 * \text{BuildLaborHours}^2 \\ &\quad + 26.41 * \text{BuildLaborHours} \end{aligned}$$

Unlike a linear relationship where the output growth rate is stable, this model suggests that there is polynomial growth in the Build area, where the output growth rate per labor hour actually increases as the number of labor hours increases. This equation would suggest some economies of scale are being achieved in the Build area, just as in the Kitting area.

3.4.2.4 Measure of Predictability of Build

Below is a plot of the predicted value vs. the actual value plotted against each other for all hours analyzed. Additionally, a summary of the model’s fit is included.



Summary of Fit

Rsquare	0.793109
RSquare Adj	0.785401
Root Mean Square Error	155.4477
Mean of Response	1408.104
Observations (or Sum Wgts)	864

Figure 9: Build Regression Analysis

For our build prediction, there is a good general trend exists relating the prediction equation to actual values. The R-square value suggests that the prediction model accounts for nearly 80% of the variation in production output. Once again, this is a decent prediction for the Build area.

3.4.2.5 Variation between Predicted and Actual Values

To test for independence the residual values were plotted against the actual output volumes. That plot of the residuals is shown below.

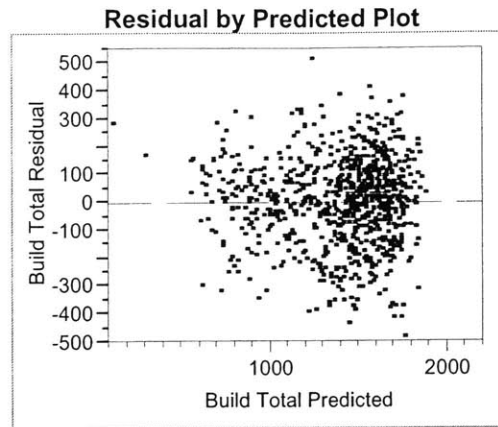
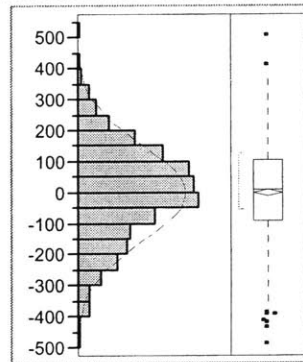


Figure 10: Build Residual Analysis

As with the kitting plots, the residual plot for build has a slight convexity, however, there are not severe trends in the residual plots, such as U-shapes. Additionally, looking at the distribution of the residuals around the predicted value, there is once again a Normal distribution. This distribution chart is shown below.

Residuals for Build Distribution



Normal(-2.01e-13,152.630221)

Moments

Mean	-2.01e-13
Std Dev	152.63022
Std Err Mean	5.1925854
upper 95% Mean	10.191574
lower 95% Mean	-10.19157
N	864

Goodness-of-Fit Test

Shapiro-Wilk W Test	
W	Prob<W
0.982654	0.0654

Figure 11: Build Residual Distribution and Goodness-of -Fit

These plots are important for showing that the residuals are normally distributed around the predicted output values. The Shapiro-Wilk test shows that 98% of the variation of the residuals can be explained through normal variation.

For the Build area, by looking at the residuals it can be seen that a mean difference from each hour of actual output of zero (mean of $-2.01e-13$). The measured standard deviation of the model is 153. In general, a model for the Build area can be calculated by using the prediction equation from Table 3 to solve for a predicted output volume, and the model's residuals are can be quantified by a normal distribution around that predicted value by a standard deviation of 153 units per hour.

3.4.3 Boxing Capacity

Generally, the Boxing regression was not as accurate as the other areas in the factory. While more data was available for the factors identified in the Area Production Headcount Factors section above, the data was much less reliable. The same factors used in Build were used to predict output in Boxing. They include Hour of Day Worked, Day of the Week, and Labor Hours. Labor Hours for Boxing is calculated with the following formula.

$$(19) \quad \mathbf{BoxingLaborHours} = \mathbf{PercentHoursWorked} * \mathbf{BoxingHeadcountTotal}$$

Labor Hours was calculated as it was in Kitting and Build and additional factors of Labor Hours Squared and Labor Hours Cubed were used in the regression to account for productivity gains that might be present do to changes in productivity due to scaling effects in the factory.

The biggest difference between Boxing and the other areas is that Boxing is fed work from Build, which is generally the constraint of the factory. Because of this, boxing will many times be starved of material and unable to work. For that reason, Boxing factors in a measure of block and starve time for the Boxing area was added into the model as well. Block and Starve time for the Boxing area was the measure of time that boxing was blocked by distribution (it could not push out boxes to distribution) or starved of work from Build. Block and Starve data for Boxing was measured using photo-eye sensors at the point immediately before and immediately after the boxing line. While

kitting showed no correlation between productivity and the block and starve time measurements, boxing did show a good correlation. For this reason, block and starve times are used as factors in the creation of our prediction equation for boxing.

3.4.3.1 Parameter Estimates for Boxing

Using the factors mentioned above, a multivariate regression was run for the boxing area. Below area the parameter estimates by factors for the Boxing area.

Term	Estimate	Std Error	T Ratio	P-Value
Intercept	-1145.928	314.1743	-3.65	0.0003
Total Block Box	-445.1005	156.5409	-2.84	0.0046
Starve Total	-373.928	44.72017	-8.36	<.0001
Hour[00]	91.597076	30.09283	3.04	0.0024
Hour[01]	259.52653	39.26438	6.61	<.0001
Hour[02]	193.32873	92.12238	2.10	0.0361
Hour[03]	-369.0458	129.881	-2.84	0.0046
Hour[04]	-20.59078	128.8326	-0.16	0.8731
Hour[05]	-11.27967	34.28826	-0.33	0.7423
Hour[06]	111.12703	27.73306	4.01	<.0001
Hour[07]	-56.57066	28.06712	-2.02	0.0441
Hour[08]	73.588522	28.39782	2.59	0.0097
Hour[09]	236.59696	28.05585	8.43	<.0001
Hour[10]	253.96343	27.92961	9.09	<.0001
Hour[11]	-589.7338	28.86436	-20.43	<.0001
Hour[12]	-150.4103	28.679	-5.24	<.0001
Hour[13]	264.44002	28.30215	9.34	<.0001
Hour[14]	186.84029	31.92569	5.85	<.0001
Hour[15]	-351.908	130.4618	-2.70	0.0071
Hour[16]	-34.01453	26.85168	-1.27	0.2055
Hour[17]	190.80668	28.21742	6.76	<.0001
Hour[18]	-307.8806	27.17461	-11.33	<.0001
Hour[19]	-583.1138	27.57487	-21.15	<.0001
Hour[20]	266.35928	27.83412	9.57	<.0001
Hour[21]	264.91244	28.51699	9.29	<.0001
Hour[22]	-143.765	27.70316	-5.19	<.0001
Day of the Week[FRI]	15.177396	12.76464	1.19	0.2347
Day of the Week[MON]	-72.16153	14.52159	-4.97	<.0001
Day of the Week[SAT]	15.495413	25.44874	0.61	0.5427
Day of the Week[THU]	16.677354	13.54749	1.23	0.2186
Day of the Week[TUE]	6.424319	12.54809	0.51	0.6088
Boxing Labor Hours	71.532529	12.7602	5.61	<.0001

Figure 12: Boxing Regression Equation Estimate

The important information that can be pulled out from these equations is as follows. As in the Build area and Kitting area models, the regression baseline case for the categorical Terms, Hour and Day of the Week, were 11pm and Wednesday, respectfully. With factor significance defined by when the P-Value column contains a value that is less than 0.15, there are some assumptions that we can make from our data.

3.4.3.2 Build Prediction Equation

Like in the other areas, knowing the Terms and Estimates, a Prediction Equation can be created that will estimate the average output given the Intercept, Day of the Week

(DOW), Hour, and the number of Labor Hours worked in the Boxing area. Unlike Kitting and Boxing, the amount of time that the Boxing area was blocked and starved proved to be valuable in predicting the output of the Boxing area. To find the average output in the Boxing area, the following equation is derived.

$$\begin{aligned}
 (20) \quad \text{BoxingOutput} &= \text{BoxingInterceptEstimate} \\
 &+ \text{BoxingDOWEstimate} \\
 &+ \text{BoxingHourEstimate} + \text{BoxingLaborEstimate} \\
 &+ \text{BoxingStarvedTotal} + \text{BoxingBlockedTotal}
 \end{aligned}$$

The InterceptEstimate for the above equation is found to be -1145.928 in the Parameter Estimate table for the Boxing area. Additionally, as in the other areas, the Day of the Week estimate and the Hour estimate can be found as well, by matching the day and the hour being predicted with the Day of the Week and Hour estimates in the Parameter Estimates table. Finally the LaborHourEstimate term can be determined by calculating the Labor Hours for the Boxing area using the following equation.

$$\begin{aligned}
 (21) \quad \text{BoxingLaborEstimate} &= 0.00259 * \text{BoxingLaborHours}^3 \\
 &- 0.71 * \text{BoxingLaborHours}^2 \\
 &+ 71.53 * \text{LaborHours}
 \end{aligned}$$

With the prediction equation calculated, the parameter estimates can be analyzed. Below are some observations that we could make from the Boxing area model.

3.4.3.3 *Observations from the Data*

For first shift, good production hours include the hours of 9am, 10am, 1pm, and 2pm. These hours generate at least 100 units per hour *greater* output than the model baseline of 11pm. Poor production hours include the hours of 3am, 11am, and 12pm. These hours generate at least 100 units per hour *less* output than the model baseline. As expected the hourly productivity trends are similar to the trends seen in Build. When excluding lunch hour, 11am, and the first hour of the shift, the average productivity of the workers *increases* as the shift goes on by an average of about 18.5 units per hour.

For second shift, good production hours for Shift 2 include the hours of 5pm, 8pm, 9pm, 1am and 2am. These hours generate at least 100 units per hour *greater* output than the model baseline case of 11pm. Poor production hours for Shift 2 include the hours of 6pm, 7pm and 10pm. Each of these has output volumes that are over 100 units per hour *less* than the baseline case. When excluding lunch hour, 7pm, and the first hour of the shift, the general productivity of the workers also *increases* as the shift goes on by an average of about 38 units per hour.

While days are generally mixed in measured significance, there are a couple of interesting facts can be pulled out. Monday performance is significant and shows that productivity is generally poor, yielding 56 units *less* per hour than other hours and Saturday is a highly productive day yielding an extra 43 units per hour worked. Also, by using these predicted values and observing the general trend for the week, it can be seen see that output *increases* by an average of 14 units per hour per day throughout the week. This trend is fairly significant, however, variation does exist that could mean the trend data is not enough to measure daily effects.

One big indicator that this shows is that breaks once again have an impact on productivity. The hour following lunch in first shift, 12pm, produces 150 units *less* than the baseline hour, and the hour before lunch during second shift, 6pm, produces 307 units less than the baseline hour. There is a two-hour window of negative impact around lunch breaks for each shift.

Finally, by looking at the relationship between labor hours and productivity it can be seen that productivity is once again a polynomial function. An increase in output due to labor hours can be approximated using Equation 21, which is shown again below.

$$\begin{aligned} \text{LaborEstimate} &= 0.00259 * \text{LaborHours}^3 - 0.71 * \text{LaborHours}^2 \\ &+ 71.53 * \text{LaborHours} \end{aligned}$$

This shows that productivity is not a linear relationship, but one that has polynomial growth in output units with every labor hours invested, just as in the other areas. This equation would suggest that there are some economies of scale that increase the productivity of the area as the area becomes more staffed.

3.4.3.4 Measure of Predictability of Boxing

Below is a plot of the predicted value versus the actual value plotted against each other for all hours analyzed. Additionally, a summary of the model's fit is included.

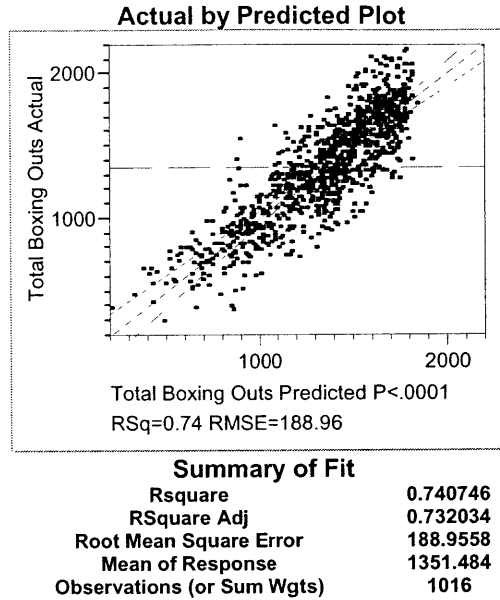


Figure 13: Boxing Regression Analysis

For our boxing prediction, there is a close general trend exists relating the prediction equation to actual values. The R-square value suggests that the model accounts for nearly 74% of the variation in production output. Once again, this is a decent prediction for the boxing area, but not great. Some amount of variance should be eliminated from this model to make it more reflective of actual production numbers.

3.4.3.5 Variation between Predicted and Actual Values

To test for independence, the residual values are again looked plotted against the actual output volumes. That plot of the residuals is shown below.

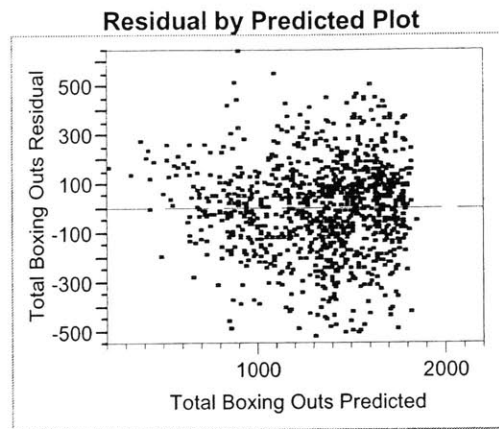
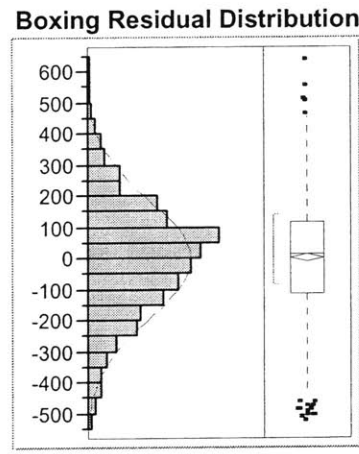


Figure 14: Boxing Residual Analysis

As with the kitting and build plots, the residual plot for boxing has a slight convexity, however, there is not severe trends in the residual plots. Additionally, from the distribution of the residuals around the predicted value, there is once again a Normal distribution. This distribution chart is shown below.



Normal(-1.69e-12,185.859)

Moments

Mean	-1.69e-12
Std Dev	185.85874
Std Err Mean	5.8309073
upper 95% Mean	11.442012
lower 95% Mean	-11.44201
N	1016

Goodness-of-Fit Test

Shapiro-Wilk W Test	
W	Prob<W
0.982226	0.0192

Figure 15: Boxing Residual Distribution and Goodness-of-Fit

These plots are important for showing that the residuals are normally distributed around the predicted output values. The Shapiro-Wilk test measuring Normality shows that 98% of the variation of the residuals can be explained through normal variation. For boxing, by looking at the residuals it can be seen that there is a mean difference from each hour of actual output of zero (mean of $-1.69e-12$). Our measured standard deviation of the model is 186. In general, we now have a model for the Boxing area that can be calculated by using the prediction equation from Table 3 to solve for a predicted output volume, and the residuals can be quantified by a normal distribution around that predicted value by a standard deviation of 186 units per hour.

3.4.4 Distribution Capacity

While this is not tracked in the normal WIP tracking tool of the factory, some measurement of the distribution area has been completed through other internal methodology. Analysis suggests that Distribution capacity is about 3300 units per hour. Variability in the Distribution area is caused by downtime and other factors. However, the maximum-recorded output from Distribution was 4275 Units per Hour, so it is assumed that the Distribution is able to make up for downtime events by working at a much faster rate of speed than the other areas. Additionally, in case there is an excursion that causes Distribution to fall behind the other production areas in the factory, the storage facilities in the Distribution area can be used to temporarily hold product until the Distribution area catches up with the rest of the factory.

3.5 Example - Predicting Area Output by Hour

For each Kitting, Build, and Boxing, the methodology used to collect the data, the prediction equations, and the analysis of the residuals was discussed. With that information, this section will give an example of how that analysis can be used to predict the output for the area. For this example, let's assume that the prediction equation is being used to predict the output for the Build area, given the following influencing factors.

