ABSTRACT

LANDGE, SWAPNIL JAYANT. Employing Simulation Modeling in the Lean Six Sigma Methodology. (Under the direction of Dr. Jeffrey Joines and Dr. Stephen Roberts).

Lean and Six Sigma concepts have brought tremendous change in the world. The notion of removing wastes using the Lean tools from a system coupled with the statistical data analysis of Six Sigma embedded into a continuous improvement framework has become standard practice in process improvement in a variety of manufacturing industries, health care and other service systems, and business offices. In certain cases, the implementation of change may face government red tape, permissions, financial hurdles and time constraints. Most Lean Six Sigma projects fail at the improvement stage or control stage either due to barriers of cost, time or permission to implement the change. They might fail because the improvements implemented were costly but did not realize enough gains.

Simulation modeling is a very useful tool in imitating real world operations which allows one to forecast the statistical effect of operational changes performed in a virtual environment in a fast and inexpensive way to determine the best course of action. In some cases, the real system may not exist or, owing to constraints, cannot be changed easily which makes Simulation Modeling advantageous over traditional tools. Simulation analysis has been used mostly as a standalone process improvement tool. While the tools associated with Lean and Six Sigma have been combined to form an overarching framework, the Simulation Modeling tool has not been used as widely as tool in the Lean Six Sigma methodology.
A case study of analyzing and improving a driver motor vehicle (DMV) system is used to illustrate how Simulation Modeling can be integrated into the Lean Six methodology and to discover the caveats that one has to be aware of when employing the simulation tool in a Lean Six Sigma project. The simulation model was able to demonstrate that a few of the potential improvements would have very little effect on the overall time in system. Also, several scenarios were run to optimize the system based on day and month of the DMV to help with staffing needs. Various hypotheses made by the Lean Six Sigma team were tested and validated. Different Schedule patterns for additional workers, part time workers and temporary workers was tried and tested using KN Algorithm to identify the best schedule and number of additional, part time and temporary workers required. The worker schedule pattern was checked using additional workers, temporary workers and part time workers to check the influence on the time in system. Optimal values of their schedule were obtained using KN algorithm. The simulation model helps in understanding the real time effects of all these changes for any time of the year and can be used to predict the effect of various changes that can be made to the offices and effects on the various output responses.
Employing Simulation Modeling In the Lean Six Sigma Methodology

by
Swapnil Jayant Landge

A thesis submitted to the Graduate Faculty of North Carolina State University in partial fulfillment of the requirements for the degree of Master of Science

Industrial Engineering and Textile Engineering (Co-Major)

Raleigh, North Carolina
2015

APPROVED BY:

_______________________________  ________________________________
Dr. Jeffrey Joines            Dr. Stephan Roberts
Committee Co-Chair          Committee Co-Chair

_______________________________  ________________________________
Dr. Russell King            Dr. Jerome Lavelle
Committee Member            Committee Member
BIOGRAPHY

Swapnil J. Landge was born on July 20, 1990, in the city of Vadodara, India to Jayant Landge and Kalika Landge. In June 2008, he completed graduated from Don Bosco High School in Vadodara. In June 2012 finished his Bachelor’s Degree with First Class in Textile Engineering from The Maharaja Sayajirao University of Baroda. He is very interested in product development and entrepreneurial activities. He has also been actively involved in organizing national-level technical and non-technical events for his college as well as campus recruitment events. During his undergraduate education, he started building websites for corporate and academic institutions as a hobby. After graduation, he invented a device for measuring yarn length and registered a patent. Based on his invention, he developed a product-based enterprise called “textil inc.” and started selling this product to some of the textile industries. In early 2013, he registered a second patent on his spiral coiling machine for Card and Draw-frame machines. During the summer of 2014, he designed an electromagnetic water pump called i-jet for nail salons and this product is going to be commercialized in July 2015.

Currently, he is pursuing his Master of Science degree in Textile Engineering and Industrial Engineering. He is working with Dr. Joines, who has been his mentor for past 2 years. Swapnil and Dr. Joines are working on a project with the North Carolina Department of Transportation for optimization of their offices using Simulation Modeling with SIMIO™.
ACKNOWLEDGMENTS

The author would like to thank Dr. Joines for his tireless efforts and guidance throughout the project. He would like to heartily thank him for the knowledge he shared with the author and helped him understand some of his weaknesses and strengths. The author greatly respects him for his patience and his principles. Moreover, he would also like to thank Dr. Roberts for his guidance on the project and thesis. His expertise and detailed thinking ability inspired the author to understand the level of thinking one should cultivate while approaching a project.

The author also wants to thank Dr. King, Dr. Lavelle and Dr. Zhang for being on his committee and taking time for this master’s thesis. He also wants to thank Dr. Hinks for believing in him. The author wants to thank Dr. Rust for everything he has achieved in his graduate studies and giving him opportunities and advice that helped the author pave his way towards success. Finally, the author would like to thank his parents and family for having faith in him and motivating him in his most difficult times.
TABLE OF CONTENTS

LIST OF TABLES...........................................................................................................viii

LIST OF FIGURES........................................................................................................ix

Chapter 1 Introduction .................................................................................................1

Chapter 2 Literature Review .........................................................................................4

  2.1 Six Sigma and Design for Six Sigma.................................................................4

    2.1.1 DMAIC Process......................................................................................... 6

  2.2 DMADV Process .................................................................................................10

    2.2.2 DFSS vs DMAIC......................................................................................12

  2.3 Lean Systems ......................................................................................................12

  2.4 Simulation ..........................................................................................................17

    2.4.1 Simulation Types.......................................................................................18

    2.4.2 Simulation Methodology ..........................................................................20

  2.5.1 Simulation in Six Sigma and Design for Six Sigma .....................................23

    2.5.2 Simulation in Lean...................................................................................26

Chapter 3 Research Methodology ..............................................................................28

  3.1 Define Phase .....................................................................................................28

    3.1.1 Mapping the Process................................................................................30

    3.1.2 Understanding Worker Behavior ..............................................................34

  3.2 Measure Phase ..................................................................................................35

    3.2.1 Data Collection at the DMV.....................................................................36

    3.2.2 Issues with the Data .................................................................................40

    3.2.3 Input Modeling..........................................................................................42

  3.3 Analyze Phase ...................................................................................................45

    3.3.1 Building the Initial Current State Simulation Model of the DMV ............47

    3.3.2 Simulation tools and software used..........................................................47
3.3.3 Validation and Verification of the Model .................................................. 68

Chapter 4 Data Analysis and Results ................................................................. 73

4.1 Improve Phase ............................................................................................... 73
  4.1.1 Looking at 7:00 AM Start ........................................................................... 74
  4.1.2 Optimizing the Starting Times for Additional Workers ......................... 81
  4.1.3 Optimizing Three Additional Workers .................................................... 82
  4.1.4 Optimizing Two Additional Workers ....................................................... 88
  4.1.5 Adding One Additional Worker ............................................................... 90
  4.1.6 Comparing Best Starting Schedules for Different Number of Workers . 92
  4.1.7 Effect of Adding Three Temporary Workers ......................................... 102

4.2 Control Phase .............................................................................................. 109

Chapter 5 Conclusions and Future Work ......................................................... 111

  5.1 Summary of Conclusions ............................................................................. 111
  5.2 Limitations and Recommendations ........................................................... 112
  5.3 Future Work ............................................................................................... 113

References ......................................................................................................... 115

Appendices ....................................................................................................... 118

  Appendix A ...................................................................................................... 119
LIST OF TABLES

Table 2.1: DMAIC Process ........................................................................................................ 6
Table 2.2: DMADV Process ..................................................................................................... 11
Table 2.3: Seven Wastes of Lean Principles .............................................................................. 13
Table 3.1: Distributions that Were Used to Model the DMV ......................................................... 45
Table 3.2: Fields of the Customer Table .................................................................................... 50
Table 3.3: Customer Processing Time Table ............................................................................... 52
Table 3.4: Added State Variables and Properties of the Worker Object ..................................... 54
Table 3.5: Added State Variables of the ModelEntity ................................................................ 60
Table 4.1: Scenarios of Different Schedules for 14 Workers (Three Additional Workers) and its Effect on Time in System (Monday) ........................................................................ 85
Table 4.2: Description of the Temporary Work Schedule Pattern ........................................... 105
LIST OF FIGURES

Figure 2.1: Lean Guiding Principles ................................................................. 15

Figure 2.2: Graphical Representation of 24 Lean Tools and Their Broader Categories
(taken from Goforth, 2007) .................................................................................. 16

Figure 2.3: Lean Six Sigma DMAIC Process Overview with Tools ...................... 17

Figure 2.4: Simulation Model of a Process ............................................................ 18

Figure 3.1 DMV office Process flow chart ........................................................... 32

Figure 3.2: Flow chart for customers taking a road test ....................................... 32

Figure 3.3: Flow Chart for Customers Taking Written Test ................................... 33

Figure 3.4: Flow chart for customers needing a State ID card, reinstatement or other transactions ................................................................. 34

Figure 3.5: Understanding the system behavior based on the location ................. 34

Figure 3.6: Data Collection Plan Worksheet ......................................................... 38

Figure 3.7: Examiner Availability Data Collection Plan ........................................ 39

Figure 3.8: Problems with some of the collected data and how simulation served useful in finding solution to the problems ......................................................... 41

Figure 3.9: Fitted Distribution for Other Transaction Customer Type .................. 44

Figure 3.10: Portion of the SIMIO™ Customer Data Table .................................. 51

Figure 3.11: Customer Processing Time and Priority Table .................................. 53

Figure 3.12: Customer Processing Table as a Related Table to the Customer Table 53

Figure 3.13: Home Node Assignment Table for each Worker Based on the Worker ID ................................................................................................................. 55
Figure 4.3: Comparison of Old and New System by Changing Number of Workers and its Effects on the Average Time Spent in Waiting Queue ........................................77

Figure 4.4: Number of Balking Customer Comparison between Old and New Systems ................................................................................................................................................................................................................78

Figure 4.5: Comparison of Number of Customers Seen between Old and New Systems ................................................................................................................................................................................................................79

Figure 4.6: Average Time in System for Each Day of the Week of High and Low Volume Month................................................................................................................................................................................................................80

Figure 4.7: Comparison of Average Time in System between Low-Volume and High-Volume Month by Day................................................................................................................................................................................................................80

Figure 4.8: Worker Notation Explanation................................................................................................................................................................................................................81

Figure 4.9: Time in System Based on Hour with Best Scenarios Selected Using KN Algorithm for 14 Workers with Different Starting Times for a Monday of Mid-Volume Month................................................................................................................................................................................................................83

Figure 4.10: Average Time in System Based on Hour with Best Scenarios Selected using KN Algorithm for 14 Workers with Different Starting Times for a Monday of High-Volume Month................................................................................................................................................................................................................84

Figure 4.11: Scenarios of Different Schedules for 14 Workers (Three Additional Workers) and its Effect on Average Time in System (Monday) .........................85

Figure 4.12: Scenarios of Different Schedules for 14 Workers (three additional workers) and its Effect on Number of Balking Customers (Monday) .........................86
Figure 4.13: Scenarios of Different Schedules for 14 Workers (three additional workers) and its Effect on Average Time in System (Friday)........................................87

Figure 4.14: Scenarios of Different Schedules for 14 Workers (three additional workers) and its Effect on Number of Balkers (Friday).....................................................88

Figure 4.15: Scenarios of Different Schedules for 13 Workers (two additional workers) and its Effect on Average Time in System (Monday)..............................................89

Figure 4.16: Scenarios of Different Schedules for 13 Workers (two additional workers) and its Effect on Average Time in System (Friday)..............................................90

Figure 4.17: Scenarios of Different Schedules for 12 Workers (one additional worker) and its Effect on Average Time in System.................................................................91

Figure 4.18: Scenarios of Different Schedules for 12 Workers (one additional worker) and its Effect on Time in System (Friday).................................................................92

Figure 4.19: KN Algorithm-based Best Scenarios of Different Worker Populations and their Effect on Average Time in System (Monday, Mid-Volume).................93

Figure 4.20: Time in System over the Time Customer Arrives........................................93

Figure 4.21: KN Algorithm-based Best Scenarios of Different Worker Populations and their Effect on Average Time in System (Monday, High-Volume).................94

Figure 4.22: KN Algorithm-based Best Scenarios of Different Worker Populations and their Effect on Average Time in System (Friday, Mid-Volume).....................95

Figure 4.23: Number of Balking Customers for a Monday of Mid-Volume Month based on the Number of Workers present.................................................................96
Figure 4.24: Number of Balking Customers for a Friday of High-Volume Month based on the Number of Workers Present .................................................................97

Figure 4.25: KN Algorithm-based Best Scenarios of Different Worker Populations and their Effect on Time in System (Friday, High-Volume) ........................................98

Figure 4.26: Number of Customers Served on Monday of a Low-Volume Month with Different Numbers of Workers and Best Schedules Selected Based on KN Algorithm ........................................................................................................99

Figure 4.27: Number of Customers Served on Monday of a Mid-Volume month with Different Numbers of Workers and Best Schedules Selected Based on KN Algorithm ........................................................................................................99

Figure 4.28: Number of Customers Served on Monday of a High-Volume month with Different Numbers of Workers and Best Schedules Selected Based on KN Algorithm ........................................................................................................100

Figure 4.29: Number of Customers Served on Friday of a Low-Volume Month with Different Numbers of Workers and Best Schedules Selected Based on KN Algorithm ........................................................................................................101

Figure 4.30: Number of Customers Served on Friday of a Mid-Volume Month with Different Numbers of Workers and Best Schedules Selected Based on KN Algorithm ........................................................................................................101

Figure 4.31: Number of Customers Served on Friday of a High-Volume Month with Different Numbers of Workers and Best Schedules Selected Based on KN Algorithm ........................................................................................................102
Figure 4.32: Effect of Temporary Workers Working from 7 AM to 11 AM on Average Time in System per hour .................................................................103

Figure 4.33: Effect of Temporary Workers Working from 10 AM to 2 AM on Average Time in System per hour .................................................................104

Figure 4.34: Average Time in System for Different Full and Part Schedules........106

Figure 4.35: Comparing Different Part Time Schedules for One, Two and Three Temps .................................................................107

Figure 4.36: Effect of Temporary Workers Working from 7 AM to 11 AM on the Number of Balking Customers .................................................................108

Figure 4.37: Effect of Temporary Workers Working Full and Part Time on the Number of Customers Served .................................................................109

Figure 72: Greensboro Office Model .................................................................119

Figure 73: Durham Office Model .................................................................120

Figure 74: Charlotte Office Model .................................................................120
Chapter 1
Introduction

Since the time they have existed, people have been trying to improve them in order to reduce cost and raise productivity because the profit margins are very low. This has been especially true for manufacturing systems. Henry Ford simplified and optimized the assembly line for making cars by reducing setups and producing cars more quickly. These process improvements led to producing cars that were less costly and thus more affordable for consumers. These same concepts were introduced in the 19th century in meat packing and textile mills. Ad hoc methods of process improvement have been used for quite some time. After World War II, Japanese companies were facing a resource crisis; hence, the Lean manufacturing concept started as a way to systematically eliminate wastes such as overproduction or defects in any system in order for businesses to survive. Implementing Lean as a continuous improvement philosophy helped the Japanese auto companies who owned very little world market share to become the world leader. Lean thinking was born there and brought back to the US in later years.