Factor	Value
<i>Day</i>	Monday
<i>Hour of the Day</i>	9am
<i>Headcount in Build</i>	343 people
<i>Portion of Hour Worked</i>	Full Hour (1 hour)

Table 6 - Example Factors for Determining Output

Using these factors, and the prediction equation the average (expected) output can be calculated for the Build area. From the prediction equation, the Intercept for the model is shown to be -2201 units. Because the day of the week is Monday, the Output is computed by using the Estimate of the “Day of the Week [Monday]” Term, resulting in an adjustment of -56 units. Since the hour of the day is 9am, the Hour[9] Estimate would be used to find that there is an adjustment of 146 units. Finally, to calculate the Labor Hours in the Build area, the number of workers is multiplied by the percent of hour worked, as in Equation 16.

$$\begin{aligned}
 \text{BuildLaborHours} &= \text{PercentHoursWorked} * \text{BuildHeadcountTotal} \\
 &= 1.00 \text{ (full hour of work)} * 343 \text{ workers} \\
 &= 343 \text{ workers}
 \end{aligned}$$

In this case, the labor hours will be 343 labor hours as 343 floor associates each worked one full hour. Knowing the labor hour’s value, and utilizing the Build Labor Hours Equation, the Labor Hours adjustment is calculated, using Equation 18, as shown below.

$$\begin{aligned}
 \text{BuildLaborEstimate} &= 0.000072 * \text{BuildLaborHours}^3 - 0.071831 \\
 &\quad - \text{BuildLaborHours}^2 + 26.405038 * \text{BuildLaborHours}
 \end{aligned}$$

Plugging in the BuildLaborHours value, 343, into the equation, the following Build Labor Estimate is calculated.

$$\begin{aligned}\text{BuildLaborEstimate} &= 0.000072*343^3 - 0.072*343^2 + 26.4*343 \\ &= 3511\end{aligned}$$

With the output due to labor hours estimated, and the other term estimates known, the average output in the Build area can be calculated using Equation 17, shown again below.

$$\begin{aligned}\text{BuildOutput} &= \text{BuildInterceptEstimate} + \text{BuildDOWEstimate} \\ &+ \text{BuildHourEstimate} + \text{BuildLaborEstimate}\end{aligned}$$

Plugging in the BuildLaborEstimate value, 3511, as well as the Build Intercept value of -2201, the Day of the Week (DOW) adjustment of -56, and the Hourly adjustment of 146, the output is determined as shown below.

$$\begin{aligned}\text{BuildOutput} &= - 2201 - 56 + 146 + 3511 \\ &= 1400 \text{ Units}\end{aligned}$$

Thus, the average output in the Build area given the example values is calculated to be 1400 units. While this is a good estimate of output, as the area analysis showed above, the output will vary compared to our predicted output. By characterizing the residuals, or measurement that the predicted value was off from the actual output for the Build area, it was found that the residuals were characterized by a Normal distribution around our predicted value, with a standard deviation of 153 units per hour. A graph of this distribution can be seen in the Figure below.

Build Residual Distributions

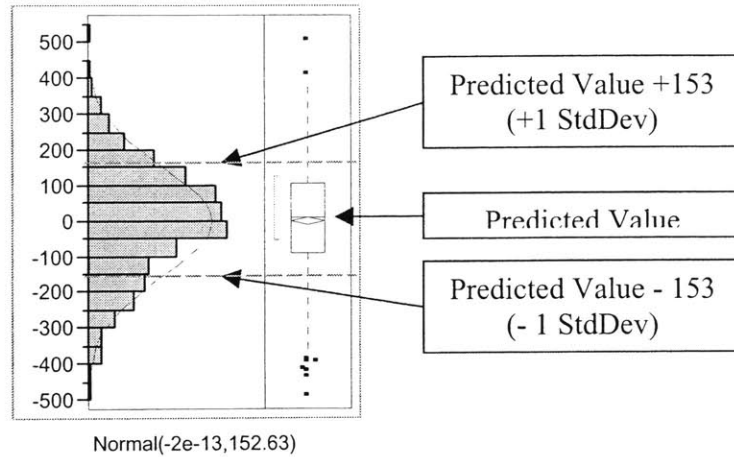


Figure 16: Predicted Output with Normal Distribution of Residuals

. The expected output calculated from our prediction equation for Build, labeled as the predicted value above, can be seen in a residual plot, as the zero value for the residual plot. Likewise, one standard deviation above and below the predicted value is labeled at +152.63 and -152.63 respectively. By definition of a Normal distribution, 68% of all the actual data points will fall between +/- 152.63 units of the predicted value. In fact, because the distribution of values have the characteristic of being a Normal distribution, the probability that the true output will fall above a given value, below a given value, or within a given range of points above or below our prediction can be calculated as well. In fact, a Normal distribution can be used to calculate the probability of getting any output value, given a predicted output value and standard deviation of the residuals. As the optimization model is built this becomes important.

4 BUILDING A NEW HEADCOUNT OPTIMIZATION MODEL

An optimization model can be used to iteratively determine the maximum or minimum value of one variable, given a set list of constraints. There are two bits of information that the optimization needs in order to be created. First, the optimization model needs an objective function. The objective function is the goal for the optimization, which in the case of the Dell headcount problem, would be to determine the minimum headcount needed in order to meet the production output demand each shift.

An optimization model also needs constraints defined for the model, in order to set bounds for the possible solutions for the problem. In the headcount optimization, the

model will be bound by a couple of things. First, the output should always be greater than or equal to zero. This in turn implies that the headcount level should never fall below zero. Additionally, the output from each area every day should be equivalent. Over time, there should never be an imbalance in the amount of material made from area to area in a just-in-time system. The prediction equations and the variation that has been determined for each area hour will be a constraint of the optimization as well. Using this information, the average output for the shift and variability can be determined. Using the shift output for each area, and the variability the optimization can determine the appropriate headcount needed in each area to meet the demand of the factory and ensure the bottleneck does not move between areas.

Finally, the optimization created for this headcount model will allow the managers of the model to vary the Shift, Day, Hours in the Shift, and Output Demand. In addition, the managers will be able to determine what level of confidence they want to ensure that the constraint area meets the production volume for the Shift, as well as the percent confidence that the constraint does not move throughout the factory. With all of the above information, all the pieces are available to build the optimization model. The headcount in each area will be varied, until a minimum headcount to meet the overall production volume is met. The above optimization process will be discussed in great detail throughout this Section and the optimization output will be described.

4.1 Determining Mean Output and Standard Deviation for a Shift

The prediction equation analysis of each of the areas could be used to determine each area's hourly output, given factors such as labor hours, day, and time of the week. While it is important to understand the hourly mean and variance that exists for each area, as calculated in the last section, Dell's headcount planners are not interested in predicting headcount hour by hour. Instead, they are interested in understanding how much headcount is needed to match their shift output needs within the factory. That means the optimization model should have a way of predicting headcount needed for an entire shift, which requires that we calculate the mean and standard deviation of the output for a whole shift. This is not straightforward because the shift lengths can range between five and 11.5 hours depending on the demand. However, given the length of the shift, the shift output mean and variance can be approximated. This approximation is described below.

4.1.1 Accounting for the Length of the Shift

The factory is broken into two shifts and the demand for the shifts and hours of work are determined with only one day of lead-time. The floor associates come in the next day and work until the backlog of orders is depleted to a given level. Because of this, shift length can vary to some degree from day-to-day. However, once the number of hours is determined for the following day, the actual schedule that the floor associates work is fairly consistent. Because the hours that are worked are consistent, and the prediction equations created for each area have hourly approximations of output rates, the information about what hours the associates work can be used to better determine the shift output. Knowing how many work hours are planned for the shift, the schedule will tell what hours to include in our model. The shift total will be the total of each hour's mean output, and the variability for the shift can be estimated as well.

Looking at First shift, to determine what hours the associates will work, given the scheduled shift length (SL), the following business rules can be used.

Shift	Shift Scheduled Hours											
<i>First</i>	Start Time:											
	<table border="1"> <thead> <tr> <th>Shift Length (SL)</th> <th>Start Time</th> </tr> </thead> <tbody> <tr> <td>SL ≤ 9 hrs</td> <td>6am</td> </tr> <tr> <td>9 hrs ≤ SL ≤ 11 hrs</td> <td>6-(SL - 9) am</td> </tr> <tr> <td>SL ≥ 11 hrs</td> <td>4am</td> </tr> </tbody> </table>	Shift Length (SL)	Start Time	SL ≤ 9 hrs	6am	9 hrs ≤ SL ≤ 11 hrs	6-(SL - 9) am	SL ≥ 11 hrs	4am			
	Shift Length (SL)	Start Time										
	SL ≤ 9 hrs	6am										
	9 hrs ≤ SL ≤ 11 hrs	6-(SL - 9) am										
	SL ≥ 11 hrs	4am										
	End Time: Start Time + Shift Length											
	Lunch: 7pm											
	Break: varies – generally 10 pm											
	Start and End Examples:											
<table border="1"> <thead> <tr> <th>Shift Length</th> <th>Start Time</th> <th>End Time</th> </tr> </thead> <tbody> <tr> <td>7 ½ hours</td> <td>6am</td> <td>1:30pm</td> </tr> <tr> <td>10 ½ hours</td> <td>4:30am</td> <td>3pm</td> </tr> <tr> <td>11 ½ hours</td> <td>4am</td> <td>3:30pm</td> </tr> </tbody> </table>	Shift Length	Start Time	End Time	7 ½ hours	6am	1:30pm	10 ½ hours	4:30am	3pm	11 ½ hours	4am	3:30pm
Shift Length	Start Time	End Time										
7 ½ hours	6am	1:30pm										
10 ½ hours	4:30am	3pm										
11 ½ hours	4am	3:30pm										

Table 7 - TMC First Shift Work Hour Rules

From Table 7, we see how the start and end time for the shifts are determined within the factory for Shift 1. An example of how to use the above table to determine the start and stop time of the shift is given below. Assuming that the shift being worked is first shift (S1). Assuming also that the shift length is scheduled to be 10.5 hours, the start time and stop time of the shift can be calculated using the equations below:

$$\begin{aligned}
 (22) \quad S1 \text{ Start Time} &= 6\text{am} - (\text{Shift Length} - 9)\text{hrs} \\
 &= 6\text{am} - (10.5 - 9)\text{hrs} \\
 &= 4:30\text{am}
 \end{aligned}$$

$$\begin{aligned}
 (23) \quad S1 \text{ Stop Time} &= \text{StartTime} + \text{Shift Length} \\
 &= 4:30\text{am} + 10.5 \text{ hrs} \\
 &= 3 \text{ pm}
 \end{aligned}$$

Using the equations shown in the table above, the start and stop times for the first shift with any shift length can be determined. Likewise, the shift hours for the second shift are easy to determine as well. A table that includes the business rules to determine the second shift (S2) hours is shown below.

Shift	Shift Scheduled Hours						
<i>Second</i>	<p><i>Start Time:</i> Always starts at 4pm. <i>End Time:</i> Start Time + Shift Length <i>Lunch:</i> 7pm <i>Break:</i> varies – generally 10 pm</p> <p>Start and End Examples:</p> <table border="1"> <thead> <tr> <th>Shift Length</th> <th>Start Time</th> <th>End Time</th> </tr> </thead> <tbody> <tr> <td>7 ½ hours</td> <td>4pm</td> <td>11:30pm</td> </tr> </tbody> </table>	Shift Length	Start Time	End Time	7 ½ hours	4pm	11:30pm
Shift Length	Start Time	End Time					
7 ½ hours	4pm	11:30pm					

Table 8 – TMC Second Shift Work Hour Rules

Second shift always starts at 4pm, and the end time is determined by the number hours that will be worked during the shift. As an example, a 10.5-hour second shift will have a start time of 4pm and an end time of 2:30am. The scheduled work hours will be important in the calculation of the mean output for the shift, demonstrated in the next section.

4.1.2 Calculating the Mean Output for the Shift

As mentioned, knowing the hours that will be worked now can be used to determine which hours will be worked during each shift. Because each hour's output is distributed normally, the individual hour mean outputs for each area can be summed to find the average output for the shift. For example, if the model is being used to

determine the mean Build output for Shift 1, and the shift lasts eight hours, starting at 6am, a mean output for the shift can be determined by adding up the mean hourly outputs for the hours of 6am, 7am, 8am, 9am, 10am, 11am, 12pm, and 1pm. As an example, if it is assumed that the output for the first shift is being calculated, and it is for a Monday with 343 factory associates working, the average shift output would be equal to the mean output for each hour between 6am and 1pm. To calculate the mean, the hourly means are summed to create an average output for the shift. In this example, the mean shift output would be as shown in the following table.

	Labor Output	Intercept	Monday Adjustment	Hour Adjustment	Total Output
6am	3511	-2201	-56	41	1295
7am	3511	-2201	-56	-140	1114
8am	3511	-2201	-56	56	1310
9am	3511	-2201	-56	146	1400
10am	3511	-2201	-56	140	1394
11am	3511	-2201	-56	-587	668
12pm	3511	-2201	-56	-208	1046
1pm	3511	-2201	-56	133	1386
				Shift Avg =	9614

Table 9 - Shift Output for an 8-hour shift on Monday with 343 workers

As shown above, the hourly outputs were calculated from the prediction equation for the Build area. The average output for the shift is equal to sum of each hour's average output, which would be **9614** units out of the Build Area.

4.1.3 Calculating the Variance of Output for the Shift

While the average output is good to know there is also power in knowing the amount of variability that exists during the shift. The most basic approach assumes that each hour's output is independent of the output from any of the other hours' outputs. This assumption would infer that if a bad hour were experienced in the factory, it would in no way impact the output from any other following hour. Calculating the amount of variability in the shift output in this case is easy to calculate. To do this, the shift output would equal the square root of the sum of the squares of each individual hour's standard deviation. In this case, the standard deviation for every hour is equivalent, so the

equation to calculate the standard deviation of the shift output variability is written below.

$$(24) \quad \text{Shift StdDev} = (\text{Number of Hours in Shift} * \text{StdDev(Hour)}^2)^{1/2}$$

As an example, using this equation, the standard deviation of the output variability for the Build area for an 8-hour shift would be equal to:

$$\text{Shift StdDev} = (8 * (153)^2)^{1/2} = 433 \text{ units}$$

In this example, by definition of Normal distributions, there is a 68% chance that Build's output would fall between 433 units above and below the average of 9614 units, or restating this, of falling between 9183 and 10049 units. By calculating the shift output averages and standard deviations for each area, validation of the shift output predictions can be done by comparing our calculated averages to the actual output for known hours. That activity was performed and is described below.

4.1.4 Validating the Area Models

In order to validate each areas model the actual output was compared with what the prediction equation for each area estimated the output. In order to validate the model, the residuals were tested for Normality. If the actual output minus the predicted output is truly a Normal distribution, the difference between the actual and the predicted values should mean that 50% of the points should lie above the zero and the other half should lie below zero. Stated a different way, when comparing the actual and predicted outputs, the predicted value should lie above the actual value, roughly 50% of the time, and it should lie below the actual value roughly 50% of the time. Additionally, if the residual value is truly a normal distribution, it should create a standard normal distribution where 68% of the actual output falls within one standard deviation above or below the predicted value. For the simple estimate of variability, that would mean 68% of the actual shift outputs should fall within 433 units of the mean. The results of the validation for the Build area are found in the Appendix. What was found was that when the standard deviation for the shift was calculated, assuming that each of the hours was independent, only 58% of the shift data points fell within the estimated 1 standard deviation, or 433 units, of the output

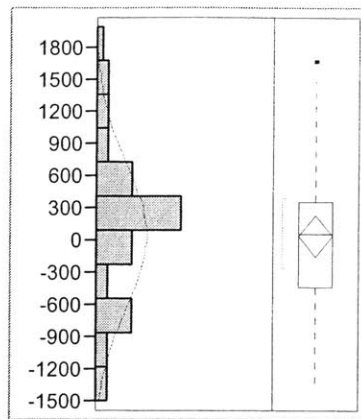
for the Build area. This suggests that the assumption that each area's output was independent of every other hour's output was most likely incorrect.

Looking at the distribution a different way, a second method can be used to determine the standard deviation of output each shift. Instead of the equation used above to determine the standard deviation, which assumes no interdependence between factors in our prediction equation, a new equation can be used, as seen below.

$$(26) \quad \text{Shift StdDev} = (\text{Number of Hours in Shift} * \text{StdDev(Hour)})^2 + (\text{Hourly Covariance of All Factors})^2)^{1/2}$$

This equation includes a term "Hourly Covariance of All Factors" that represents the total interdependence of each factor, such as hour of the day and day of the week that could exist and should be used to predict the shift output. Ideally, the interdependence between each combination of factors in the regression model could be determined, however, the prediction equation would become very complex. Part of the consideration when creating an optimization model would be the ease of replication and implementation. For that reason, the interdependence between all factors was looked at as a whole. To solve for the standard deviation, the "Hourly Covariance of All Factors" was approximated until 68% of the historical data fell within one standard deviation of the mean. The "Hourly Covariance of All Factors" was approximated to be 230 units per hour.

While this approximation was a good method that takes into account the number of hours in the shift, ease of implementation was again a major concern for this model. Because this prediction technique would need to be done on a systematic basis, and someone would support the model with less knowledge about statistics, it was decided that an even easier method needed to be created. To simplify the process for approximating the output, the final approach used was to subtract the average shift output calculated in the previous section from the actual shift output and look at the distribution. In this case, the predicted shift output minus the actual output can be seen in the normal distribution as shown below.



Normal(38.1333,673.7)

Goodness-of-Fit Test

Shapiro-Wilk W Test

W	Prob<W
0.972491	0.4980

Figure 17 - Distribution of Predicted Shift Output - Actual Shift Output

The distribution of the predicted shift output minus the actual shift output is a normal distribution with an average of 38 units per shift and a standard deviation of nearly 690 units. That means our prediction equation had nearly the same average output as the actual output, and 68% of the actual values were within 690 units of the predicted shift output. For the case of our headcount optimization, that was a close enough approximation to use overall.

The same type of analysis was done for Kitting and Boxing was done as well. These areas had predicted output that averaged the actual output overall, and they had standard deviation values of 700 units per hour and 750 units per hour respectively. These relationships will be used for an overall factory model shortly.

4.2 Choosing a Constraint

One problem with the old headcount model at Dell is that the model for the factory sets the output target of each area equal to the overall target for the factory area. Below is the old factory headcount model as shown in the thesis description section.

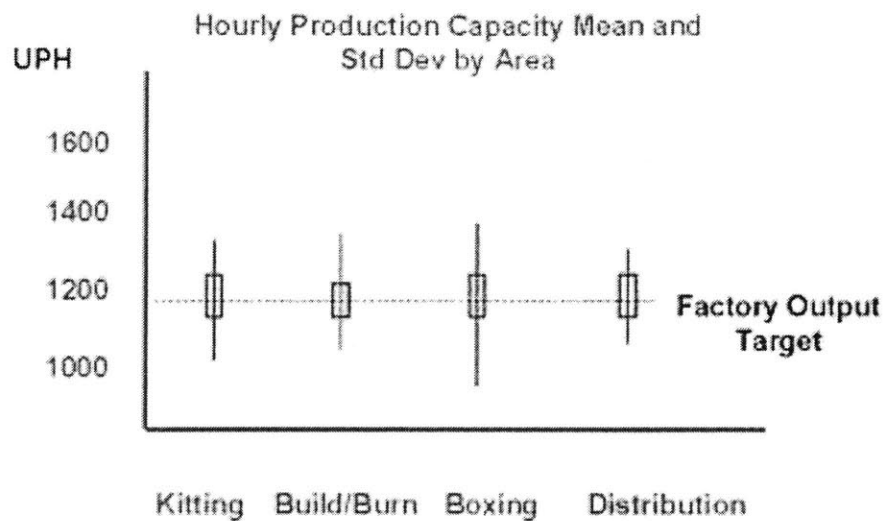


Figure 18 – Old Model Output by Area

Recalling the issues seen by using this type of output model, it was shown that with the areas output targets set to the factory output target, the constraint of the factory could move between areas, and may cause the situation where you miss your daily output targets for the factory consistently. To address this issue, a new headcount model was built using the Theory of Constraints assumption that your factory constraint should be chosen and the constraint of the factory should always stay in one area of the factory. The Theory of Constraints also states that after deciding which area will be the factory constraint, the production capacity of each of the other areas should be raised enough to prevent them from ever becoming the bottleneck of the factory again. Elevating the headcount in the non-constraint areas can help boost those areas production capacities. By making one area remain the constraint, controlling the factory output is much easier, because to control the output of the factory, you need to only concentrate on controlling the output of the constraint area in the factory. Also, there is much less variability possible in the output of the factory.

To decide which area will be the factory constraint can be determined by looking at some key information. First, there are hard constraint limits in the factory due to the capital equipment capacity or hard limit bottlenecks. An area will never have higher production rates than that of the hard limit bottlenecks. The second main consideration is the cost of adding production capacity is a main concern. The area with the highest marginal cost per extra unit produced is the main determinant of cost of capacity.

Another factor that may be considered is the area's activity involved in the production of the material and the level of expertise and activity that needs to be performed in each area of the factory. For areas with a large amount of activity of highly skilled or specialized jobs, it is more important to keep activity level high. If these areas start and stop frequently, repetition is interrupted, and the concentration of the workers could be impacted. This could cause less than optimal throughput rates, or high error potential. One final consideration is determining which area has the lowest amount of variation in its output rate. By selecting the area with the lowest variability, the factory output rate will be less variable and more controllable, and therefore easier to predict.

In the case of the TMC Factory, the Build area was chosen as the constraint. Build is the highest capital cost area in the factory to upgrade and it is also the most labor intensive. Additionally, from the analysis of each of the areas, the Build area appeared to have the lowest variability of any of the production areas. For those reasons, it makes sense that the bottleneck should remain in Build. This area was also identified by production control as the designed constraint for the factory. Recalling the historical Block and Starve data, it was shown that the constraint of the factory was Build at nearly 21% of the time. With this information in mind, the model was designed to force Build to always be the constraint of the factory. In order to create a model that forces Build to be the slowest area in the factory, the production capacity of Kitting, and Boxing can be raised to ensure that Build is the slowest performing area in the factory. The following graph demonstrates how this is accomplished.

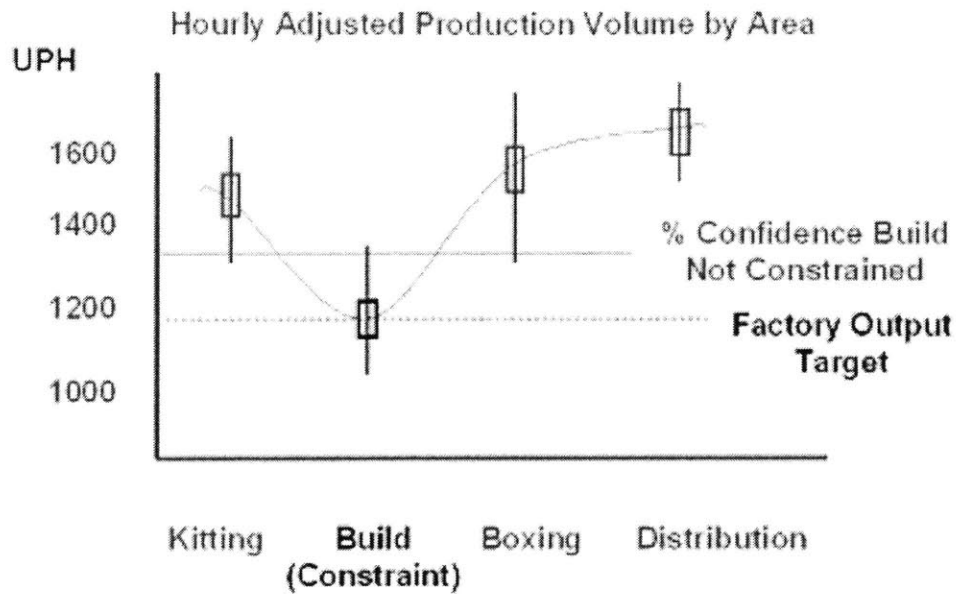


Figure 19 - The New Factory Headcount Model

Applying this design to the area prediction models that were derived, we can now build an intelligent headcount model that will make sure the constraint stays where we want it in the factory. Because each of the area output distributions are normal distributions, it is easy to calculate the probability that the area will perform above or below a given production volume given the input factors such as labor hours or day of the week. Because those probabilities can be calculated, headcount can be adjusted in Kitting and Boxing that will artificially raise their area capacities and ensure these areas outperform Build a majority of the time (to the confidence interval that is desired by the factory).

Given the basic premise and design of the factory headcount model above, the final system optimization model was created using the area prediction models from Kitting, Build and Boxing. To do this, the model takes the prediction equation of each area, and sets their headcount targets so that Kitting and Boxing will statistically not allow the Build area to be blocked or starved a majority of the time. By doing this, Build will remain the slowest area in the factory and the Build area will solely determine the factory output. An explanation of how the optimization works follows.

4.3 Factors Used to Create the Headcount Optimization

Now that the design for the optimization has been discussed, the overall headcount optimization can be created. To do this an optimization was created to include the three

possible constraint areas – Build, Kitting and Boxing. To build an optimization some specific information needs to be defined. First, the objective function must be established. The objective is to minimize the headcount needed to achieve the needed output target for the area. The solver utility will vary headcount, given all other factors to meet the output desired for the shift with the minimum amount of headcount possible.

4.3.1 Inputs for the Optimization

The optimization being created is setup to allow the managers within the factory to determine headcount needed for varying conditions. Because of this, it requires the ability for managers to configure the solver to meet the conditions by which the solver will try to optimize the headcount in the factory. These configuration variables are included below:

Variable	Description
Target Output	The target output for the shift is defined by the factory’s shift demand
Shift	The model can be run for first or second shift.
Number of Hours Worked	This is the total hours worked in the shift.
Day of the Week	The model can be run for Monday, Tuesday, Wednesday, Thursday, Friday, or Saturday.
Percent Confidence of Making the Output Target	This is used to determine the probability of making the output goal for the shift. This is set to 50% for general case. (This will be explained in further detail below.)

Table 10 - List of Variables for Build Optimization

While most of the above factors are straight forward, percent confidence is a little difficult to understand. This attribute is further described below.

4.3.2 Percent Confidence of Meeting our Output Goal

If there is a goal for the output needed within a shift of 9000, the probability of achieving that goal will vary as the mean output of the area changes. If the area is staffed with a greater amount of headcount that the amount needed to average 9000 units per shift, the probability of reaching 9000 units per shift will be greater than 50%. Assume that for the example in which there is a goal of reaching 9000 units per shift and that the factory is staffed at a level that typically would average 9614 units of output. Normal distributions allow the computation of the probability of meeting an output goal given all

of the different production factors, if we know simply the average output and the standard deviation of the residuals. With the mean and standard deviation of 9614 and 690 respectively, we can solve for the probability of building at least 9000 units. Graphically, this is shown below:

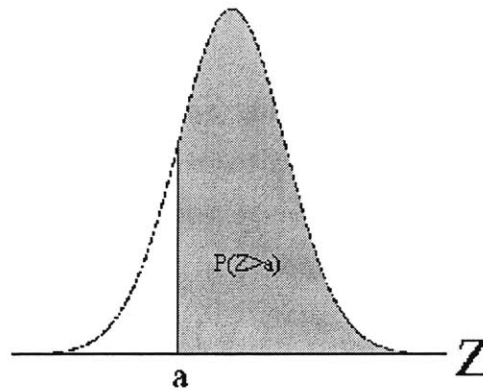


Figure 20 - Probability of at Least “a” units being produced

The above picture shows the probability (in grey) of getting “a” units or greater units produced in the factory given an average output of 9614 units. The following equation in conjunction with a cumulative normal distribution table from a Statistics text can give us an estimate of the probability of greater than “a” units from our factory area.

$$(27) \quad P(Z > a) \quad = \quad 1 - \Phi((\mu - a) / \sigma)$$

In the above equation, the “a” variable represents the number of units the model is trying to exceed. By plugging in the mean of the Normal Distribution, μ , and the standard deviation, σ , of the Normal Distribution, we get the following results.

$$\begin{aligned} P(Z > a) &= 1 - \Phi((9000 - 9614) / 690) \\ &= 1 - \Phi(-0.89) \end{aligned}$$

Looking in the included Normal Distribution table in the Appendix, we can now calculate the probability that the actual output will exceed 9000 units.

$$\begin{aligned} P(Z > a) &= 1 - 0.1867 \\ &= 0.8133 \end{aligned}$$

Using cumulative distribution tables included in the Appendix, the probability of exceeding 9000, in a distribution with a mean of 9614 and standard deviation of 690 is nearly 81%, meaning there is an 81% chance of hitting 9000 units per shift if the average output achieved for the shift with the given headcount is 9614.

Using this above relationship, a percent confidence as well as the other factors discussed can be chosen by the managers at Dell, and the solver program will adjust the headcount in order to make sure that the workforce meets the output goal. The solver will stop when the output objective is met, and it has the minimum headcount possible.

4.4 Creating the Build Headcount Optimization

Using the optimization methodology that was discussed above, the Build area headcount model was created. The output goal for the Build area is equal to the output goal for the factory. That is because the Build area is the constraint of the factory and as mentioned, the Theory of Constraints states that the factory output should be equal to that of its constraint. For this optimization, the model assumed a 50% confidence level of achieving the productivity output in the factory. With a 50% confidence level, over time the predicted output will average out to equal the actual average output of the factory. However, if it is more crucial that the factory meets its production output level, the managers might set the factory's percent confidence at 60% or even 80% in order to have a higher chance of meeting the factory's production goal. This factor is at the manager's digression. The manager can also vary the day, shift and shift length as well. With these factors, and knowing the build standard deviation was estimated to be 690 units, the optimization will iterate until it determines the headcount needed to meet the output needs. The final optimization model for build is shown in the Appendix.

4.5 Kitting Headcount Optimization

Like the Build optimization, to optimize the headcount needed to meet a given output level in the Kitting area requires essentially the same information as was needed in the Build optimization. However, there are a couple of distinct differences. The output goal

for the Kitting area is different than the output goal in the Build area in that the Kitting area is not the constraint in the factory. While the Build area's job is to set the pace of the factory, and has its output goal set to the output target for the factory, the Kitting area's job is to make sure that the Build area is never starved of material. Because of this, the goal for the Kitting area should be to be able to produce units much faster, and thus, never starve the Build area. For that reason, the Kitting area's production goal varies with the Build area output.

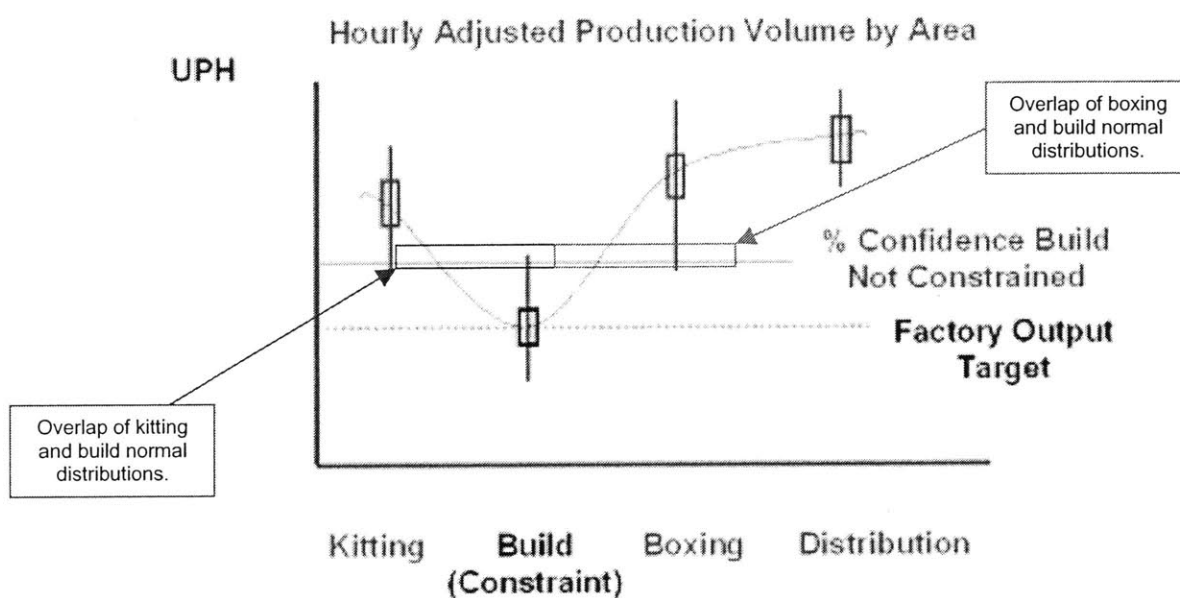


Figure 21 - Adjusted Production Volumes of Kitting and Boxing Output with Build Output

As shown in the above figure, the Kitting area's output goal must be set higher than the Build area's output goal. With the Build area's output determined to be the factory's output target, there is some variability in the actual amount produced by the Build area. The variation in the Build area has a standard deviation of 690 as discussed, and the Kitting area has a shift standard deviation as well of 700. To make sure that the Kitting area will never starve the Build area of material, the Kitting area's output must be elevated enough to ensure that despite the variability that exists in each of these there is a low probability that the Build area will outperform the Kitting area.

To find the probability that the Kitting area will outperform the Build area can easily be calculated. Because the Kitting and Build output values are both Normal distributions,

the actual difference between the two values is, itself, a Normal distribution. To calculate the Normal distribution of the difference between the Kitting and Build Output, the following relationships can be used.

$$(28) \quad \text{KB Diff. Mean} \quad = \text{Kitting Output} - \text{Build Output}$$

$$(29) \quad \text{KB Diff. Std Dev} \quad = (\text{Kitting Output Std Dev}^2 + \text{Build Output Std Dev}^2)^{1/2}$$

With the difference mean and standard deviation defined, the method used earlier to determine the probability of meeting the shift goal given a different mean and standard deviation can be used here as well. Determining the probability of the Kitting area having a higher output than the Build area is equivalent to determining the probability that the difference between Kitting Average Output and the Build Average Output (Diff Mean) being greater than zero, given the standard deviation of the difference (Diff Std Dev). To help demonstrate how this is calculated, an example is included below.

4.5.1 Example: Probability of Kitting Output Greater than Build Output

Assume that Kitting and Build have an average output of 10,000 and 9,300 respectively. Also assume that Kitting and Build have Standard Deviations of 690 and 700 as discussed in previous sections. Using equations 28 and 28 to solve for the mean and standard deviation of the difference in production output between the Kitting and Build area, the following calculations can be performed.

$$\begin{aligned} \text{Diff Mean} &= 10000 - 9300 \\ &= 700 \text{ Units per Shift} \end{aligned}$$

$$\begin{aligned} \text{Diff Std Dev} &= (690^2 + 700^2)^{1/2} \\ &= 983 \text{ Units per Shift} \end{aligned}$$

Using those two values, the probability of Kitting outperforming Build can be calculated using Equation 27, which expressed the probability of achieving or exceeding

a production goal given the mean and standard deviation of the difference between their Normally distributed values.

$$\begin{aligned} P(Z > a) &= 1 - \Phi((9300 - 10000)/983) \\ &= 1 - \Phi(-0.71) \\ &= 1 - 0.2389 \\ &= 0.7611 \end{aligned}$$

Therefore, there is a 76% chance that Kitting area will outperform the Build area in the factory for the shift. Because the Kitting area outperforms the Build area, the Build area should not be starved of material.