While Lean was focused on eliminating wastes including quality defects, it was not as focused on quality as on other issues. During the late 1900s, many forms of quality process improvement were being employed to meet customer needs like total quality management (TQM), quality control charts, and other company-grown quality initiatives. Starting in the 1980s, global competition forced companies to become more focused on producing quality products in a timely fashion or lose market share. Motorola took on an ambition to produce quality products in a shorter time horizon and turned around their tech company. They took
what others had been doing using statistical quality data and placed it inside of a continuous improvement framework called Six Sigma where the five stages were define, measure, analyze, improve, and control (DMAIC). Others like Jack Welch of GE took Six Sigma beyond manufacturing and applied it to improve all aspects of the corporation. In many cases, the tools that were already available were packaged into a framework that was consistent every time the methodology was applied.

Since its creation, people have been using the computer to be build virtual systems of the real world. In the late 1980s, as computer systems became more accessible and faster, people began building simulation models of manufacturing as well as service systems to analyze them. These virtual systems allowed the designers to play “what if” scenarios to make improvements or make better decisions without costly mistakes or interacting with the real system. Simulation Modeling tools have gained wide acceptance across many different industries. In recent years, the tools of Lean have been embedded into the Six Sigma process to produce the Lean Six Sigma methodology. While a simulation tool is an improvement methodology and can act as another tool during the methodology, it has not been widely used during Lean Six Sigma projects.

For example, the North Carolina Department of Motor Vehicle (NC DMV) locations were having a problem with long processing times for customers. The percentage of satisfied customers was just 75% while the remaining customers claimed that their visit time was unreasonable. Therefore a Lean Six Sigma project was started to develop improvement strategies to assist the DMV. There was concern that many of the potential improvements may be costly to implement or other improvement ideas might be missed. Therefore,
simulation modeling offers a potential tool to carry out all possible changes and improvements in a virtual environment which may be more cost effective and quicker.

This thesis will describe the potential use of Simulation Modeling in the Lean Six Sigma process and the caveats one may encounter in employing Simulation Modeling in the process. This document consists of five parts where the first chapter will introduce the world of Lean, Six Sigma and Simulation. Chapter 2 will outline all the basic elements of the process improvement methodologies of Lean, Six Sigma, DMAIC and Simulation Modeling with some examples of case studies that describe the use of simulation in these processes. Chapter 3 will describe the methodology of using simulation in a Six Sigma project during define, measure, and analyze phases of the DMAIC Six Sigma process using the DMV as the case study. Chapter 4 will show how simulation is used in the improvement and control phase of the project, while Chapter 5 will give conclusions and point to future work that can be further carried out.
Chapter 2
Literature Review

As mentioned previously, Lean thinking, Six Sigma, and Simulation Modeling are all tools used in improvement of processes. Most process improvement methodologies deal with systems that are complex and rarely perform the way they are expected to. These three tools are often used in combination with one another as will be described.

2.1 Six Sigma and Design for Six Sigma

Six Sigma is a business philosophy focusing on continuous improvement to reduce and eliminate variability. In a service or manufacturing environment, a Six Sigma (6σ) process would be virtually defect free (i.e., only allowing 3.4 defects out of a million opportunities of a process or 99.99966% defect free). However, most companies operate at four sigma which allows 6,210 defects per million. Six Sigma began in the 1980s when Motorola set out to reduce the number of defects in its own products. Motorola identified ways to cut waste, improve quality, reduce production time and costs, and focus on how the products were designed and made that utilizes statistical tools and the scientific method (Schroeder et al., 2005). Six Sigma grew from this proactive initiative of using exact measurements to anticipate problem areas. In 1988, Motorola was selected as the first large manufacturing company to win the Malcolm Baldrige National Quality Award. As a result, Motorola's methodologies were launched, and soon their suppliers were encouraged to adopt the 6σ practices. Today, Fortune 500 companies who use the Six Sigma methodology achieve significant cost reductions which have increased the methodology’s adoption in all types of industries (Hilton and Sohal, 2012).
Six Sigma evolved from other quality initiatives such as International Organization for Standardization ISO, Total Quantity Management (TQM) and Baldrige to become a quality standardization process based on hard data and not hunches or gut feelings, hence the mathematical term, Six Sigma. Six Sigma utilizes a host of traditional statistical tools but encompasses them within a process improvement framework. These tools include affinity diagrams, cause & effects, failure modes and effective analysis (FMEA), Poka Yoke (mistake proofing), survey analysis (voice of customer), design of experiments (DOE), capability analysis, measurement system analysis, statistical process control charts and plans, etc. (Lee and Choi, 2006; Mellat-Parast and Digman 2011; Hilton and Sohal, 2012).

The Six Sigma framework utilizes one of two different structured methods (i.e., DMAIC or DMADV) depending on whether one is improving a current process or product or creating a new process/product. The DMAIC process utilizes five different steps: Define, Measure, Analyze, Improve, and Control which are done sequentially with many loop backs to earlier steps. Design for Six Sigma, or the DMADV process, is utilized when one is creating a new process or product or the current cannot be improved. This process uses five steps similar to those of DMAIC: Define, Measure, Analyze, Design, and Verify (Breyfogle, 2003; Pyzdek and Keller, 2014). Both frameworks utilize data-intensive solution approaches and eliminate the use of anecdotal evidence or intuition in making decisions and improvements. Both processes use continuous improvement from one stage back to the beginning. For example, if during the analyze phase one determines a key input is not being measured, new metrics have to be defined or new projects can be defined once the control
phase is reached. The stages often feedback to previous stages as data is gathered and analyzed.

2.1.1 DMAIC Process

The Six Sigma method is based on the DMAIC process and is utilized when the product or process already exists but it is not meeting the specifications or performing adequately as described in Table 2.1. Several different tools are used throughout the methodology.

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Define</td>
<td>Identify, prioritize and select the right projects. Once determined, then define the project goals and deliverables for the problem.</td>
</tr>
<tr>
<td>Measure</td>
<td>Define the key product/process characteristics to create a baseline by gathering data.</td>
</tr>
<tr>
<td>Analyze</td>
<td>Identify the key process determinants and/or root causes of variability.</td>
</tr>
<tr>
<td>Improve</td>
<td>Develop solutions and optimize performance by eliminating defects.</td>
</tr>
<tr>
<td>Control</td>
<td>Sustain the current gains and future process performances.</td>
</tr>
</tbody>
</table>

2.1.1.1 DMAIC – Define

The define phase is the most important phase of the DMAIC process, as it helps to understand and define the problem as well as establish goals. This is achieved through thorough brainstorming and identifying the project needs and deliverables from the customer’s perspective. A project charter is prepared to define the problem (i.e., problem statement) and set goals (i.e., mission statement) for the project, identify stakeholders, define resources, and calculate investment and savings cost analysis. Some important tasks during the define stage are development of critical to quality (CTQ) points as well as mapping the process using a SIPOC and/or detailed process map. CTQs are the measurable characteristics deemed important by the customer used to determine the capability of the process or product.
Voice of the customer (VOC) data is used to establish the CTQs. The Supplier, Inputs, Process, Outputs and Customers (SIPOC) is a high-level diagram to identify who are the external and internal suppliers and customers, determine the inputs and outputs to the process, and develop a 10,000 foot view of the process (i.e., five or seven process steps). SIPOC can assist in identifying the root cause analysis, and data that need to be collected, including VOC data from all customers. A detailed process map can be used to fully understand the entire process by identifying potential failures, data collection points, boundaries, etc. (Breyfogle, 2003).  