4.5.2 Determining the Output Goal

Knowing our output goal for Build, the goal for the Kitting area can be determined. Basically, the optimization model created will take as input the day, shift, the number of hours in the shift, and a probability that management wants the Kitting area to outperform the Build area. Using these factors as well as the Build output goal and the variability of the Kitting and Build area, the optimization will set a mean output goal for the Kitting area as in the section above. Then the optimization will fluctuate headcount in order to ensure the output goal is met and the minimum headcount is used. The spreadsheet for the Kitting area is shown in the Appendix.

4.6 Boxing Headcount Optimization

The Boxing and Kitting area essentially calculated in the same manner. Once again, the Boxing area's job is to make sure that Build is not blocked and can always push material out. For that reason, the Boxing area's production goal is again determined by the Build area's output. Once again, the probability of the Boxing area outperforming Build is equivalent to finding the probability that the difference between the Boxing area output and the Build area output is greater than zero. Given the Boxing and Build mean outputs and the standard deviations of the two areas the probability of the difference can be determined by calculating the mean difference between the two areas output levels and the standard deviation of the Boxing area outputs minus the Build area outputs. The

equations for the mean difference and standard deviation between the Boxing area output and the Build area output are given below:

$$(30) \quad \text{BB Diff. Mean} \quad = \text{Boxing Mean Output} - \text{Build Mean Output}$$

$$(31) \quad \text{BB Diff. Std Dev} \quad = (\text{Boxing Output Std Dev}^2 + \text{Build Output Std Dev}^2)^{1/2}$$

Like in the example for the Kitting area, the standard deviation of the difference between the Boxing output distribution and the Build output distribution could be determined outright, using Equation 31 above. Knowing that the Boxing output distribution standard deviation equals 750, and the Build output distribution standard deviation equals 690, the standard deviation of the difference between these area distributions is be calculated.

$$\begin{aligned} \text{Diff. Std Dev} &= (750^2 + 690^2)^{1/2} \\ &= 1019 \text{ units} \end{aligned}$$

Thus, the standard deviation of the difference between the Boxing and Build area is 1019 units. Now that the mean difference between the two areas output means and the standard deviation of the difference between the means has been characterized, they can to determine the probability that Boxing will outperform Build. Just as in the Kitting area, to do this an output target for Boxing is set that ensures that Boxing outperforms Build a majority of time. Basically, the optimization model created will take as input the day, shift, the number of hours in the shift, and a probability that management wants kitting to outperform Build. Using these factors, the output goal for the Boxing area will be determined by the optimization and then the optimization will fluctuate the headcount in Boxing to determine the minimum headcount needed to meet the area's production goal.

One final difference between Boxing and Kitting is that in Boxing, Block and Starve metrics are used to help us approximate the output in the area. Since it was a predictor in our equation, the managers in the area can estimate the amount of block and starve time that will be expected in the Boxing area over the shift. For the initial model

the block and starve time was set to the average block time recorded for the duration of the internship. The averages over the period, as computed above were roughly 30% Starve time, and 5% Block time. These values were used in the optimization. The spreadsheet for the Boxing area is shown in the Appendix.

4.7 Optimization Model Summary

In order to make it so the user of the Headcount Optimization only had to input data at one place, a control page was created to put the Optimization model together. The input needed for the optimization to run, predicting headcount in each of the three areas, could be consolidated into a short list of input parameters. They include the Shift Output Goal, the Probability of Build meeting the Shift Output Goal, the Probability that the Kitting area outperforms the Build area, the Probability that the Boxing area outperforming the Build area, the Day of the Week, the Shift, the Number of Hours in the Shift, the Block Time in the Boxing area, and the Starve Time in the Boxing area. As described in the previous section, the model also took into account the standard deviations for each of the areas in determining the optimal output level of each area. With the above information, an optimization model for the factory was created to minimize the total headcount needed in the factory, establish and maintain a constraint in the Build area, and ensures that the factory meets its output goal. The overall optimization is included in the Appendix.

5 HEADCOUNT MODEL RESULTS AND RECOMMENDATIONS

After creation of the overall optimization model, it was tested against historical headcount and output data to measure any improvements that could exist by using the headcount optimization model. To do this, one week of output totals by shift from each month in May, June, July, and August was used. For each of these days, all of the input factors were known so the optimization could be run to determine the optimum headcount needed to meet the actual output for the shift. Once computed, the optimization results were compared against actual headcount levels for each shift of each day. With each comparison, a number of key assumptions were made. Each of these assumptions is described in detail below.

The first assumption made when running the optimization model for the comparison was to assume that there was a 50% chance that predicted output of Build

meets the actual output for the factory over the days discussed. Over the 4 weeks, the average output level of the actual and predicted values should be equivalent.

In order to ensure that Kitting never starves Build, the assumption was made that Kitting would outperform Build 70% of the time. Assuming a queue is built up before Build starts in the morning and Kitting outperforms Build 70% of the time, it was inferred that the queue between Kitting and Build would never be depleted enough that Build will be starved through the day.

In order to ensure that Boxing never blocks Build, the assumption was made that Boxing would outperform Build 70% of the time as well. Assuming all units after Build were cleared before Build started in the morning and Boxing outperforms Build 70% of the time, it was inferred that there would not be sufficient buildup of inventory after Build that would prevent Build from moving units out of their area.

The Boxing area optimization is unique in that Block and Starve time is needed in its area model to predict output from the Boxing area. By optimizing headcount in the Kitting area, the Build area and the Boxing area, the amount of block and starve time should decrease in the factory. The Kitting area should no longer starve the Build area, which in turn should not starve the Boxing area of material. That is important for two reasons. First, the speed of the factory is assumed to be equal to be the speed of Build. Second, because the factory is optimized, we can be more confident that the Kitting area will not starve the Build area, and the Boxing area will not block it.

Recalling the block and starve data for the factory, the Kitting area was starving the Build area 12% of the day (Build Starve Time). This assumption also ignores the amount of time that the Kitting area gets starved, because it is such a small fraction of time that even if starved, the kitting area will work fast enough to rebuild up the queues before the Build area could diminish them. In turn, the Build area 32% of the day starves the Boxing area. Because the Kitting area can be controlled to not starve boxing, the model can eliminate that 12% Build area starve time. In turn, the Build area will not starve Boxing for that 12% of the time as well. For the model results, the reasonable assumption was made that Boxing would be starved 20% of the day, which was equivalent to the original 32% starve time minus the 12% Kitting starve time.

Since this model does not impact the amount of time that Boxing is blocked by distribution, the assumption was made that the amount of blocked time in Boxing will

remain the same at 4%. With these assumptions, the actual headcount versus the predicted headcount for the 4 weeks of output was compared. Their results are discussed below.

5.1 Test Results

The shift-by-shift results for the optimization headcount compared to actual headcount are shown in the Appendix. In general, there would have been a pretty substantial drop in headcount needed to meet the output demand for these weeks if the optimized headcount levels had been used. The results are shown in the following Figure.

OPTIMIZATION STATISTICS

209	<- People Savings First Shift
323	<- People Savings Second Shift
9	<-Hours in Shift
4788	<- Total Hours
\$8.50	<-Cost Per Hour
\$40,698.00	<- 1 month Savings
\$488,376.00	<- Yearly Savings

Assumptions:	
50% Probability of Build Making Output	
70% Probability of Kitting Outperforming Build	
70% Probability of Boxing Outperforming Build	
4% Block Time in Boxing	
20% Starve Time in Boxing	

Table 11 - Cash savings of optimized versus actual headcount levels

With a conservative estimate from the human resources on employee salaries, the average reduced employee hours has an average salary of \$8.50 per hour³, an average monthly savings of \$40,698 and a yearly substantial headcount savings of \$488,376 could be obtained by using the optimized headcount plan. Additional savings could be inferred if the cost of employee benefits were included in the cost per hour statistic above. The average cost per hour for benefits averaged \$3 within TMC. Factoring in these savings as

Human Resource Statistics – Shift 1 and Shift 2 permanent salary averages are \$9.33 and \$9.15 per hour respectively (not including benefits). Assuming no shrinking headcount. Shift 1 and Shift 2 temporary salary averages are \$8.50 and \$9.35 per hour respectively.

well, the yearly potential savings were \$660,744. This analysis is included in the Appendix. While the results are good, the regression models could be better. Below are suggestions that would improve the prediction.

5.2 Hard Constraint Analysis Recommendations

While the results above showed dramatic cost savings had the headcount management tool been used, there are some key recommendations that could make the headcount prediction more accurate. Some recommendations to make the hard constraint analysis more accurate are included below.

5.2.1 Kitting-to-Build Recirc Recommendations

There are two components that are needed to predict throughput of the Kitting-to-Build Recirc more accurately. First, the percent downtime of the Recirc would be helpful and could be obtained by measuring the amount of time that the Recirc is actually running in a given hour. Downtime will impact the maximum throughput rate of the Recirc and should be tracked accurately. A second recommendation is to track the total number of tote rescans that occur on the Recirc by counting the number of totes that the production conveyer system did not push to a Build line while the Build line was not completely full. This happens from time to time and requires knowing which line the tote should be pushed to. Each unit that is not pushed to a Build line drops the capacity by 2 units - one for the unit that missed its line, and one for the tote that is unable to get on the Recirc because the slot is full. Knowing these two key bits of data will help managers determine the maximum capacity of the Recircs and the overall capacity achieved per hour as well.

5.2.2 Burn Rack Recommendations

At times, the burn racks have been thought to be the constraint of the factory. While this thesis analysis has shown that the burn racks are not currently the constraint, from the max capacity estimated, the burn racks could slowly become a possible constraint as the production capacity of Build, Kitting, and Boxing increases. However, no real measurement of the burn rack capacity exists. To track the Burn rack capacity, the following could be done.

First it is important to know the number of Burn Cells that can be used every hour. This will be important to determine the maximum capacity of the Burn area, as

non-useable burn cells lower the maximum throughput. By coupling the total number of burn racks with information about the average pre-test, test, and post-test times every hour, the Burn capacity can be calculated, as it was in the Hard constraints section. Knowing the Burn capacity will be important to be sure that the Burn area is not becoming the factory constraint. Another important measurement that could be collected is the time that individual computer systems move in and out of the Burn area. Currently, the data is averaged every hour, but by understanding the individual times for each computer, the variation that may exist in the Burn area can be measured as well. This will also give a more realistic estimate of how often Burn may be the constraint in the factory.

One final suggestion in the Burn area is that Dell may want to track pre- and post-dwell times in each quad for each hour and holding the associate responsible for that quad more accountable for quad performance. One method of tracking burn rack status would be to look at the Real Time Monitor Board slot-tracking screen shown in the Figure below.

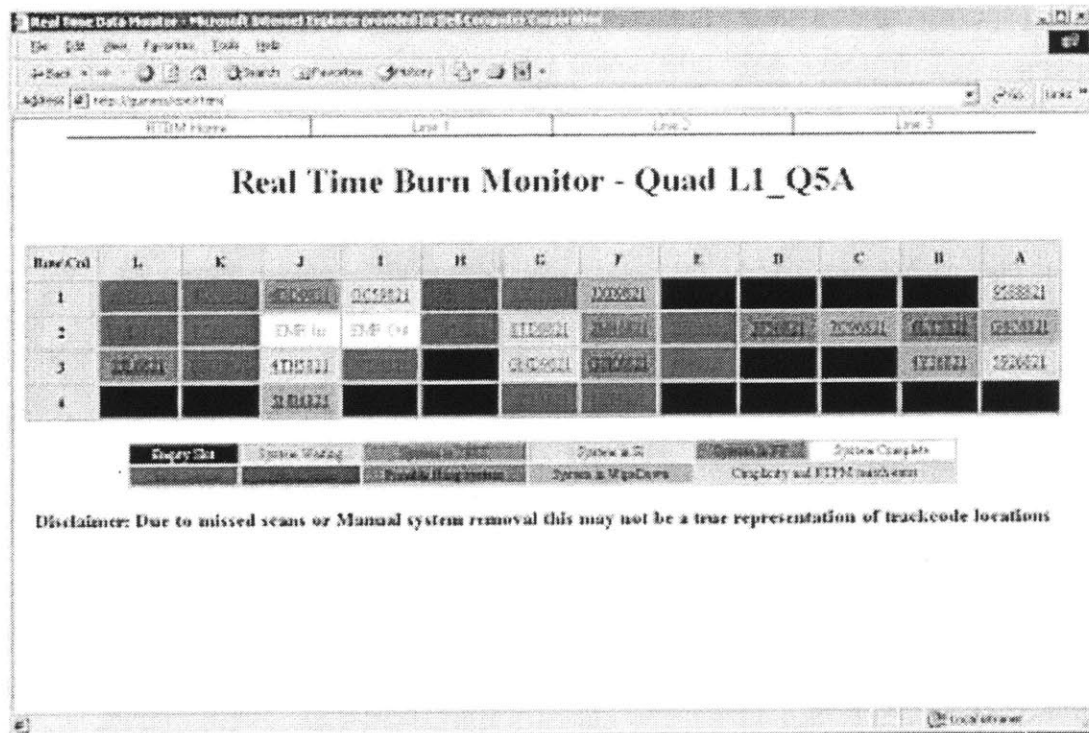


Figure 22 - Real Time Burn Monitor Screen

Every burn slot status is tracked in this system real time. It would be good to track the cumulative times of the different states, such as the total amount of time the computers are awaiting the start of the Burn process, of all of the slots in the quad each hour. This would help isolate if an associate is overworked by measuring the throughput rate in the area, or simply by counting the number of slots per hour that are waiting for the Burn process to start or the associate pushing the computer on to the next process step. Additionally, it would help determine how Burn errors affect throughput at the burn racks. If the Burn sequence has to be restarted often due to downloading errors, the Burn area will not be utilized as well as it should be. Finally, the monitor could help identify if there is a capacity shortage in the Burn area. If inventory is waiting to get into the burn area, and the slots are all filled with computers actively burning software, then the total number of Burn slots is a constraint.

5.2.3 Boxing-to-Build Recirc Recommendations

There are two components that are needed to predict throughput of the Build-to-Boxing Recirc more accurately. In general, the estimate for throughput on the Boxing-to-Build Recirc is hard to estimate. It appears that when downtime is factored in, the theoretical and estimated capacity is close enough to the maximum Build area output in the factory, that this may be a hard constraint in the factory shortly. This should be validated. Additionally, as in the Kitting-to-Build Recirc, data should be available to measure downtime for the Recirc and computer totes that have to re-circulate through the Build-to-Boxing Recirc as well.

5.3 Capacity Data Collection Recommendations

While each factory area's capacity estimates are fairly good for this factory, there are some ways to make the prediction equations much more accurate. Some descriptions for some suggestions to improve capacity modeling at TMC are included below.

5.3.1 Track temporary headcount placement and movement in the factory

It is very important to track temporary headcount by area each day. Currently, it is extremely difficult to accurately predict how many people are working in each area. The temporary headcount numbers are forecasted totals that are assigned to different areas, however, the workers are not necessarily working in the forecasted area. When the workers arrive on site, the management team dispositions the workers to the areas in the

factory that need them the most. Tracking placement and movement of employees is important, because without knowing accurate number of workers in each area, productivity measurements would never be fully accurate.

A secondary affect of not knowing where the temporary headcount resides in the factory is that correlation of safety, productivity or quality by percentage of temporary workers in each area is not possible. The correlations that were run based on these estimates were not accurate enough to pull meaningful trends out of. If more detailed tracking of temporary headcount were done, it would improve those correlations.

5.3.2 Define Block and Starve Metrics in the Factory

Incentives should be created that measure the Boxing area and the Kitting area on how often they block or starve the Build area respectfully. Since Build is the constraint, every minute that it is down is a wasted minute of production capacity. If the factory is designed to feed the constraint, all other areas in the factory should guarantee that they do not impede the constraint area, regardless of whether that rate is above or below the factory output goal. To estimate the block goal for the Kitting area the following formula can be used.

$$(32) \quad \textbf{Kitting Block Time} = 1 - \textbf{Build Output} / \textbf{Kitting Adjusted Capacity}$$

This estimates the percent excess capacity that Kitting area has, and determines the amount of downtime the area could expect because the slower Build area blocks the Kitting area. For example, if Kitting has a predicted capacity of 14,000 units per day and Build has an output level of 13,000 units, the expected downtime as calculated with Equation 32, will be as follows.

$$\begin{aligned} \textbf{Kitting Block Time} &= 1 - 13,000/14,000 \\ &= 1 - 0.9286 \\ &= 0.0714 \\ &= 7.14\% \end{aligned}$$

Likewise, to estimate the starve time for the Boxing area the same estimate can be calculated for Boxing Starve time as was done for the Kitting Block time above. The following formula can be used.

(33) Boxing Starve Time = 1 – Build Output/Boxing Adjusted Capacity

By calculating the expected block and starve times, now the Kitting and Boxing areas have an incentive to work at a pace that increases the amount of time they are outperforming the Build area, and ensures that build will be filled to capacity.

5.3.3 Use Monitors to Display Block and Starve Time against Goal

As part of this thesis, a new real time monitor was created to help track and the accumulated time for each shift that the Kitting area was blocked by the Build area and could not push material onto the Kitting-to-Build Recirc. Through a display in the factory, this information could then be displayed to the factory workers real time. Likewise, a monitor was created to measure the amount of time each shift that the Build area starved the Boxing area so there would not be a buildup after the Build-to-Boxing Recirc of computer systems waiting to be boxed. This information could be displayed real time on the factory floor as well. These monitors were already designed and the placement of these monitors can be seen in the graphic below.

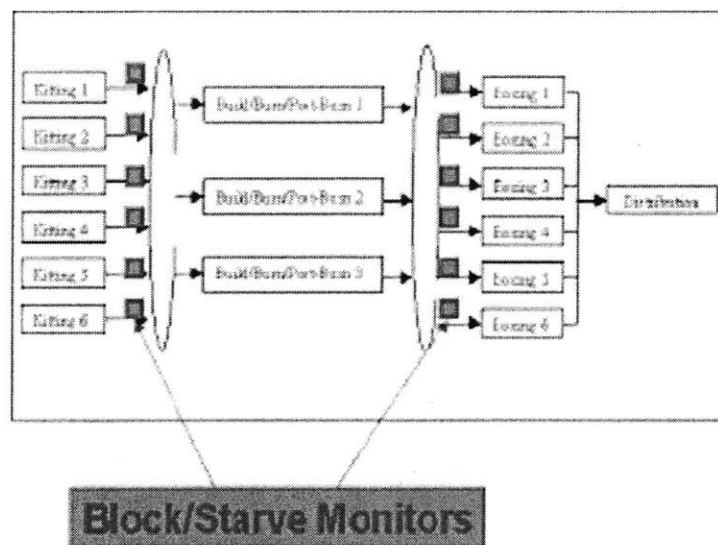


Figure 23 - Placement of Block and Starve Monitors in the Factory

By placing the monitors before the Recirc, the motivation for the employees in the Kitting area would be to not just prevent the Build area to be starved every minute, but actually to build a large queue of inventory up in front of the Build area. This buffer would insure that Build always has totes available for assembly when they are needed despite fluctuations between the Kitting and Build area production areas. Likewise, monitors would be placed after the Build-to-Boxing Recirc, and would measure the amount of time that Boxing is starved. The motivation for the Boxing area would be to clear the Build-to-Boxing Recirc completely of material, which would ensure that Build will always be able to push out as much material as possible. By reducing the block and starve time in Build, it will be easier to make better predictions of output in Build, because the output would measure only the output levels in Build, unaffected by the block and starve potential that existed in the old headcount model.

One additional side effect of putting up monitors is that the employees on the line would see the performance goal for the shift, and can set their speed in order to beat their target. As mentioned, the real time monitor has been created already and the display for Kitting Line 4 can be seen in the Figure below.

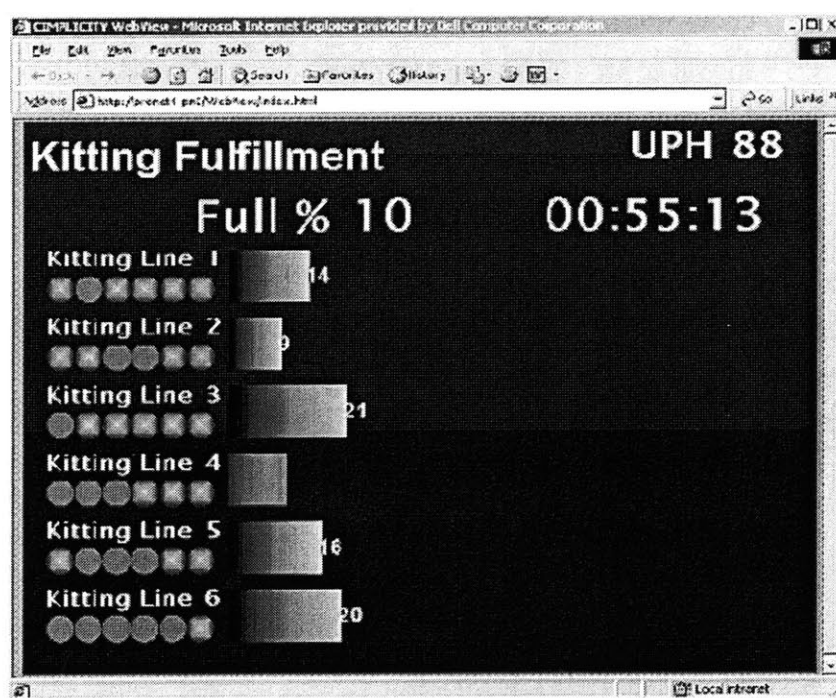


Figure 24 - Kitting Line 4 Real Time Block Monitor

For the above monitor, the Recirc was setup to ensure that Kitting Line 4 had specific slots that it could push out material to, and it was also setup to feed Build Line 2 only. As Kitting Line 4 outpaced the Build Line 4, and completely filled its slots on the Kitting-to-Build Recirc, Kitting Line 4 became blocked, and started to accumulate blocked time for the shift represented by “Full %” in the monitor. For the monitor in the display, the line was shown to be blocked 10% of the time by Build, and was pushing out totes at a rate of 88 units per hour. The other Kitting lines are displayed as well for the Kitting Associates to see how the other lines are doing. Finally, the monitor also shows if there is material backed up in each of the Kitting Lines by showing the number of photo-eyes that are covered before the Kitting-to-Build Recirc. Kitting Line 4 in the monitor is showing that there are currently three totes, shown as bright dots, lined up that are waiting to be pushed onto the Kitting-to-Build Recirc.

It is important to note that as the factory begins “water falling,” or sending material from any Kitting line to any Build line, a second benefit can be seen. In order for any one Kitting line to be blocked by Build, the Build line queues and the Kitting-to-Build Recirc must be completely filled. This, in turn means that every Kitting line will be blocked at the same time. For this reason, the Kitting lines will need to work together to build a queue large enough that no material can be moved out of any of the Kitting lines. Under the old system, each Kitting line was measured according to the number of units per hour it put out. It was possible for one Kitting line to steal output capacity from another line. By being the fastest to fill the Recirc, one line can have a high unit per hour count, while another line has a low unit per hour count because there is no space to push material onto the Recirc. Instead, measuring by the percent blocked time in Kitting as well as “waterfalling” between the Kitting and Build Lines, it forces all the lines to work together to insure that the blocked percentage target is met.

The same methodology for tracking Kitting area blocked time also applies to tracking the Boxing area starved time. The monitors display the same basic information as the Kitting area displays, however instead of blocked time for each line, starved time is tracked. Like in the Kitting area, when “waterfalling” between the Build and Boxing areas starts, all lines will need to work together to ensure that the Boxing lines remain starved by the Build area.

5.3.4 Start Kitting and Boxing 15 minutes earlier than Build

In order to ensure that a queue is entirely built up before the Build area and all computer systems awaiting Boxing are purged from the line after the Build area, it is important to start the Kitting and Boxing areas early. By starting these areas early, a buffer can be built up that will ensure Build will have material available if there is a large fluctuation of output levels in the Kitting. Additionally, it will ensure that there is no material that would prevent Build from pushing out computer systems after they are assembled. After a month of checking Build area and Kitting area queues after each shift, on average, only one line in Kitting was fully blocked by material awaiting processing within the Build area. Likewise, the Boxing area queues were not always depleted at the end of each shift. By remaining disciplined about building a queue up in front of the Build area, and clearing material after it, Build will be able to move product more freely into and out of its area. By doing this, the factory would be much more predictable, and the constraint will remain in the Build areas. This will make the optimization prediction equations for the areas much more accurate as well, because the factory constraint will not shift as much.

5.3.5 New Downtime Tracking Methodology

Monitoring factory downtime is extremely important when looking at a production facility. Within TMC, factory downtime in meetings was tracked as the downtime in the Build area alone, because it was assumed to be the bottleneck. However, unless the constraint remains in Build, downtime metrics are useless. The constraint may move due to downtime in the different areas, or an imbalance in area capacity that forces the constraint to move to a different area of the factory. To prevent the constraint from moving, it is important to feed the Build area, and ensure that each area has ample resources to recover from downtime and keep from impacting the Build areas production flow.

A second issue with downtime tracking is that collecting data has been a struggle. The problem is that there are so many different types of downtime that are being rolled up into one downtime metric for the factory. Some of the data sources come from production databases, where downtime can be measured directly. Other downtime sources are simply call logs, where measuring downtime is not as easy to quantify. All of these data sources are rolled one downtime metric for the factory. The model presented

in this thesis takes a different approach on downtime. It requires that data be available for throughput, block and starve time and time down in the Build area as metrics to compute an accurate prediction of factory throughput.

When a system goes down in the factory, the downtime's impact on production is tracked in two ways. First, when there is downtime of a hard constraint production engineers need only to know whether its downtime caused the hard constraint to become the factory constraint or not. The second impact would be measuring the impact of downtime in one of the other production areas. In that case, it would be important to determine whether Build was blocked or starved because of the downtime seen in these areas. If Build is blocked or starved, the impact to the factory can be quantified by looking at the hourly output volumes in Build.

5.3.6 Track Hours Worked in the Shift

The number of hours worked in a shift at TMC was not tracked accurately. To estimate the hours worked during a shift for the optimization, the hours in each shift had to be counted that had output near average amounts for that shift. If a final hour or beginning hour in the shift actually had output values near $\frac{1}{2}$ the predicted value, the schedule was assumed to have a $\frac{1}{2}$ hour shift in the start or end time. Because of these estimates, the regression correlations could be off.

For this reason, a system should be setup that tracks shift start, shift end, and the time workers move on and off of the factory floor for breaks, lunch, or training. A new program in the factory was implemented in the factory that had supervisors enter the start and stop time of their employees manually, but a system like that relies on accurate recording by the manager. Because it relies on human estimation, and does not account for breaks, training, or all hands meetings even that downtime tracking was inaccurate. An automated system of recording the hours that are worked on the factory would be nice. One suggestion to track start or stop time of each shift might be to track a piece of reliable machinery such as a Recirc or conveyer system that constantly runs during each shift. This might give a good estimate of shift start and stop times by looking at when the machinery is stopped or started. While individual breaks would not be recorded still, this is at least a more reliable method for tracking shift start and stop time than the current process of manually recording of the times.

5.3.7 Track Headcount in Each Area

The regression analysis for the optimization relies on knowledge about how many employees are working in each area on a given day. As mentioned, this information was estimated in the headcount forecasts created each week, however workers get moved around in the factory quite a bit. It would be nice to record where headcount is in the factory at every instance. One suggestion might be a badge reader using a technology similar to Radio Frequency Identification that could be placed at stations throughout the factory and could record that movement of each employee. This would let the managers have insight into where their employees are at any time. While Dell would incur costs as it first sets up this technology, true headcount predictions are hard to determine without tracking where and when people are working in the factory. This is important, because as variability exists in the model, extra headcount is needed to ensure that production output is met. As the variability in the model is reduced, less of a headcount buffer is needed to ensure production output is met. The dollar savings could be large. Another benefit is that an employee tracking system would also give management insight into how much associate time is spent in training, or off the shop floor.

Another suggestion to improve headcount tracking on the floor is that Dell should track the placement and movement of their temporary employees. While the headcount forecasts predict the amount of temporary headcount needed in each factory area, many times temporary headcount comes in under one manager and is distributed differently throughout the factory. Additionally, in the headcount tracking spreadsheet, all of the temporary employees are actually recorded under one cost center. To ensure that the temporary employees can be tracked, separating the temporary employees into different cost centers and automatically tracking where temporary headcount moves throughout the factory during a day would be good. By knowing where headcount is utilized, projections of how many employees are needed in each area, and how issues like quality and safety are impacted by temporary hires in the factory could be much better approximated.

5.3.8 Track Product Mix Variability through Each Line

There is a good deal of variability that is introduced into the model by product mix and it is not tracked by line. The time to kit, build, and box systems is different for every system type, and that impacts the throughput of each area. Information about the product

mix running on each line in the factory exists in the company database, however the complex queries to pull that data out have not been written. If that data could be pulled out, it could be used as a major factor in make correlations much more reliable.

6 INCORPORATE NON-PRODUCTION FACTORS

There are other factors besides productivity that are important to when modeling headcount for a factory. These factors include Quality, Injury Rates, and Employee Retention Rates. While these are indirect factors, they are no less important. For example, if quality is lagging due to workers producing at an extremely high production level, there is some tradeoff between high productivity rates and high defect. These indirect metrics need to be gathered and evaluated. Some comments on each of these factors follow.

6.1 Quality

A regression model was run that correlated the number of defects in the TMC factory to the following factors: defects, total number of shipped PC's, headcount, total number of hours worked per day, and day of the week. Additionally, to account for quality improvements over time due to general quality improvement projects, the model also includes the workweek as a factor in the prediction model. Including the workweek tracks the general quality improvements over time, and assumes that quality defects reduce at a given rate each week. The most reliable way to measure defects is to measure the number of defects found in at electronic test station, which test the computer functionality before it leaves the plant. This data is published as "Defects per Million" or DPM data. Other quality indicators used within Dell, such as "Field Incident Rates" which test the quality of the computer after it leaves the factory, were regressed against production factors as well, but no other defect metric was found to have high correlations with the production environment. The regression performed for the DPM data is described below.

6.1.1 Quality Data Source

The comprehensive list of factors used to correlate the "Defects Per Million" rate with the production environment at TMC is in the following table.

Variable	Description
Total Headcount	The total headcount is a measurement of headcount in the Kitting, Build and Boxing areas
Hours in the Shifts	This is a measure of the total number of hours worked in the day (both shifts). This is one measure of fatigue.
Total Output	This measures the total output from the factory for each day.
Labor Hours	This is measure of total people resources that worked during the day. It is calculated by multiplying the total Hours in the Shifts times the Total Headcount
Percent Overtime	Another measure of fatigue. This measures the total number of overtime hours divided by the total number of hours worked in the day. Overtime is considered any time over 8.5 hours of work during a shift.
Day of the Week	This is the day of the week for the model.
Fiscal Week	This tracks general quality improvements over time. Since individual projects are not tracked, this is meant to approximate the total quality improvement project impacts.
Repairs	This data comes from the Defect per Million data, and counts the repairs that were caught at the end of the line, before shipping to the customer.

Table 12 - Production Factors Used to Predict Quality Defects

6.1.2 Quality Results

Given the input variables above, a multivariate regression was run for the time period of the internship in order to create a prediction model for the impact on quality given factory metrics. The prediction equation for the results is shown below:

6.1.2.1 Quality Parameter Estimates

The regression relating the production factors to the number of quality defects found in the electronic test area, at the end of the production line, was run, and the parameter estimates for the model are included in the Table below.

Parameter Estimates				
Term	Estimate	Std Error	t Ratio	P-Value
Intercept	-14.9282	247.2869	-0.06	0.9520
Sum(Total Output)	0.0083185	0.007164	1.16	0.2494
Sum(Hours in Shifts)	14.743202	9.385728	1.57	0.1206
Sum(Total Headcount)	0.1910467	0.163518	1.17	0.2465
Day of the Week[FRI]	25.973749	21.31228	1.22	0.2269
Day of the Week[MON]	-37.41838	20.02095	-1.87	0.0657
Day of the Week[SAT]	-23.28484	38.29652	-0.61	0.5451
Day of the Week[THU]	17.607038	19.01152	0.93	0.3575
Day of the Week[TUE]	-10.60176	19.45675	-0.54	0.5875
Fiscal Week	-8.57578	1.920941	-4.46	<.0001
Total Output/Labor Hours	178.62303	71.14831	2.51	0.0143
Mean(Percent Overtime)	414.70742	187.7923	2.21	0.0304

Table 13-Prediction Equation for Quality versus Production Factors

6.1.2.2 Observations from the Data

The above prediction equation gives us some insight into our quality defects as they are related to the Production factors. First, there is a good correlation between Fiscal Week and Repair totals. That shows that there is a fairly constant improvement in Repairs caught at electric test from week to week. This trend is an approximation of the weekly improvement in quality due to quality improvement projects that are being implemented in the factory or improvements in general builder skills. The model suggests that the number of defects is actually decreasing at a rate of 9 defects per week.

A second major consideration when looking at builder quality is fatigue. In this model, there are several main factors used to track fatigue. They are composed of output volume, hours in the day, and headcount. Specifically, the number of hours of work per day, total number of labor hours, and the number of units built per laborers in production were found to be significant. General trends indicate that as the total output per headcount grows, the number of defects grow as well. Additionally, as the number of labor hours in production grows, the number of defects grows as well. Finally, the data shows that as headcount grows the defect rate grows as well. This might suggest that the areas are getting more crowded, which causes more defects.

Looking at fatigue in a different way, the Total Output/Labor Hours factor has the second highest correlation estimate for the quality model. This shows that the more output each individual associate works, the more Repairs are found at electronic test. When the average Total Output/Labor Hours is increased by one unit per person per day, defects climb by roughly 179 units for that day. An average output per person per hour is usually 1.0-1.5 units per hour. An example of how this is calculated is shown through the example below. Assume in this example, the following information.