2.1.1.2 DMAIC – Measure

The measure phase is the next step where sample data is gathered to be considered for analysis and the establishment of baseline to determine if improvement has occurred. Metrics are established to that will measure/determine if CTQs are met. Many tools can be used to establish a baseline (i.e., capability analysis, Pareto or run charts). If customer specifications are present (i.e., lower (LSL) and/or upper (USL) specification limits) capability analysis can be calculated which can determine capability of the current process. The terms Cp and Cpk are calculated using the upper and lower specifications, the mean of the process data (\( \bar{X} \)) and the standard deviation of the data (\( s \)). Cp measures if the process is capable of meeting the customer specs (Cp = 2.0 would be a Six Sigma process). Cpk measures whether the process is currently meeting specifications. If Cp and Cpk are equal, then the process is centered between the specifications; otherwise, it would be closer to either the USL or LSL (Breyfogle, 2003).
In this phase, a thorough brainstorming is done to identify the project metrics and potential root causes and determine the necessary data requirements to gauge the project deliverables. During this phase, careful consideration of whether data is to be collected once or several times is necessary, recognizing the difficulties in repeating the data collection. Some of the useful tools used for this step are a data collection plan and measurement system analysis (MSA). The data collection plan is the developed method or the tables that are to be used to collect the data and is very important when the Six Sigma team may not be the ones actually collecting the data. The Six Sigma team decides what should be recorded and the frequencies and accuracy of the data collected. Measurement systems analysis determines the capability of the system used to measure. This analysis shows how repeatable and reproducible the measurement system is (i.e., how much variation is due to the process versus the system used to collect the data).

2.1.1.3 DMAIC – Analyze

The analyze step is used to identify the root causes of the various problems identified during the define phase and to represent the causes of variation or defects. Based on the collected data, the variation is calculated for each response variable, and relationships between the control (i.e., independent) and the response (i.e., dependent) variables are tried to be established. Some of the tools used during the analyze phase are histograms, Pareto charts, fish bone (i.e., causes and effect) diagrams, FMEA, five whys as well as statistical tools to
perform hypothesis testing and establish relationships between independent and dependent variables using regression and correlation. Histograms are used to check the nature and distribution of the data while Pareto charts are used to identify the severity (i.e., frequency) of causing factors. Fish bone or Ishikawa diagrams are used to analyze the cause and effects of the variation in the response metrics. Failure mode effects analysis (FMEA) is used to identify the modes of failures and their effects. Each mode of failure is given severity level, probability of occurrence and risk level, which are computed based on the level of severity based on frequency of occurrence. These modes are often identified from the SIPOC or process maps from the define stage. The 5 whys are used to question why is something occurring to get to the root cause. The analysis step is used to determine the baseline as well as what x’s are potentially causing the variation in the system.

2.1.1.4 DMAIC – Improve

From the identification of the potential root causes from the Analyze step, the Improve phase is concerned with eliminating these problems in order to lead to decreased variation and/or defects depending on the objective. The common tools used in this step include FMEA again, error proofing to reduce human error by automation, training or redesign of the process, design of experiments (DOE), brainstorming, and potentially simulation modeling. DOEs are used to evaluate the effect of controlled factors on a response variable which is under investigation by planning, conducting, analyzing and interpreting a set of controlled tests. Design of experiments can be performed by keeping all control variables constant except one whose effect on the response is measured by altering the levels of this factor. This one factor at a time (OFAT) approach has been proven to not be very effective in comparison
to some of the new design methodologies since this method does not take account potential interactions among the independent variables. There are several DOE designs (i.e., full and fractional factorial, response surface, etc.) used to screen out factors of importance to determine a relationship among the dependent and independent variables. Multiple runs (i.e., replicates) may be necessary to determine which causes are the most important and the best values for these inputs. It is very important once solutions to the problem are found that a pilot is run to determine the true effect of the change.

2.1.1.5 DMAIC – Control

The objective of this phase is to sustain the improvements yielded during the improvement stage through a control plan. The control plan uses statistical control charts for monitoring, training workers, and visual controls to determine a problem before it happens.

2.2 DMADV Process

If the process or product does not exist and needs to be developed, the Design for Six Sigma (DFSS) process (DMADV) has to be employed. Processes or products designed with the DMADV process typically reach market sooner and have decreased costs and less rework. Even though the DMADV is similar to the DMAIC method and starts with the same three steps, they are quite different as defined in Table 2.2.
### Table 2.2: DMADV Process

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Define</td>
<td>Determine customer needs and specifications through voice of customer and then identify metrics to meet the CTQs.</td>
</tr>
<tr>
<td>Measure</td>
<td>Based on the objectives set during define phase, Measure the required inputs for analysis.</td>
</tr>
<tr>
<td>Analyze</td>
<td>Identify the key process options necessary to meet the customer needs.</td>
</tr>
<tr>
<td>Design</td>
<td>Design a detailed process or product that will meet the customer needs as well as be a Six Sigma product/process.</td>
</tr>
<tr>
<td>Verify</td>
<td>Make sure the design performance and ability will meet the customer needs where the customer can be internal and/or external to the organization.</td>
</tr>
</tbody>
</table>

#### 2.2.1.1 DMADV – Design

The first three phases of the DMADV process are similar to the DMAIC method except one relies heavily on VOC, both internal and external to the process, to really determine the customer needs and establish metrics to measure those needs which will be used in the design phase to evaluate design alternatives. Unlike DMAIC, the fourth step is development of the new process or product. The process or product is designed from the start to meet Six Sigma standards, and hence it is technically termed as DFSS (Design for Six Sigma). The main objective of the design phase is to design the process or product according to the customer needs using the house of quality (i.e., quality function deployment (QFD)), benchmarking, prototyping, and robust selection (Breyfogle, 2003)

#### 2.2.1.2 DMADV – Verify

In this phase the capability and requirements of the product or process are verified to satisfy the customer needs, such as prototyping or process creation, as well as maintain consistent variation under the control limits set at the beginning of the phase. The main idea of the
verify phase is to ensure that the product or process will follow a Six Sigma quality standard when delivered to the internal or external customer.

2.2.2 DFSS vs DMAIC

There are several times throughout a Six Sigma project when the team needs to evaluate whether they should be following the DMAIC or the DMADV process. In the Define phase a team may realize that the problem they are addressing exists because a process or product does not exist. During Analyze and Improve, teams sometimes discover that removing the identified root causes is not going to provide enough improvement to the process, or they may find the process is so broken that replacing it is the best answer. When these situations occur, the team should transition their project to the DMADV path, but they should not discard the work they have done because it may be useful in the DMADV process.

2.3 Lean Systems

Many people confuse Lean with Six Sigma. Six Sigma is a continuous improvement or management philosophy to help reduce the process variability while Lean is a management or manufacturing philosophy of eliminating wastes and non-value added processes from a system. Lean originated out of the Toyota Production system (TPS) during the 1970s (Taj, 2008; Shah and Ward, 2007). Lean principles were popularized in the USA by Womack et al. (1990) in the book *The Machine that Changed the World*. The elimination of waste is key in Lean systems, and Toyota defines three types of waste: muda (i.e., non-value-added work), muri (overburden), and mura (unevenness). Most people think of the non-value added form of waste when referring to Lean (e.g., a part sits in queue for ten minutes before being processed for one minute which represents ten minutes of non-value-added time). According
to Lean methodology, there are seven different types of waste that could be present in a system as given in Table 2.3.

<table>
<thead>
<tr>
<th>Waste</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transportation</td>
<td>Physically moving products from one place to another when it is not required in the actual processing.</td>
</tr>
<tr>
<td>Inventory</td>
<td>All raw materials, WIP, finished products or sub-assembly products that are currently not processing.</td>
</tr>
<tr>
<td>Motion</td>
<td>Waste is concerned with unnecessary motion of people, goods or machines during the processing of the part (e.g., looking for a tool)</td>
</tr>
<tr>
<td>Waiting</td>
<td>Parts or people waiting at the next step of production (i.e., queued up)</td>
</tr>
<tr>
<td>Overproduction</td>
<td>Making more than the demand would increase the finished goods inventory and also hold cost which is a waste. In certain cases, goods perish and depreciate in value.</td>
</tr>
<tr>
<td>Over processing</td>
<td>Poor tools, poor service or poor product design creates unnecessary processing which takes time and cost, hence it is considered as a waste. Sometime even over-engineering the product that a customer doesn’t want to pay for is a waste. For example, if the customer is ok with 90% defect free product, then making 99% defect free product is waste.</td>
</tr>
<tr>
<td>Defects</td>
<td>Defect is an important waste as it requires cost and time to overcome, inspect and replace the defected parts.</td>
</tr>
</tbody>
</table>

These defined wastes are identified to reduce the non-value added activities in a manufacturing process in hopes of decreasing cycle time. Lean deployment also involves use of various other tools like Jidoka, JIT, Kanban, value stream mapping, and pull system. Womack and Jones (1990) define the Lean guiding principles as a five step process to achieve a Lean enterprise which involves numerous tools as seen in Figure 2.1. The first step in Lean thinking is determining the Value of the process or product. Value is defined from the perspective of the end customer and represents something they could be willing to pay. Does the activity change the form, fit, or function of the product or service? Value is defined by the customer by having the right product or service at the right time at an appropriate price. The Value stream step determines the set of actions required to bring a product or
service to the hands of the customer. All actions can be categorized into three different buckets: Value added activity and Non-value added activity that cannot be avoided owing to constraints (sometimes called business-value added). Inspection is a non-value-added activity because it wastes resources to perform the action that is unnecessary if one created a correct process. Just because a customer requires an activity, this requirement does not make it business-value-added. However, FDA requires inspection of drug makers by law and it has to be done in order for drug makers to stay in business.