Variable	Value
Total Headcount (Kitting, Build, Box)	1,042 workers
Hours in the Shifts (Daily Work Hours)	17 hours
Total Output	22,998 units

Table 14 - Total Output/Labor Hours Example

To determine the Output/Total Labor Hours, the Total Labor Hours would first need to be calculated. The equation to determine the Total Labor Hours in the factory is included below.

$$(34) \quad \text{Total Labor Hours} = \text{Total Headcount} * \text{Hours in the Shifts}$$

Plugging in the total headcount and the total number of hours for the day into Equation 34 yields the following result.

$$\begin{aligned} \text{Total Labor Hours} &= 1042 \text{ workers} * 17 \text{ hours} \\ &= 17,714 \text{ hours} \end{aligned}$$

Now, the total Output per Labor Hour can be calculated using the following equation.

$$(35) \quad \text{Output/Labor Hours} = \text{Total Output/Total Labor Hours}$$

Using the total labor hours calculated above, and knowing the total output for the factory as given in the example data, the Output per Labor Hour can be calculated.

$$\begin{aligned} \text{Output/Labor Hours} &= 22,996/17,714 \\ &= 1.298 \text{ units} \end{aligned}$$

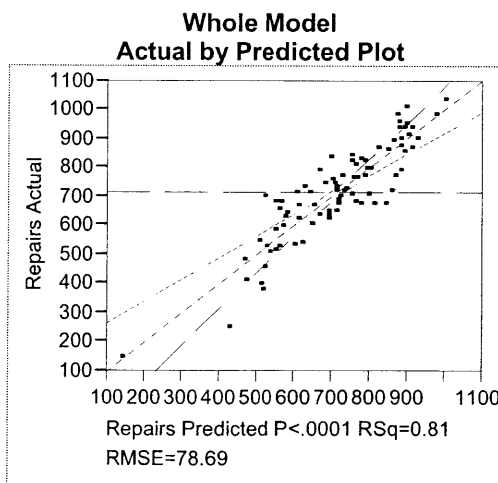
With this calculation in mind, it is important to note the best volume days per hour were measured to be near 2.5 units per hour. With a 1.2 raise in average units per labor

hour, high volume days may increase the amount of repairs by nearly 215 units, which implies there may be a tradeoff between volume and quality defects.

Another observation about the Quality regression is that defects increase with Percent Overtime. This would suggest that for every one percent of overtime added by an associate, quality defects are raised 4.14 units per day. For example, 10% overtime would yield an additional 41.40 additional failures. Similarly, the total number of defects increases with as the Hours in the Day increases. This would suggest that you could expect repairs to increase by 14.74 units per hour as the length of the shifts (i.e. total work hours) increase. Finally, it was important to note that the day of the week has an impact on the quality of the product as well. It is important to also realize that Mondays appear to have the lowest defects of any day with 57 less defects than the baseline case of Wednesday. There is also a trend that the number of defects actually grows throughout the week. Monday has the lowest defect rate, and it grows every day through the week, until Saturday when it drops off again.

6.1.2.3 Measure of Predictability for Quality

With the prediction equation determined, the predicted value was plotted against the actual quality defect rate in the factory. This plot is given in the figure below.



Summary of Fit

Rsquare	0.805869
RSquare Adj	0.776211
Root Mean Square Error	78.68951
Mean of Response	716.9881
Observations (or Sum Wgts)	84

Figure 25 - Impact of Headcount on Quality

The above model shows an R-square value of 80.6%. This suggests that the factors selected can explain a majority, 80.6%, of the variability in the repair rates seen on the TMC factory floor. This would be considered to be a good regression fit for the model.

6.1.2.4 Variation between Predicted and Actual Values

Looking at the residuals below, we see the residuals, or the measure of how far off our predicted defect number was from the actual defect number in the regression analysis.

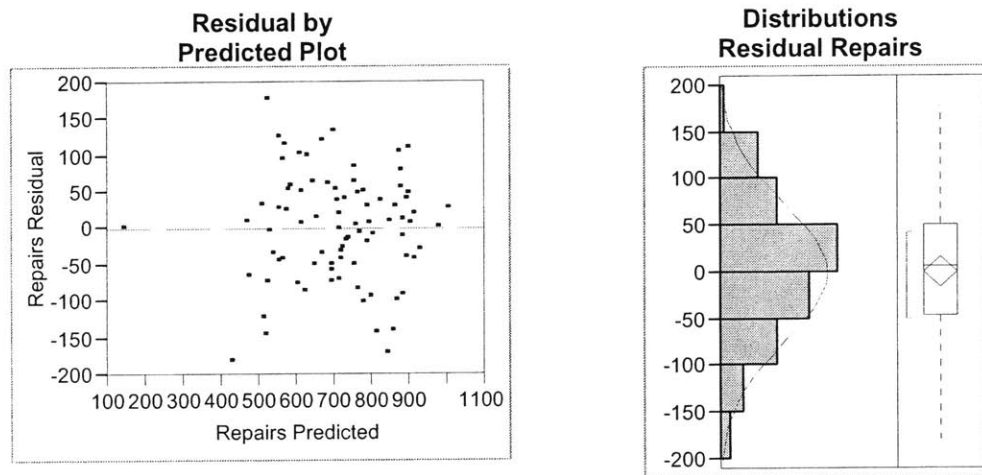


Figure 26 - Residual Plot for Quality Data

From these plots it was seen that the residuals were normally distributed with a mean of new zero, and a standard deviation of 53. There is no trending to suggest a better regression was possible.

6.1.3 Temporary vs. Permanent Employee Impact

A regression to predict quality against the above factors as well as the percentage temporary workers in the factory was run. When running the regression, there were fewer data points to use, so the results were slightly different than the above model. Below shows the same correlation ran on the reduced subset, and a correlation used to measure the impact of temporary headcount on quality.

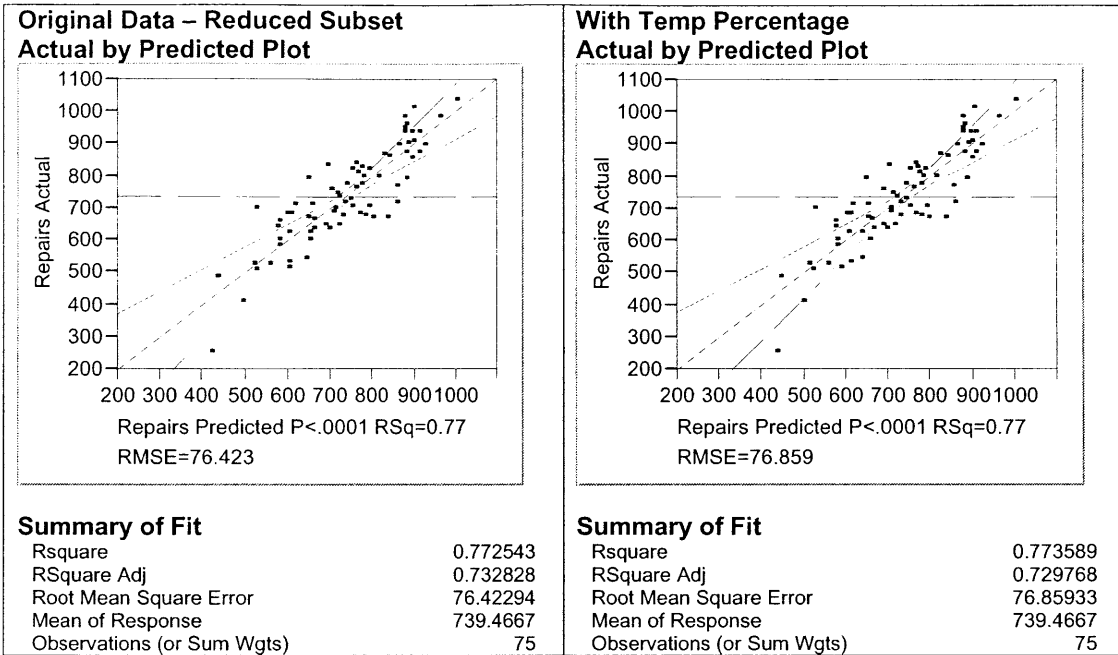


Figure 27 - Quality Regression Analysis with Percent Temporary Worker Included

The above charts show that temporary employee percentage is not a great predictor of Quality. The R-square value was only increased by .001 or .1%, and the R-square Adjusted value actually dropped, implying that the number of temporary employees is not significant. Below is the new parameter estimates for the quality regression with temporary employee percentage as a contributing factor.

Parameter Estimates				
Term	Estimate	Std Error	t Ratio	P-Value
Intercept	708.19642	624.3275	1.13	0.2610
Sum(Total Output)	0.0171003	0.012111	1.41	0.1629
Sum(Hours in Shifts)	-5.758851	20.99542	-0.27	0.7848
Day of the Week[FRI]	30.260959	21.917	1.38	0.1723
Day of the Week[MON]	-46.10433	21.24612	-2.17	0.0338
Day of the Week[SAT]	-32.02617	43.57949	-0.73	0.4652
Day of the Week[THU]	23.737951	20.71364	1.15	0.2562
Day of the Week[TUE]	-12.44247	20.94173	-0.59	0.5546
Fiscal Week	-9.521327	2.87447	-3.31	0.0015
Total Output/Labor Hours	-49.38598	183.2434	-0.27	0.7884
Mean(Percent Overtime)	533.80774	271.8886	1.96	0.0541
Sum(Total Headcount)	-0.082689	0.321202	-0.26	0.7977
Temp Percentage	139.09252	259.804	0.54	0.5943

Table 15 - Predictor Equation for Quality versus Production Factors including Temporary Employee Percentage

This shows similar conclusions to our base model. We find that the Temp Percentage factor, or percentage of temporary employees, is not a significant term, as its

P-Value is greater than 0.15, at 0.5943. This suggests that the percentage of temporary headcount with respect to permanent employees does not matter when quality is being determined. Additionally, we find that instead of Output per Labor Hour and Headcount being significant, the Total Output becomes the main factor. The Fiscal Week and Percent Overtime remain the top predictors of our Repair rate. Because the model will be used to determine Headcount's impact on Quality, the first model is preferred. It tells us how we will impact quality by adjusting headcount.

6.1.4 Limitations of the Results

While the model had a high R-square value showing statistical significance, the model has some shortcomings. The first shortcoming of this model is that the data is rolled up into daily quality metrics, instead of by shift. By tracking the defects found per shift, there will be more data points for an overall model, which would in turn suggest that a stronger correlation could be found. Additionally, quality could be very shift dependent. By tracking the defect rate by shift, it could help to isolate problems that could occur by one shift or the other. With the quality data combined for a daily total, shift trends cannot be determined.

Another improvement that could be implemented that could enhance the quality data could be to track defects by products within the factory. That information exists, but is buried in the factory databases, and would require some complex queries to be written. However, this data would be helpful in isolating build problems by product within the factory.

Another improvement to the quality regression would to design a better way to track quality improvement projects and their impact on the number of defects found on the factory. Currently, it is hard to quantify the impact of specific production floor changes on quality, and so the quality trend is tracked by workweek. Finding a method to measure defect rates by quality improvement projects instead of using average improvement rates would help eliminate some of the variability in the model.

Finally, additional data points would be good for correlations between quality defects and temporary employee percentage as well. The results were inconclusive for the data points that were used for the regression, and more data points could be helpful.

6.2 Retention Rate

One of the objectives of the thesis was to determine if there were a correlation between the retention rate of floor associates and the different conditions on the factory floor. The manufacturing managers were particularly interested in determining whether overtime, work hours, or high production volumes had an impact on the number of employees that left the company. While multiple attempts were made to find some correlation between the floor conditions and the number of employees that quit, no correlation was found.

There are a few factors that certainly impact the accuracy of any prediction model. First, many of the employees were temporary workers. Sustaining a workforce of temporary employees is difficult under most conditions. There is no great bond between temporary workers and the host company, and so there is less motivation for a temporary worker to stay with the company given other opportunities or commitments. Secondly, the event where an employee leaves Dell cannot always be correlated to an immediately preceding event. Employees may leave the company days or even weeks after the triggering event occurs, which adds some additional variability into a retention rate prediction model. Finally, there will always be employees that leave Dell for reasons external to the company itself. Because many employees leave for reasons other than work related issues, extra noise is added into retention rate model that will make the prediction much less accurate.

The above examples represented some examples of why accurate prediction of employee retention rate was not fruitful at the TMC. To better predict retention rate within the factory, more information such as when the employee termination notice was submitted, the employee's reason for quitting, and differentiating between temporary and permanent employees would be helpful in filtering out employee terminations that were not caused by production issues. This data was not available for the time frame tracked for the headcount optimization, and collection of this data would be difficult. It may ultimately be very difficult to find any correlation between employees that quit and production factors at all.

6.3 Injuries

Correlating injuries to headcount is not as easy as creating a regression analysis. Over the period of the internship, the maximum amount of injuries that occurred during a

shift was three. Given the many factors that contribute to cause the injuries, a standard regression analysis was not helpful. More complex models were run on the data as well, including a Poisson regression analysis, however, the output from the regression was not significant. Instead of running a regression model for Injury Rates, only general trends from the data could be found. The results of the trend analysis are included in the following sections.

6.3.1 Injuries Data Source

The data used for injuries came from reports that were accumulated and published including all Injuries as well as Near Misses. Injuries are defined as reported incidents that occur in the factory in which an employee was physically impacted. Injuries include minor cuts, bumps, and scrapes if they are reported. Near Misses includes reported incidents that occur on the floor in which bodily harm was avoided, however, the risk of harm existed. These are tracked to enable employees to prevent injuries on the floor before they happen.

6.3.2 Injuries and Near Miss Results

While prediction equations could not be derived for the number of Injuries and Near Miss incidents that occurred on the floor, some interesting information could be observed. The number of Injury and Near Miss Incidents were compared under three different situations. First, the number of Injury and Near Miss Incidents were compared under conditions of different workloads in the factory. Secondly, the number of Injury and Near Miss Incidents were observed under different overtime scenarios and the trends were documented. Finally, the number of Injuries and Near Miss Incidents were observed in the factory for each day of the week, and the trends were documented. Each of these observations can be seen below.

6.3.2.1 Injuries – Workload Analysis

While these predictions cannot be used to “calculate” expected impact of additional headcount on the factory Injury rates, the following three graphs show some basic trends that are interesting.

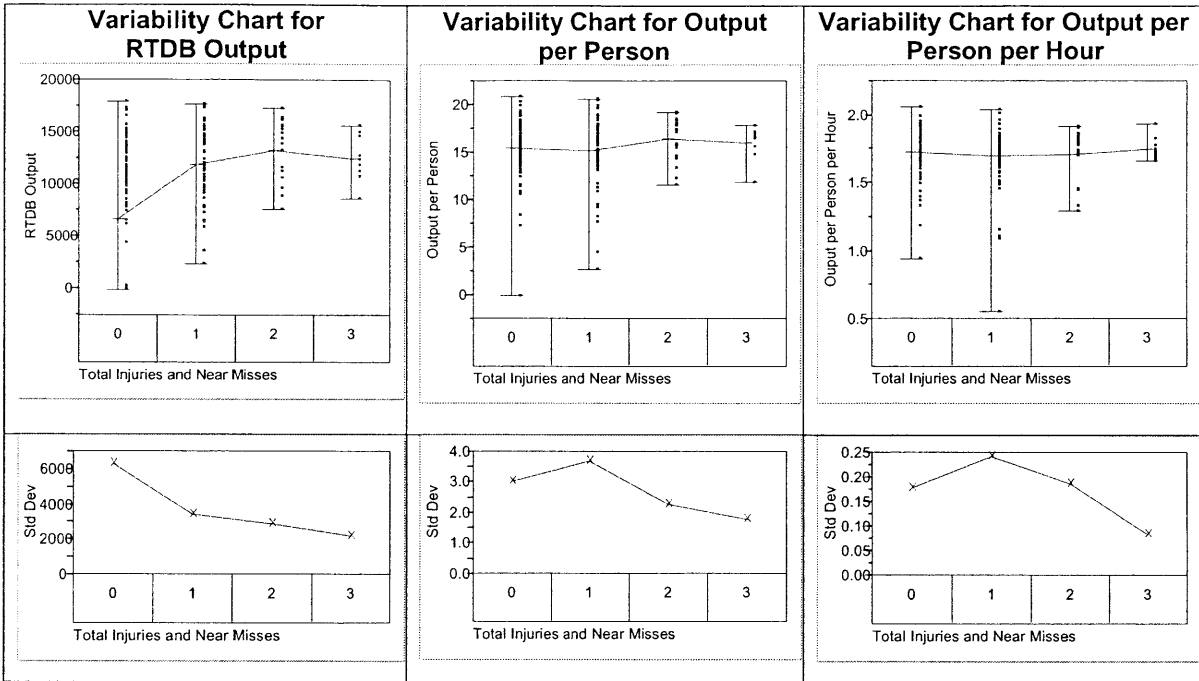


Figure 28 - Mean Injury with different Output Levels

Looking at the number of Injury and Near Miss Incidents for different Production output rates, there are three important trends that can be seen. First, from the graph on the left, it can be seen that as the total output of the factory must grow from an average of 5,000 units to about 10,000 units to increase from zero to one injury in the factory. However, above 10,000 units higher injury totals are easily possible. By increasing the average production in the factory per shift to 12,500 the average Injury and Near Miss incident rate rises to two. This trend seems to indicate that as the production volumes increase, there is an increasing chance of Injury or Near Miss Incidents in the factory.

The middle chart, measures output per person over a shift. The trend remains fairly constant. With slight growth in output per person it is raising the injury rate. The minimum output per person raises from zero for zero injuries, to about 12.5 for 3 injuries. The average output raises from 15 units per person to 15.5 for 3 injuries.

Finally, the output per person per hour trend shows us how injuries are impacted by total output expected each hour. We see the average injuries raise from 0 injuries for 1.75 per hour to 3 injuries when 1.85 units per hour are expected.

While these trends are not directly measurable or predictable, they are telling to the fact that mild fluctuations in Output per Person or Output per Person per Hour can impact the injury rates in the factory.

6.3.2.2 Injuries – Overtime Analysis

Below is a graph that shows the number of injuries with respect to overtime.