Once the value stream has been identified, the Flow step is used to create a continuous sequence of value-added tasks along the value stream, eliminating all the waste discovered. Once the next sequence is moving toward a Pull system often referred to as just in time (JIT), no upstream activity should produce a product or service until the downstream customer asks for it. Material or information is only presented when needed and requested. The goal is then to move toward perfection and not just strive to get ahead of the current competition. Notice the process repeats itself, as Lean is a never ending process (Womack and Jones, 1990).
Value stream mapping is a useful step in becoming lean, and this activity helps in identifying value added and non-value added activities. Value stream mapping is similar to the detailed process maps but with added information. Information flow as well as product flow is depicted. Everything is converted into the units of time based on the customer needs called the Takt time which is defined by the effective working time per shift divided by the customer requirement per shift. The practical use of Takt time is having people understand how often and when product should be advancing. When those times are missed, it clues people into needing to look immediately for the problem or source of the extra time. The value added and non-valued added times are documented along with other types of data information on the map. The current state map depicts the flow of product and information through the value stream. It documents all process measures in terms that can be used to identify waste but does not identify solutions; that comes later. Next, an ideal state map is
developed showing perfection in all cycles: In this map, every step and action will add value for the end customer, and the map will display how the process would move, with no impediments to flow. The ideal state may not be achievable based on the current state, so a future state is developed which has addressed problems discovered (i.e., the wastes in the system) in the current state that brings one closer to the ideal state.

Lean encompasses a wide range of tools that are used to implement changes as summarized by Figure 2.2 taken from Goforth (2007). Many of the tools still use the Japanese words (e.g., Poka Yoke or mistake proofing).

![Graphical Representation of 24 Lean Tools and Their Broader Categories](image)

**Figure 2.2: Graphical Representation of 24 Lean Tools and Their Broader Categories (taken from Goforth, 2007)**

In recent years, Lean principles and tools have been combined within the Six Sigma methodology to form Lean Six Sigma as a means to improve organizational performance.
Lean Six Sigma is the latest generation of improvement approaches, and many companies have experienced superior performance and customer satisfaction, efficient results in sales forecasts, effectiveness and efficiency of sales force as well as decreased inventory costs through successful Lean Six Sigma implementation (Motwani et al., 2004; Antony and Banuelas, 2002; Sharma, 2003; Marti, 2005; Hesselschwerdt, 2006; Gabor, 2001). Figure 2.4 shows the Lean Six Sigma DMAIC process with both the Six Sigma tools and Lean tools one can use at the various steps.

**Variation Reduction**
- Charter
- VOG/CTQ
- Project Plan
- Stakeholder Analysis
- Process Map

**MEASURE**
- Project Metrics
- Potential Xs
- Data Collection Plan
- Defect Count
- MSA
- Capability
- Baseline

**DEFINE**
What problem are we addressing?
- Charter
- VOG/CTQ
- Project Plan
- Stakeholder Analysis
- Process Map
- Value Stream Map

**ANALYZE**
What are the root causes of the problem?
- 5 Whys
- Process Maps
- FMEA
- Graphical Analysis
- Hypothesis Testing

**IMPROVE**
What is the best solution to remove each root cause?
- DOE
- Pilot
- FMEA
- Error Proofing

**CONTROL**
How can we insure the gains are maintained?
- SPC
- Control Plan
- Audits
- Training
- Translation

**Waste Reduction (Lean)**
- 7 Waste Determination
- Cycle Time
- WIP
- Takt Time

**Graphical Data Analysis**
- Future State VSM

**5 S**
- Pull Systems
- Error Proofing
- Kaizen Events
- Kanban
- Load Leveling

**Process Flow Maps**
- Visual Controls
- Audits
- Training

**Figure 2.3: Lean Six Sigma DMAIC Process Overview with Tools**

### 2.4 Simulation

The previous sections discussed the Lean and Six Sigma methodologies and highlighted some of the tools that can be used. Simulation Modeling is another tool that is often used for process improvement. Models represent the imitation of the operational behavior of a system or process, as seen in the graphical representation in Figure 2.4. There are different types of
models (e.g. mathematical models, statistical or regression equation models derived from data or DOEs, computer simulation models, etc.). Simulation modeling is a tool to mimic the real-world systems in a virtual environment in order to avoid making physical prototypes or using the real system, thus decreasing the cost and time in attaining the objectives (Akshay, 2012). Simulation of system or process models is the method of replicating and running the models in time to compute or predict the future behavior and specific objectives desired.

Figure 2.4: Simulation Model of a Process

2.4.1 Simulation Types

There are four types of simulation models: Monte Carlo simulation, discrete event simulation, agent based simulation, and system dynamics. Apart from Monte Carlo simulation, all are dynamic modeling paradigms.
2.4.1.1 Monte Carlo Simulation

Monte Carlo Simulation is a statistical method of repetitive random sampling of data distributions to obtain mathematical functions that represent operations of complex systems. The method of Monte Carlo simulation involves the following pattern: 1) modeling the system as a series of probability distributions, 2) Repeatedly sampling the CDFs, and 3) tallying the required statistics. In Monte Carlo simulation, the whole system is simulated a large number of times where each simulation is referred to as a realization of the system. For every realization, the simple random values corresponding to every uncertain parameter is sampled. This system is simulated in time for computing the performance of the system.

2.4.1.2 System Dynamics

System dynamics, developed at MIT by Forrester in the 1950s, is a methodology in which Simulation Modeling is used to frame the dynamic nature of complex systems. System dynamics models capture the nonlinearity of complex systems (e.g., policy making, economic systems, etc.) using time delays, feedback loops, stocks which represent a quantity of interest at given point of time, and flows representing the rate or speed of variables over time explaining the behavior of an entire system. These systems are run until they reach equilibrium and are further defined by Sterman (2000).

2.4.1.3 Agent Based Simulation (ABS)

Bonabeau (2002) describes ABS as a method for modeling a complex system by developing simple autonomous decision making agents who follow simple behavior rules defined by the modeler. The interaction of these agents with one another and the environment of interest over time generate the system characteristics and outputs. The system properties emerge as a
result of the interaction of these agents with each another and the environment which is of interest to the modeler. ABS is often used to model human behavior since agents can be used to mimic their counterparts in the real world.

2.4.1.4 Discrete Event Simulation

A discrete event simulation is a computer-based programmed modeling approach for representing real or hypothetical systems by noting only those points in time that represent a state change of system. These points are considered as a series of discrete events. Some people consider DES as Monte Carlo simulation models on a network, capturing queuing and utilization over time. For example, let’s consider an event in time such as arrival of people in an office where each particular arrival is considered an event. Each event takes place at a discrete time interval which can be plotted on a graph of arrivals versus time as a set of discrete points in time which is thus called discrete events. Virtual computer modeling of these events in time is called discrete event simulation.