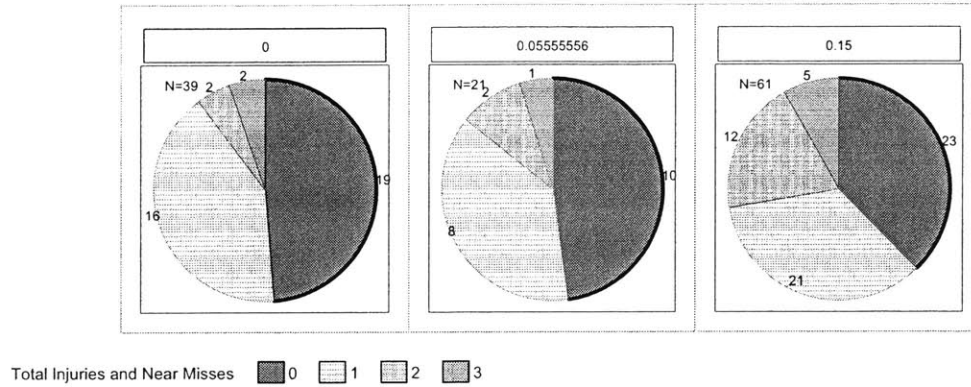


Figure 29 - Injuries vs. Overtime Pie Charts

The charts above show the number of injuries given the percent overtime worked during the shift. The general trend shows that as the percentage of overtime increases, the number of injuries increases as well. The percentage of overtime for the given shift is displayed above each chart. Also, the total number of data points is listed next to the Pie chart. Plotting the data in a different way, the average injury rate given different percentages of overtime as well as the expected shift injury rate is graphed in the following Figure.

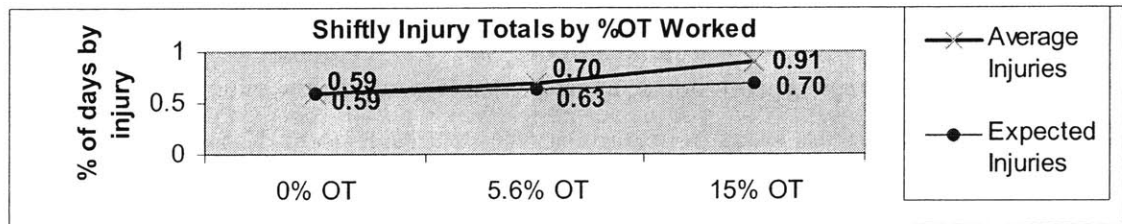


Figure 30 - Expected versus Actual Injury and Near Miss Rate for Different Overtime Rates

Basically, the expected injury rate assumes that the likelihood of an Injury or Near Miss incident within a shift is flat throughout the full shift. The regular work schedule is an 8.5-hour workday. Given an injury average rate of 0.59 with no overtime, the injury rate per hour could be found as in the equation below.

$$\begin{aligned}
 \text{Hourly Injury Rate} &= \text{Avg. Daily Injury and Near Miss Rate/Hours in Shift} \\
 &= 0.59 \text{ injuries /8.5 hours} \\
 &= 0.069 \text{ injuries per hour}
 \end{aligned}$$

Given the hourly injury rate of 0.069 Injury or Near Miss Incidents, a 5.6% overtime, which adds a half hour to the total work time, should have a 0.63 expected Injury or Near Miss rate. Additionally, a 15% overtime rate would add 1.5 hours to the shift. Its expected Injury or Near Miss incident rate would be 0.70. However, the Injury and Near Miss Incident Rate increases at a faster pace, reaching 0.91 at 15% overtime. This suggests that higher overtime yields higher amounts of injuries.

6.3.2.3 Injuries - Day of the Week Analysis

Finally, an attempt was made to track the Injury and Near Miss incident rate by the Day of the Week.

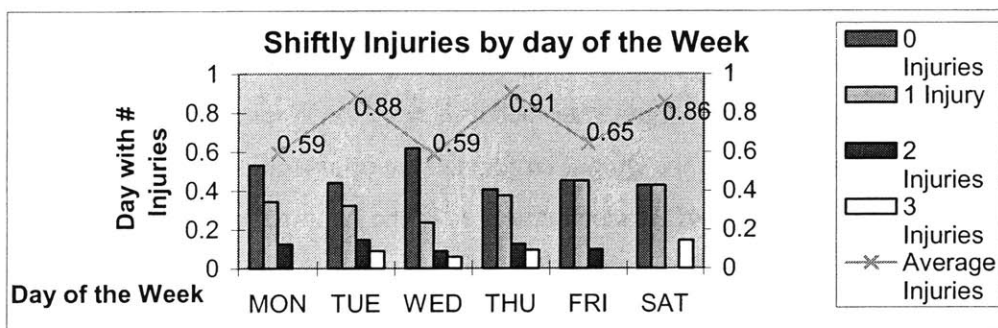


Figure 31 - Injury and Near Miss Rate by Day of the Week

From the above graphs we see that injury rates are highest on Tuesday, Thursday, and Saturdays. There is no discernable pattern that the Injury and Near Miss Incident rate follows, indicating that fatigue that may occur from as the number of consecutive days of work increases is not occurring. Monday and Wednesday appears to be the lowest occurrence of injuries. Thursday has the highest Injury Rate.

7 SUMMARY AND CONCLUSIONS

Dell is in an industry that has very high variability and cyclical demand bursts. Dell's business model, Build to Order and Just-In-Time manufacturing dictates that they will always have variability in its production facilities. The purpose of this thesis was to analyze Dell's production system and create a headcount strategy to help Dell manage bursts of production variability within the Topfer Manufacturing Center.

Through an analysis completed at TMC, three objectives were accomplished. First, an in-depth analysis of the constraints within the factory was completed to help Dell better understand its manufacturing floor. The analysis identified and quantified the hard constraints in the factory, using factory data. Additionally, each of the Kitting, Build and Boxing areas in the factory were analyzed to quantify their capacities, and determine how factors such as Headcount, Day of the Week, Hour of the Day, and Block and Starve time impact productivity within each area. Some key data surfaced from this analysis related to each area, such as productivity trends by Hour and Day of the Week, as well as the impact of breaks within the production area.

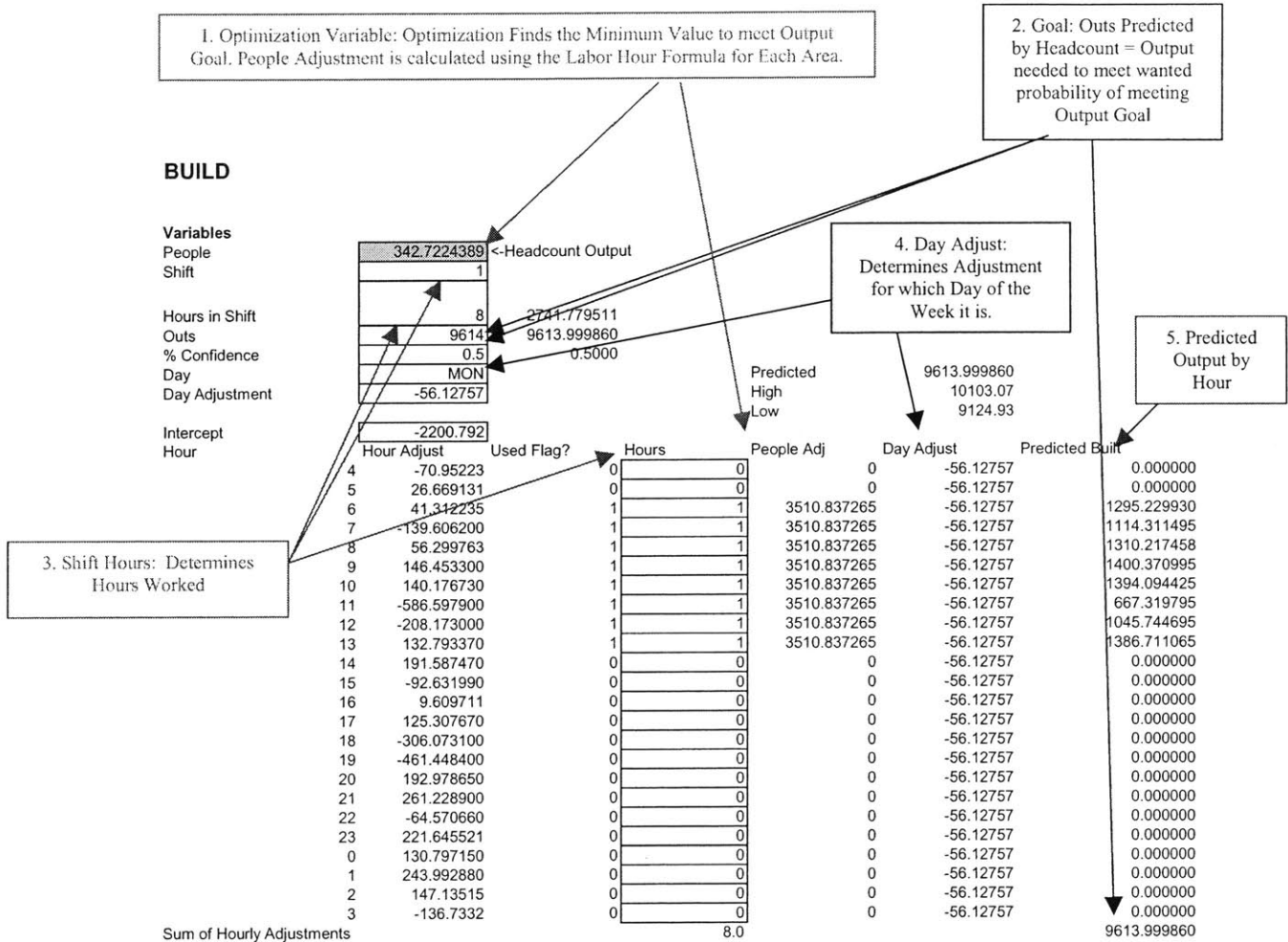
After production capacities were determined for each area, the factory was observed at from a macro-level. The constraint of the factory was identified through analysis of Block and Starve time within the factory. Further, the design of the factory was discussed and some analysis was completed to determine the best area for the constraint to be located in the factory, the Build area. An optimization was then suggested that allowed managers to input data such as Day of the Week, the Shift, the Number of Hours in the Shift, and the Output target, and the optimization would statistically minimize the amount of Headcount needed in the Kitting, Build, and Boxing areas in order to keep the constraint in the Build area and meet the demand of the factory. This model was validated against real production data, and found to reduce the cost of labor in the factory substantially.

Additionally, an analysis was completed to try to correlate non-production outcomes, such as retention rate, quality, or injury rates to production factors. While the correlations were not very strong between production factors and the employee Retention Rate or the Injury and Near Miss incident rate, the Injury and Near Miss incident rate gave some indication that overtime and higher volumes of production had a noticeable impact on Injury and Near Miss incidence rates and that the Day of the

Week could have an impact as well. On a bright note, there was high correlation between Quality, measured in Defects, and production factors such as Labor Hours, Day of the Week, Hour of the Day, and the Workweek.

Finally, recommendations were given with every step of the analysis that would enable Dell to achieve better correlations between productivity factors, such as headcount, and production output, quality, retention rate, and injuries using the data they have already. Additionally, recommendations were presented to acquire new information that might reduce variability of the models as well. While the general correlations presented in this thesis were fairly good predictors of headcount needs and quality in the TMC factory, with more data, and better data data, better models can be achieved. By using the tools and suggestions in this thesis, primarily the optimization model, Dell will be able to react quickly to demand variability bursts that impact their facility. By utilizing the optimal headcount level, Dell can ensure that the factory meets its shift demand and minimize the headcount needed to achieve it.

APPENDIX



Appendix Table 1 - Example of Headcount Optimization Model . The input for this optimization is included in the top box on the left-hand side. The hour adjustments are derived from the prediction equation derived from the regression analysis. Additionally, Solver needs rules to be created in order to bound the possible solution that we are solving for (in this case Headcount). The rules for the optimization are as shown by arrow above.

BUILD

Variables

People	342.7224389	<-Headcount Output
Shift	1	
Hours in Shift	8	2741.779511
Outs	9614	9613.999860
% Confidence	0.5	0.5000
Day	MON	
Day Adjustment	-56.12757	

Predicted	9613.999860
High	10103.07
Low	9124.93

Intercept	-2200.792					
Hour	Hour Adjust	Used Flag?	Hours	People Adj	Day Adjust	Predicted Built
4	-70.95223		0		-56.12757	0.000000
5	26.669131		0		-56.12757	0.000000
6	41.312235		1	3510.837265	-56.12757	1295.229930
7	-139.606200		1	3510.837265	-56.12757	1114.311495
8	56.299763		1	3510.837265	-56.12757	1310.217458
9	146.453300		1	3510.837265	-56.12757	1400.370995
10	140.176730		1	3510.837265	-56.12757	1394.094425
11	-586.597900		1	3510.837265	-56.12757	667.319795
12	-208.173000		1	3510.837265	-56.12757	1045.744695
13	132.793370		1	3510.837265	-56.12757	1386.711065
14	191.587470		0	0	-56.12757	0.000000
15	-92.631990		0	0	-56.12757	0.000000
16	9.609711		0	0	-56.12757	0.000000
17	125.307670		0	0	-56.12757	0.000000
18	-306.073100		0	0	-56.12757	0.000000
19	-461.448400		0	0	-56.12757	0.000000
20	192.978650		0	0	-56.12757	0.000000
21	261.228900		0	0	-56.12757	0.000000
22	-64.570660		0	0	-56.12757	0.000000
23	221.645521		0	0	-56.12757	0.000000
0	130.797150		0	0	-56.12757	0.000000
1	243.992880		0	0	-56.12757	0.000000
2	147.13515		0	0	-56.12757	0.000000
3	-136.7332		0	0	-56.12757	0.000000
Sum of Hourly Adjustments			8.0			9613.999860

Appendix Table 2 - Build Headcount Optimization Model . The input for this optimization is shown in the box in the upper left. The hour adjustment values show the values that come from the prediction equation that we derived from the regression analysis for the Build area. The headcount output is the value derived by the optimization.

KITTING

Variables						
People	78.5903323					
Shift		2 Hours in Kitting	Total Hours			
Hours in Shift	8	628.7226584	4289.981497			
Outs	10857.677148	10194.718519	<-Build Goal			
% Confidence	0.75	0.750000	<-Probability of Beating Build Goal			
				Predicted	10857.677148	
				High	11363.97	
				Low	10351.39	
Intercept	-2477.419					
Hour	Hour Adjust	Used Flag?	Hours	People Adj	Day Adjust	Predicted Built
4	0	0	0	0	53.257286	0.000000
5	-21.939780	0	0	0	53.257286	0.000000
6	269.592730	0	0	0	53.257286	0.000000
7	8.187006	0	0	0	53.257286	0.000000
8	86.295521	0	0	0	53.257286	0.000000
9	330.527750	0	0	0	53.257286	0.000000
10	237.541500	0	0	0	53.257286	0.000000
11	-613.465600	0	0	0	53.257286	0.000000
12	-253.510700	0	0	0	53.257286	0.000000
13	252.952220	0	0	0	53.257286	0.000000
14	10.430631	0	0	0	53.257286	0.000000
15	-378.224000	0	0	0	53.257286	0.000000
16	-124.849700	1	1	3731.746268	53.257286	1182.734854
17	375.598700	1	1	3731.746268	53.257286	1683.183254
18	-142.623500	1	1	3731.746268	53.257286	1164.961054
19	-625.456900	1	1	3731.746268	53.257286	682.127654
20	408.598700	1	1	3731.746268	53.257286	1716.183254
21	365.320920	1	1	3731.746268	53.257286	1672.905474
22	-78.345750	1	1	3731.746268	53.257286	1229.238804
23	218.758243	1	1	3731.746268	53.257286	1526.342797
0	209.973560	0	0	0	53.257286	0.000000
1	65.359689	0	0	0	53.257286	0.000000
2	-600.7254	0	0	0	53.257286	0.000000
3	0	0	0	0	53.257286	0.000000
Sum of Hourly Adjustments			8.0			10857.677148

Appendix Table 3 - Kitting Headcount Optimization Model. The input for this optimization is shown in the upper left hand portion of the above figure. The hourly adjustments show the values that come from the prediction equation that we derived from the regression analysis for the kitting area. The “people” cell represents the headcount needed in the kitting area and is the value derived by the optimization.