2.4.2 Simulation Methodology

The following are the basic steps that are used to develop simulation models of a system or process (Banks, 2000).

1. Problem Definition – This step involves understanding and documenting the problem to be addressed by simulation. It is necessary to understand whether simulation can be used to solve the problem or not.

2. Project Planning – The tasks for completing the projects are sub-categorized in a sequential order and planned using a Gantt chart to manage serial and parallel modeling.
3. System Definition – This involves the details and flexibility of the system to be modeled. In this phase, the system behavior and the performance measures to be analyzed are studied.

4. Model Formulation – A flow chart of how things will operate is created in this step. All variables are defined and the basic model is formed.

5. Input Data Collection and Analysis – After formulating the basic model, the data required to drive the model are collected and fitted to theoretical distributions.

6. Verification and Validation – Verification is the examination of whether the simulation programs logically resemble the operations of real system. Validation is determining whether the conceptual model can replace the real model.

7. Experimentation and Results Analysis – After the model has been verified and validated, experimentation is performed with analysis of the results helping to improve a process.

2.4.2.1 Verification and Validation

Verification, validation and testing (VV&T) of simulation models (SM) should be performed after every important step of simulation. The responses of the simulation should not be considered as binary variable. This can create errors in the response because the response is not always perfect considering random numbers and random distribution, so a range of response should be created. The accuracy of the SM is judged based on the prescribed intended uses. The VV&T of simulation models should be performed in two steps. First, the VV&T of all sub models should be performed independently, and then the behavior of the entire model with all sub models working together should be VV&T for accuracy of the
model. The certification outcome should be presented with a level of confidence. The results should be properly marked with the level of confidence (Balci, 2010).

2.4.2.2 Data Collection and Input modeling

Input modeling is one of the most important parts of most simulation projects since it is the driver behind the uncertainty in the model. Sometimes data collection is a very expensive process in terms of cost and time. Often one develops an initial model void of real data to determine the inputs that are sensitive and needed to avoid collecting data that is not necessary. Input modeling describes inputs via probability distributions and the following hierarchy should be used in determining these distributions.

1. Use data to fit general distributions.

2. Use raw data and load discrete points into an empirical distribution.

3. Use the distribution suggested by the nature of the process or underlying physics.

4. Assume a simple distribution and apply reasonable limits when lacking data.

One important aspect of data collection is to identify the required data from the model like processing section, arrival section, storage or waiting section, reliability of system and other potential project-specific data. The first method is gathering data from automated systems or a manual tracking system. Often the data has to be manipulated, cleaned and filtered before one is able to fit appropriate distributions. There are numerous fitting algorithms and software like EasyFit™, ExpertFit™, Crystal Ball™, JMP™, and Minitab™ that utilize goodness of fit tests (i.e., Chi-square, Anderson Darling (AD), or Kolmogorov-Smirnov (KS)) determine the best distribution. Chi-square is used for discrete distribution fitting. Though all three can be used for continuous distribution fitting, the AD test does a
better job of fitting to the tails of the distribution while the KS test looks for distributions that fit the middle of the distribution. If one is not able to gather enough data to get a sufficient fit or no distribution will accurately fit the data, then one can use the raw data by utilizing an empirical distribution.

In case of no data, collecting the necessary data is the third step of data collection. While collecting the data, try to eliminate errors and to eliminate Johnson effect by being unnoticed during the process (Joines, 2015). In cases where you have insufficient data or no data at all, expert opinion or reasonable bounds using distributions like triangular distribution or BetaPert based on the minimum, maximum, and the most likely value can be used. Further, analyze the sensitivity of the data by some trial runs to identify which data is very sensitive, and then find the correct resolution of the data to avoid errors. For those input where output is very sensitive, try to select a very fine resolution (Sturrock, 2008).

2.5 Simulation in Lean Six Sigma Framework

Six Sigma is a process improvement methodology focused on reducing variation while Lean is a process improvement methodology focused on eliminating waste and in turn reducing cycle time. Computer Simulation Modeling projects have also been focused on improvement as well, but Computer Simulation Modeling is really a tool that can be utilized inside the DMAIC and Lean methods and offers advantages over other tools when used properly.

2.5.1 Simulation in Six Sigma and Design for Six Sigma

Simulation offers an advantage over typical tools, especially where the analysis and improvements are very difficult to implement due to cost, lack of resources, large organizations, standard systems, government policy barriers, skepticism and other
unfavorable reasons (Ferrin et al., 2005). Many of the statistical tools require the data to be normal which often doesn’t occur, especially with issues like lead times, where Simulation Modeling does not make such assumptions.

2.5.1.1 Define Phase

Six Sigma practitioners have to estimate the cost savings for each project to be certified or justify the project typically. However, these cost forecasts are often made on point estimates of key parameters (i.e., raw material cost, customer/product demand, cost of capital, currency rates, etc.). By employing Monte Carlo simulation, variability and/or ranges on these point estimates can be employed to provide a more reliable estimate on the potential savings. Along these lines, several projects have been proposed and simulations can be utilized to help management perform project selection based on resource constraints and objectives.

2.5.1.2 Measure Phase

Simulation uses a lot of data collected but may not be as useful as a tool during this phase. One could use simulation to model the measurement system to ascertain long term effects of reproducibility and repeatability, predicting the impact of error on the system.

2.5.1.3 Analyze and Improve or Analyze and Design

Analyze and Improve of the DMAIC process are the two phases that can utilize simulation the most. In most of the Six Sigma projects, analyze and improve phases are some of the difficult phases to implement, owing to dealing with real systems. Using the real system is better in terms of capturing complexities and interactions but also involves potential costs, resource limitations, or is otherwise not practical. Therefore, simulation can serve as an
important tool to meet the ends in some phases of a Lean Six Sigma project in the analyze and improvement phases. One of the main tools used during the analyze and improve phases is design of experiments (DOE). But the implementation of design of experiments with replications can be expensive and tedious. The time for running the set of experiments makes it impractical to determine the baseline or gauge the improvements of the process. Simulation can also be used in mitigating the cost of DOEs (e.g. cost of raw material, shutting down the process, loss of product, etc.) (Joines, 2008). In healthcare, DOEs are generally not used since one is often dealing with patients’ lives, but simulation can remove this barrier.

For example, Martin (2005) was optimizing the set of inventory parameters like the reorder point, order quantity, and initial inventory for a large apparel company during the Improve phase of the DMAIC process. The team needed to evaluate six inventory policies and identify which one of three suppliers was best. A true DOE with sufficient replications would have taken years to complete, owing to the long lead times, the number of SKUs, plants, etc. Therefore, a simulation model was built and a large DOE was run to determine the best set of parameters. An inventory model was chosen that minimized inventories across all the SKUs but also maximized fill rate. A pilot of the inventory model resulted in real savings and the model was implemented company wide. The simulation model was used to eliminate all the poor inventory models, and only the best one was tested in the real system, saving time and money (Martin, 2005).

As mentioned, performing improvements (i.e., DOEs) in a health care setting is very difficult as human lives are stake. Assur (2012) used a simulation model during the improve
phase to demonstrate the benefits of implementing a RFID sensor in the outpatient surgical process. Simulation was used to verify a reduction of non-value added activities of locating supplies and equipment with the implementation of RFID as well as reduction of post-operative infections owing to mistake proofing of using the RFID tags.

2.5.2 Simulation in Lean

Simulation can be a very useful add-in for Lean projects and has been used more often as compared to Six Sigma projects (Bronislave & Robert, 2014). In case of Just in Time (JIT) inventory management, simulation plays an important role of demonstrating the cost effects of implementing JIT statistically. Simulation also has a good scope in application of Kanban signals which are sent to the previous process in flow to manage inventory levels. Simulation can also be used in eliminating wastes (Mura) by production leveling (Heijunka) which can be simulated to identify the buffer levels and identify where supermarkets can be used to balance the production and eliminate bottlenecks. One more important concept of Lean is Pull system, in which the production is done based on the demand which eliminates finished goods inventory but is only applicable in relatively stable demand. Simulation has also been used in several cases of pull system models to optimize the levels of different product systems to balance the production in case of volatile demand. Simulations of manufacturing floors are invaluable tools in the implementation of Lean manufacturing. Manufacturers are often reluctant to make a change before a simulation is performed to determine the potential impact of that change. Simulation is an inexpensive tool that is used as insurance against costly mistakes (Loyd & Czarnecki, 2002). Mao and Zhang (2008) used simulation in the development of Lean practices in the construction process to demonstrate the effectiveness
and efficiency of the various Lean improvements. Other researchers have taken the value stream maps and simulated the maps by introducing variability to determine the impact of certain processes and bottlenecks (Mcdonald et al., 2002).
Chapter 3
Research Methodology

The previous chapter discussed the basic concepts of Lean, Six Sigma and Simulation as methodologies for problem solving. The integration of Simulation Modeling as a tool within the Lean Six Sigma infrastructure was presented, but it was not as clear how easy it would be or how to utilize simulation fully in the process improvement methodology. This chapter will utilize a case study of the North Carolina DMV offices to demonstrate how one can integrate simulation into the DMAIC process and what procedures need to be changed in order to facilitate Simulation Modeling fully.

3.1 Define Phase

Recall the define stage is to understand the current process, develop a project charter and establish a baseline for comparison. Initially, the project was started by a group of consultants and office members at Department of Transportation of NC. The define phase involved the set of objectives they wanted to achieve considering the project. The team mapped out a project charter which described the problem and the mission statement. The simulation team was brought in after the initial problem statement was created. The following was established by the project team (DMV Office Lean Six Sigma Project Tollgate Report 2015).

*Problem Statement:* The process for obtaining services at the driver license office is often lengthy and requires customers to wait more than 30 minutes to initiate a transaction. Currently, there is no target for what is to be considered a reasonable wait time or total transaction; it is known, however, that presently the times are longer than
desired. The extended amount of time spent fulfilling customer needs results in lost productivity, reduced customer satisfaction and low employee morale.

Mission Statement: - The mission of this project is to reduce the average total time of customers in the driver license office by 20% by October 31, 2015. This will result in more transactions being completed by DMV, less time a customer is required to spend at the driver license office and greater execution of the organizational goal of NCDOT being a great place to work. (DMV office Lean Six Sigma Project Tollgate report 2015)

Often Six Sigma Black belt or Master Black belt personnel who lead the projects have not been trained in Simulation Modeling and potentially do not know when its power can be useful. Therefore, simulation modelers are brought in at later phases of the process rather than at the beginning. However, being able to identify at this stage that a simulation model will be useful can help eliminate delays and rework if the information is not collected in a useful manner. If analysis or potential improvements are very difficult to implement due to cost, large organizations, standard systems, government policy barriers, skepticism or any unfavorable reason (e.g., health care arena), then simulation modeling techniques can be very useful to virtually simulate the entire process, product or system to avoid costly mistakes and waste. In this case, the team leader recognized that Simulation Modeling could potentially help the team, as there were several different offices the DMV wanted to improve. Therefore, the simulation team was brought in during the end of the Define and the beginning of the Measure phases. Recall the Six Sigma process is not a linear process, as one often goes back to previous steps to update information collected.
3.1.1 Mapping the Process

One of the most important steps in the Define phase is to map the current process to determine how the current system works and define the boundaries. Depending on when the simulation modeler is included as part of the project team, they can be part of the team that creates the process map, or they may be handed the process map with information that was already developed by the team. It is better to be part of the team to develop the process, as one will have to walk and observe the process and meet with the process owner. While process mapping, the simulation modeler can determine what assumptions may need to be employed and what level detail will be represented by the simulation model. On the other hand, if the modeler comes in after the mapping is done; the process map can be utilized as a great starting place to begin to understand the process. However, one often will still need to observe and question the team and possibly the process owner, potentially repeating or validating the work that has already been done.

In the case of the DMV office project, the simulation team was brought in near the end of the mapping process. The basic flow was given, but the team still had to visit all three offices of interest (Charlotte, Greensboro and Durham) to validate assumptions as well as observe actual behavior rather than rely on the Six Sigma team’s perception. A DMV office is a place where people come for various services like issuance of state ID cards, driving permits and licenses (i.e., new, renewals or reinstatement), and vehicle titles. Yet, only few offices issue titles. In the designed models, the offices do not issue titles, but still people may come asking for them, which affects system behaviors. Figure 3.1 shows the basic flow of the workings of all DMV offices. Typically, the office starts at 8:00 AM and ends at 5:00 PM with few minutes of overtime after 5:00 PM until the number of people in the system is zero.
Customers are not checked in after 4:00 PM for a road test or a computer test. Customers also can make appointments which have specific times and the number allowed is based on a particular office. Customers arriving at the office wait in line for the greeter to attend to them and, after checking to make sure all required documents are in order, give a ticket for that particular issuance. The customers wait in the waiting area to be served by an evaluator at a workstation. Depending on the size of the office, there are three to fourteen active workstations where workers process customers and decide whether they should be issued the particular type of license or identification. The processing time at the workstation depends on the type of issuance. In case of certain transactions, a customer may visit the workstation only once (i.e., permit and renewals without road test) and in certain issuances visit the workstation twice (i.e., customer requires a road test twice). If a customer requires a written test, he or she will go to one of the computer workstations to take the test. Customers who pass the test will go back to the waiting area to be seen by an evaluator; otherwise they will leave the system. If they pass all the steps for the type of issuance, they will enter the photo wait area to have their photos taken before leaving.
Figure 3.1 DMV Office Process flow Chart

Figure 3.2 shows the detailed flow of customers who require a road test. In cases where the person visits the workstation twice, there are different processing times. A customer first visits a workstation where the customer’s records are checked and then he or she is taken for the road test. After the road test, the customer returns to the same workstation with the evaluator. At that time, the worker again processes the customer (i.e., eye and sign test) which takes longer in case of new records and issuances.

Figure 3.2: Flow chart for customers taking a road test
In a DMV office, customers who just need a permit, CDL or a provisional license need to appear for a multiple choice question format test either on a computer or written. In this model it is described as a written test. Figure 3.3 describes the process followed by customers who are required to take only the written test. It demonstrates that first they see an evaluator and then head to the computer stations to take the test and return to the workstation to do the final eye and sign check before having their photo taken.

![Flow Chart for Customers Taking Written Test](image)

**Figure 3.3: Flow Chart for Customers Taking Written Test**

Further, some people visiting the DMV office to get State ID cards, reinstate a license, or for any other license related transaction follow a process as shown in Figure 3.4. This process map shows the process for those customers who are intending to get a State ID who first see a greeter who checks their documents at the check-in desk. Then, they have to wait in the waiting area before they head to a workstation where the evaluators check their history and legal presence. If they qualify, they are sent to have their photo taken. This process is similar for people who are getting a license reinstated and those categorized as other transactions.
Figure 3.4: Flow chart for customers needing a State ID card, reinstatement or other transactions

For renewals, the customer will only visit the workstation once, as seen in Figure 3.5 where the evaluators at a workstation check the person’s driving history, legal presence documents as well have the person perform the eye and sign test. If these tasks are successfully completed, then the license is renewed.

Figure 3.5: Understanding the system behavior based on the location

3.1.2 Understanding Worker Behavior

Even though this is not part of the mapping process, the behavior of the people in the system is extremely important for a simulation model. This type of information is generally not part of a Six Sigma process during the mapping process but might be collected later during the Analysis phase if found to be necessary. Some workers arrive to office before the office opens up at 8:00 AM and they take an hour lunch in batches of one or two between 11:00 AM to 2:00 PM based on office and number of staff members. If they are processing a
customer, they finish the transaction and then go to lunch. Further, if the worker is absent or not in the office then they are considered off shift. Under the present state, this schedule is practiced on a standard office schedule from 8:00 AM to 5:00 PM and workers wait until the number in the system is zero before closing the office. One observation which was not previously known by the Six Sigma team is that the workers will often take a ten to fifteen minute break or do additional assignments after every four to six customers seen.