BOXING

Variables

People	92.6539672				
Shift	1				
Hours in Shift	8	741.2469338		Predicted	10301.383921
Outs	10301.383921	9614.000000	<Build Goal	High	10827.47
% Confidence	0.75	0.750000	<Probability of Beating Build Goal	Low	9775.30
Starve Percentage	0.3				
Block Percentage	0.05				
Intercept	-1145.928				

Hour	Hour Adjust	Used Flag?	Hours	Block Adj	Starve Adj	People Adj	Day Adjust	Predicted Built	
4	-20.59078		0	0	-22.255025	-112.1784	0	-72.16153	0.000000
5	-11.279670		0	0	-22.255025	-112.1784	0	-72.16153	0.000000
6	111.127030		1	1	-22.255025	-112.1784	2622.320795	-72.16153	1380.924870
7	-56.570660		1	1	-22.255025	-112.1784	2622.320795	-72.16153	1213.227180
8	73.588522		1	1	-22.255025	-112.1784	2622.320795	-72.16153	1343.386362
9	236.596960		1	1	-22.255025	-112.1784	2622.320795	-72.16153	1506.394800
10	253.963430		1	1	-22.255025	-112.1784	2622.320795	-72.16153	1523.761270
11	-589.733800		1	1	-22.255025	-112.1784	2622.320795	-72.16153	680.064040
12	-150.410300		1	1	-22.255025	-112.1784	2622.320795	-72.16153	1119.387540
13	264.440020		1	1	-22.255025	-112.1784	2622.320795	-72.16153	1534.237860
14	186.840290		0	0	-22.255025	-112.1784	0	-72.16153	0.000000
15	-351.908000		0	0	-22.255025	-112.1784	0	-72.16153	0.000000
16	-34.014530		0	0	-22.255025	-112.1784	0	-72.16153	0.000000
17	190.806680		0	0	-22.255025	-112.1784	0	-72.16153	0.000000
18	-307.880600		0	0	-22.255025	-112.1784	0	-72.16153	0.000000
19	-583.113800		0	0	-22.255025	-112.1784	0	-72.16153	0.000000
20	266.359280		0	0	-22.255025	-112.1784	0	-72.16153	0.000000
21	264.912440		0	0	-22.255025	-112.1784	0	-72.16153	0.000000
22	-143.765000		0	0	-22.255025	-112.1784	0	-72.16153	0.000000
23	225.230957		0	0	-22.255025	-112.1784	0	-72.16153	0.000000
0	91.597076		0	0	-22.255025	-112.1784	0	-72.16153	0.000000
1	259.526530		0	0	-22.255025	-112.1784	0	-72.16153	0.000000
2	193.32873		0	0	-22.255025	-112.1784	0	-72.16153	0.000000
3	-369.0458		0	0	-22.255025	-112.1784	0	-72.16153	0.000000
Sum of Hourly Adjustments			8.0						10301.383921

Appendix Table 4 - Boxing Headcount Optimization Model. The input for this optimization is shown in box at the upper left hand of the Figure. The hourly adjustment values show the values that come from the prediction equation that we derived from the regression analysis for the Boxing area. The headcount needed in Boxing is the value derived by the optimization.

PRODUCTION OPTIMIZATION

Factory Outs	Kitting	Build	Boxing	Total People
People	70.15	342.72	92.66	505.53
Outs - Mean	10276.96	9614.00	10301.38	
Outs - Std Dev	700.00	690.00	750.00	
Confidence Interv	0.75	0.5	0.75	
Block			0.05	
Starve			0.3	

Output	10000
Shift	1
Day of Week	MON
Hours in Shift	8

Appendix Table 5 - Production Headcount Optimization Model. The input for the overall headcount optimization is shown in grey above. Given the varying input values, the Solver will minimize the headcount needed in the factory to meet the output demand and keep the constraint in Build.

Calendar Date	Day of the Week	Shift	Mean (Total Build People)	RTDB Total	Hours Worked	Predicted Total	High Total	Low Prediction	Predicted - Actual	Shift Std Deviation	Covariance	Total	Upper Limit	Lower Limit	Within 1 Std Dev	Above Predicted
06/06/2002	THU	First	400	14626	10	14652	15228	14077	26	482.5635709	230	712.5636	15364.56	13939.44	1	1
06/06/2002	THU	Second	369	14270	10	14431	15006	13855	161	482.5635709	230	712.5636	15143.56	13718.44	1	1
06/07/2002	FRI	First	384	14457	10	13854	14429	13278	-603	482.5635709	230	712.5636	14566.56	13141.44	1	0
06/07/2002	FRI	Second	364	13165	10	13965	14541	13389	800	482.5635709	230	712.5636	14677.56	13252.44	0	1
06/08/2002	SAT	First	349	9109	7	8666	9148	8185	-443	403.7416501	230	633.7417	9299.74	8032.258	1	0
06/08/2002	SAT	Second	338	8714	7	8849	9331	8387	135	403.7416501	230	633.7417	9482.74	8215.258	1	1
06/10/2002	MON	First	371	11672	8.5	10911	11442	10381	-761	444.9016296	230	674.9016	11585.9	10236.1	0	0
06/10/2002	MON	Second	382	11418	8.5	11617	12147	11086	199	444.9016296	230	674.9016	12291.9	10942.1	1	1
06/11/2002	TUE	First	384	11639	8.5	11785	12315	11254	146	444.9016296	230	674.9016	12459.9	11110.1	1	1
06/11/2002	TUE	Second	396	12467	8.5	12524	13055	11993	57	444.9016296	230	674.9016	13198.9	11849.1	1	1
06/12/2002	WED	First	384	12576	8.5	11790	12320	11259	-786	444.9016296	230	674.9016	12464.9	11115.1	0	0
06/12/2002	WED	Second	390	13007	8.5	12372	12902	11841	-635	444.9016296	230	674.9016	13046.9	11697.1	1	0
06/13/2002	THU	First	383	11438	8.5	12010	12540	11479	572	444.9016296	230	674.9016	12684.9	11335.1	1	1
06/13/2002	THU	Second	398	13600	9	13517	14063	12971	-83	457.8	230	687.8	14204.8	12829.2	1	0
06/14/2002	FRI	First	379	12344	8.5	11612	12143	11081	-732	444.9016296	230	674.9016	12286.9	10937.1	0	0
06/14/2002	FRI	Second	374	11551	8	11909	12440	11379	358	431.6179792	230	661.618	12570.6	11247.38	1	1
06/17/2002	MON	First	390	10178	8.5	11399	11929	10868	1221	444.9016296	230	674.9016	12073.9	10724.1	0	1
06/17/2002	MON	Second	381	10623	8.5	11591	12122	11060	968	444.9016296	230	674.9016	12265.9	10916.1	0	1
06/18/2002	TUE	First	416	14041	10	14875	15451	14300	834	482.5635709	230	712.5636	15587.56	14162.44	0	1
06/18/2002	TUE	Second	399	14972	10	15015	15590	14439	43	482.5635709	230	712.5636	15727.56	14302.44	1	1
06/19/2002	WED	First	400	15331	10	14380	14956	13804	-951	482.5635709	230	712.5636	15092.56	13667.44	0	0
06/19/2002	WED	Second	393	14513	10	14842	15418	14267	329	482.5635709	230	712.5636	15554.56	14129.44	1	1
06/20/2002	THU	First	389	14615	10	14326	14901	13750	-289	482.5635709	230	712.5636	15038.56	13613.44	1	0
06/20/2002	THU	Second	395	15195	10	15174	15749	14598	-21	482.5635709	230	712.5636	15886.56	14461.44	1	0
06/21/2002	FRI	First	378	11534	8.5	11586	12117	11056	52	444.9016296	230	674.9016	12260.9	10911.1	1	1
06/21/2002	FRI	Second	382	11313	9	12801	13346	12254	1488	457.8	230	687.8	13488.8	12113.2	0	1
06/24/2002	MON	First	395	13391	10	13622	14197	13046	231	482.5635709	230	712.5636	14334.56	12909.44	1	1
06/26/2002	WED	First	412	11559	9	13237	13783	12691	1678	457.8	230	687.8	13924.8	12549.2	0	1
07/01/2002	MON	Second	409	16563	11.5	16931	17548	16313	368	517.491777	230	747.4918	17678.4	16183.51	1	1
07/02/2002	TUE	First	420	17394	11.5	17244	17861	16627	-150	517.491777	230	747.4918	17991.4	16496.51	1	0
07/02/2002	TUE	Second	398	17678	11.5	17251	17868	16634	-427	517.491777	230	747.4918	17998.4	16503.51	1	0
07/03/2002	WED	First	415	17971	11.5	17058	17676	16441	-913	517.491777	230	747.4918	17805.4	16310.51	0	0
07/03/2002	WED	Second	403	17039	11	16783	17387	16180	-256	506.116943	230	736.1169	17519.1	16046.88	1	0
07/08/2002	MON	First	404	13883	10	13894	14470	13319	11	482.5635709	230	712.5636	14606.56	13181.44	1	1
07/08/2002	MON	Second	405	15225	10	14596	15172	14020	-629	482.5635709	230	712.5636	15308.56	13883.44	1	0
07/09/2002	TUE	First	403	15685	10	14465	15041	13890	-1220	482.5635709	230	712.5636	15177.56	13752.44	0	0

07/09/2002	TUE	Second	414	15347	10	15482	16058	14907	135	482.5635709	230	712.5636	16194.56	14769.44	1	1
07/10/2002	WED	First	409	16006	10	14658	15234	14083	-1348	482.5635709	230	712.5636	15370.56	13945.44	0	0
07/10/2002	WED	Second	420	15400	10	15685	16261	15110	285	482.5635709	230	712.5636	16397.56	14972.44	1	1
07/11/2002	THU	First	397	14494	10	14562	15137	13986	68	482.5635709	230	712.5636	15274.56	13849.44	1	1
07/12/2002	FRI	First	398	14512	10	14264	14840	13689	-248	482.5635709	230	712.5636	14976.56	13551.44	1	0
07/15/2002	MON	Second	416	12473	8.5	12534	13065	12004	61	444.9016296	230	674.9016	13208.9	11859.1	1	1
07/16/2002	TUE	First	424	12410	9	13586	14042	13130	1176	457.8	230	687.8	14273.8	12898.2	0	1
8/5/2002	MON	First	327	7391	7	7816			425	403.7416501	230	633.7417	8449.74	7182.258	1	1
8/5/2002	MON	Second	326	9816	8.5	10200			384	444.9016296	230	674.9016	10874.9	9525.098	1	1
8/6/2002	TUE	First	342	11367	8.5	10725			-642	444.9016296	230	674.9016	11399.9	10050.1	1	0
8/6/2002	TUE	Second	325	11640	8.5	10716			-924	444.9016296	230	674.9016	11390.9	10041.1	0	0
8/7/2002	WED	First	354	11167	8.5	11032			-135	444.9016296	230	674.9016	11706.9	10357.1	1	0
8/7/2002	WED	Second	345	11829	8.5	11231			-598	444.9016296	230	674.9016	11905.9	10556.1	1	0
8/8/2002	THU	First	341	12248	8.5	10950			-1298	444.9016296	230	674.9016	11624.9	10275.1	0	0
8/8/2002	THU	Second	324	12099	8.5	10941			-1158	444.9016296	230	674.9016	11615.9	10266.1	0	0
8/9/2002	FRI	First	336	13636	10	12528			-1108	482.5635709	230	712.5636	13240.56	11815.44	0	0
8/9/2002	FRI	First	310	13848	10	12465			-1383	482.5635709	230	712.5636	13177.56	11752.44	0	0
												694.6364			1 STD	>Pred
															34	27
															0.693878	0.55102

Appendix Table 6 - Build Data Validation (Average and Standard Deviation). A value of 1 was given in the "Within 1 Std Dev" column if the value was within 1 standard deviation of the predicted value.

Also, the total number of values above the predicted value was counted as well. There should be roughly 68% of the values falling within +/- 1 standard deviation, and about 50% of the values falling above the predicted value.

Actual HC	Predicted HC	Shift	Output	Shift Length
516	525	1	13326	10
500	407	1	11072	10
501	454	1	9664	8.5
498	546	1	11445	8.5
511	535	1	11397	8.5
2526	2467	-59	<-1 week people savings	
434	371	1	9202	9
424	457	1	9708	8.5
475	360	1	7841	8
470	432	1	9789	8.5
469	486	1	10702	8.5
456	504	1	10724	8.5
2728	2610	-118	<-1 week people savings	
576	460	1	11726	10
599	583	1	14155	10
603	589	1	14416	10
596	598	1	14626	10
580	602	1	14457	10
534	551	1	9109	7
3488	3383	-105	<-1 week people savings	
603	592	1	13883	10
613	649	1	15685	10
618	655	1	16006	10
605	591	1	14494	10
609	604	1	14512	10
574	604	1	8640	6
3622	3695	73	<-1 week people savings	
12364	12155	209	<-5 week people savings	

Date	Actual HC	Predicted HC	Shift	Output	Shift Length
22-Aug	512	526	2	13726	
23-Aug	505	552	2	10798	
26-Aug	511	568	2	10727	
27-Aug	523	527	2	11330	
28-Aug	526	477	2	10719	
	2577	2650	73	<-1 week people savings	
10-May	426	454	2	10916	
13-May	467	355	2	6394	
14-May	485	439	2	9437	
15-May	497	488	2	10892	
16-May	485	446	2	10279	
17-May	471	451	2	10110	
	2831	2633	-198	<-1 week people savings	
3-Jun	582	530	2	13193	
4-Jun	589	527	2	13556	
5-Jun	591	584	2	14681	
6-Jun	578	559	2	14270	
7-Jun	563	507	2	13165	
8-Jun	536	549	2	8714	
	3439	3256	-183	<-1 week people savings	
8-Jul	610	634	2	15225	
9-Jul	624	621	2	15347	
10-Jul	633	617	2	15400	
11-Jul	626	627	2	15673	
12-Jul	620	599	2	16315	
	3113	3098	-15	<-1 week people savings	
	11960	11637	-323	<-5 week people savings	

Table 7 - Validation of the Optimization Headcount Savings in TMC. The optimization's predicted headcount (HC) was compared with actual output figures from each shift for 5 days out of 4 months. The overall savings for the 5 month period was 323 workers.

209	<- People Savings First Shift
323	<- People Savings Second Shift
9	<-Hours in Shift
4788	<- Total Hours
\$11.50	<-Cost Per Hour
\$55,062.00	<- 4 Weeks Savings
\$ 660,744.00	<- Yearly Savings

Setup:	
50% Probability of Build Making Output	
70% Probability of Kitting Outperforming Build	
70% Probability of Boxing Outperforming Build	
4% Block Time in Boxing	
20% Starve Time in Boxing	

**Appendix Table 8 - Optimization Estimated Financial Savings.
Payrate includes cost of salary as well as benefits**

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-y^2/2} dy$$

Table of Normal Distribution Probabilities $\Phi(x)$										
x	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09
-1.0	0.1587	0.1562	0.1539	0.1515	0.1492	0.1469	0.1446	0.1423	0.1401	0.1379
-0.9	0.1841	0.1814	0.1788	0.1762	0.1736	0.1711	0.1685	0.1660	0.1635	0.1611
-0.8	0.2119	0.2090	0.2061	0.2033	0.2005	0.1977	0.1949	0.1922	0.1894	0.1867
-0.7	0.2420	0.2389	0.2358	0.2327	0.2296	0.2266	0.2236	0.2206	0.2177	0.2148
-0.6	0.2743	0.2709	0.2676	0.2643	0.2611	0.2578	0.2546	0.2514	0.2483	0.2451
-0.5	0.3085	0.3050	0.3015	0.2981	0.2946	0.2912	0.2877	0.2843	0.2810	0.2776
-0.4	0.3446	0.3409	0.3372	0.3336	0.3300	0.3264	0.3228	0.3192	0.3156	0.3121
-0.3	0.3821	0.3783	0.3745	0.3707	0.3669	0.3632	0.3594	0.3557	0.3520	0.3483
-0.2	0.4207	0.4168	0.4129	0.4090	0.4052	0.4013	0.3974	0.3936	0.3897	0.3859
-0.1	0.4602	0.4562	0.4522	0.4483	0.4443	0.4404	0.4364	0.4325	0.4286	0.4247
-0.0	0.5000	0.4960	0.4920	0.4880	0.4840	0.4801	0.4761	0.4721	0.4681	0.4641
0.0	0.5000	0.5040	0.5080	0.5120	0.5160	0.5199	0.5239	0.5279	0.5319	0.5359
0.1	0.5398	0.5438	0.5478	0.5517	0.5557	0.5596	0.5636	0.5675	0.5714	0.5753
0.2	0.5793	0.5832	0.5871	0.5910	0.5948	0.5987	0.6026	0.6064	0.6103	0.6141
0.3	0.6179	0.6217	0.6255	0.6293	0.6331	0.6368	0.6406	0.6443	0.6480	0.6517
0.4	0.6554	0.6591	0.6628	0.6664	0.6700	0.6736	0.6772	0.6808	0.6844	0.6879
0.5	0.6915	0.6950	0.6985	0.7019	0.7054	0.7088	0.7123	0.7157	0.7190	0.7224
0.6	0.7257	0.7291	0.7324	0.7357	0.7389	0.7422	0.7454	0.7486	0.7517	0.7549
0.7	0.7580	0.7611	0.7642	0.7673	0.7704	0.7734	0.7764	0.7794	0.7823	0.7852
0.8	0.7881	0.7910	0.7939	0.7967	0.7995	0.8023	0.8051	0.8078	0.8106	0.8133
0.9	0.8159	0.8186	0.8212	0.8238	0.8264	0.8289	0.8315	0.8340	0.8365	0.8389
1.0	0.8413	0.8438	0.8461	0.8485	0.8508	0.8531	0.8554	0.8577	0.8599	0.8621
1.1	0.8643	0.8665	0.8686	0.8708	0.8729	0.8749	0.8770	0.8790	0.8810	0.8830
1.2	0.8849	0.8869	0.8888	0.8907	0.8925	0.8944	0.8962	0.8980	0.8997	0.9015

Appendix Table 9 - Table of Normal Distribution Probabilities

REFERENCES

G. Geoffrey Vining, Statistical Methods for Engineers. Pacific Grove, CA: Duxbury Press, 1998.

Eliyahu M. Goldratt, The Goal: A Process of Ongoing Improvement. Croton-on-Hudson, NY: North River Press, 1986

Richard J. Schonberger and Edward M. Knod Jr., Operations Management. Customer-Focused Principles, 6th Edition, Irwin, Chicago, 1997.

Kenneth N. Berk, Data Analysis with Excel, Australia; Pacific Grove, Calif: Duxbury Press, 2000.

Michael A. Kropinski, The Normal Distribution Tutorial, December 14, 1997.
<http://ce597n.www.ecn.purdue.edu/CE597N/1997F/students/michael.a.kropinski.1/project/tutorial#Definition>, (May 7, 2003).