3.2 Measure Phase

After the initial Define step, the team moves into the Measure phase which is used to determine the data needed to perform the analysis as well as establish a baseline for improvement. If necessary, the measurement system may need to go under analysis. Initial data can be collected from historical information if available, but in most cases one has to set up and collect new types of data at the correct resolution. It is very important at this junction in the DMAIC to have the simulation person as part of the team (For example – The team understood the importance of simulation at this point but it is very necessary to collect data from a simulation engineer point of view rather than an analyst point of view). Data collected for Six Sigma purposes may or may not be suitable to drive a simulation model. In order to avoid wasted time collecting data that cannot be used for simulation purposes or not collecting all the data needed to drive the simulation, the simulation team member should participate in the collection data plans at the earliest stages. In many instances the detailed process map helps to establish the data that is needed for the DMAIC as well as the simulation modeling. However, there are differences between collecting data for Simulation Modeling and Six Sigma. For Six Sigma, only data associated with the important
independent variables (i.e., x’s) or the potential cause factors of the process identified during brainstorming are collected while all the data associated with the process are needed to build an accurate simulation model of the system. Some of the data needed for a simulation model may not be important from a Six Sigma perspective. While Six Sigma is a data-driven process methodology, it does not often require the amount and level of data necessary to drive a simulation model.

3.2.1 Data Collection at the DMV

Based on the detailed process maps and understanding of the system, the following was the initial data needed for the simulation.

- Characteristics of all the Customer Types
  - Percentage and Arrival Patterns by Day
  - Processing Times of the Different Steps in the Process by Customer Type (i.e., Evaluator at Workstation, Road Test Time, Computer Written Test, etc.)
- Lunch Schedules of the Workers
- Number of Evaluators, Greeters for Each Office
- Failure Rates on Road Test, Eye Test, and Written Test
- Physical Layout of the DMV Offices
- Number of Customers who Balk from System
- Volume of Customers Based on Day of Week and Month

The following additional information was needed for the Six Sigma analysis.

- Waiting times at the various points in the system
- The number of customers at each point in the system

DMV had historic data based on an archaic tracking system to answer some of the questions (i.e., volumes of customers by day and month) but the system did not have different points of
entry; customers were not tracked once they were given a ticket and until when they were seen and then left. The team brainstormed to create a plan from a Simulation Modeling viewpoint as well as analysis required to be done by the Six Sigma team. The Six Sigma team had already created a data collection plan, but they had not yet collected the data. The simulation team was able to influence and make sure the data being collected could be used. The process of data collection was done with the method of handing a sheet of paper with different records and check boxes to identify the type of transaction/steps and the times as seen in Figure 3.6. The sheet was handed to each customer when he or she entered the DMV office. The greeter collected the sheet after processing the customer and noted the time. The person then waited in the waiting area until called by an evaluator. The evaluator collected the sheet and wrote the time he or she started processing the customer as well as time when finished with the customer. The customer either left the system which would represent the record at the exit or the customer waited in line to have the photo taken. Once the customer left the photo station, the worker noted the time. The final data point per customer is collected by a person at the exit of DMV. Figure 3.7 shows the data collection plan to determine the number of examiners available at a given time. The data was collected for a week and was assumed to be representative for the whole year.
Figure 3.6: Data Collection Plan Worksheet
<table>
<thead>
<tr>
<th>Staff ID:</th>
<th>Examiner Office</th>
<th>Examiner</th>
<th>Examiner</th>
<th>Examiner</th>
<th>Examiner</th>
<th>Examiner</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7:45</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8:00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8:15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8:30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8:45</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9:00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9:15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9:30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9:45</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10:00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10:15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10:30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10:45</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11:00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11:15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11:30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11:45</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12:00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12:15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12:30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12:45</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1:00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1:15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1:30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1:45</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2:00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2:15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2:30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2:45</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3:00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3:15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3:30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3:45</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4:00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4:15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4:30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4:45</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5:00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5:15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5:30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5:45</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6:00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6:15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6:30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6:45</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 3.7: Examiner Availability Data Collection Plan**
3.2.2 Issues with the Data

Figure 3.8 shows an example of how the collected data had some problems and how simulation served in solving those problems. The first problem was the incomplete or unexpectedly high wait times were validated using simulation models. This helped in identifying the root cause of the problem which was that a few customers who arrived to the office but forgot some documents or left the office for other reasons were not checked out or accounted for, so their wait time was very high. Another important visual validation was in case of a person going for the road test. Here a person visits the workstation once and then takes a test with the evaluator. Then he comes back to the same workstation and is again evaluated for the eye and sign test. This shows that the person has two processing rounds at the workstation, but the data collected showed only one round for the time. The other round was not considered. The third problem was that the greeter check-in time accounted was a sum of waiting and processing time which caused huge impact on total time in the system. The fourth problem was that the worker availability data was collected at specific intervals rather than accounting for breaks. For example, number of workers present at 9:00 AM and then at 9:15 AM was checked, but the data do not show whether a worker returns from a break at 9:01 AM and again goes for a break at 9:14 AM, in which case he/she is considered off-shift for 30 minutes from 9:00 AM to 9:29 AM, but he or she has actually worked for 13 minutes.
Figure 3.8: Problems with some of the collected data and how simulation served useful in finding solution to the problems

One major difference in collecting data for simulation versus Six Sigma analysis is that Six Sigma often combines processing time and wait time together when the processing time is so small in comparison to wait time. While the simulation team was explicit about needing the actual processing times, some of the data (e.g., greeter processing time) that was collected combined service time (i.e., processing time) and wait time together. However, for the simulation model, the queuing times are calculated and one only needs processing time.

The initial simulation model built during the beginning of the Analysis phase served as an important tool to identify other necessary input data. For example, the evaluation time that was collected involved only one processing time for a customer, but customers can visit the evaluator more than once based on the customer type, which could have created a false representation of the system. In this case, if Simulation Modeling was not used, then the true total service time would not be identified. For example, suppose a customer wants to take a computer test. He first visits the evaluator station, then proceeds to a computer test and if he
passes, he again goes back to the evaluator, which again adds processing time which was not collected. If the data was to be recollected, then there would have been a huge time and money loss to DMV; so, using simulation, certain data was assumed based on some pilot readings and time study to identify the double processing time. This gave a real representation of the system. Therefore, Simulation Modeling can be used as a tool to determine the necessary data and correct any problems in a Measure phase of a DMAIC process.

### 3.2.3 Input Modeling

Again, the Six Sigma process is not a sequential process, meaning the initial simulation model was built during the beginning of the Analyze phase while the data was being collected as well as identifying any new data that is needed. Once the data is collected, the data needs to be cleaned. After it is cleaned, a data distribution is fitted to the processing times data. In order to fit the data to a distribution, any fitting software as mentioned in Section 2.4.2.2 can be used to fit the distributions. This research utilized the EasyFit™ fitting software because of its flexibility and ability to translate EasyFit™ parameters into the SIMIO™ parameters using the following method:

- Collect sample data with sufficient number of data points;
- Use the EasyFit™ software to Fit the Distribution;
- Analyze the Goodness of Fit based on the rank and null hypothesis if the number of data points was less 100; otherwise the PP plots were also used to choose the best; and
• Based on the best distribution by EasyFit™, collect the parameters and create corresponding SIMIO™ expression.

Figure 3.9 shows the fitted distribution (i.e. PDF and CDF) for the processing time at the workstation for customer type associated with *Other Transactions* processing times at the Charlotte office. The corresponding values of the Johnson bounded distribution parameters are also shown in the figure which involves $\gamma$, $\delta$, $\xi$, $\lambda$ where $\gamma$ and $\delta$ are the shape parameters, $\lambda$ is the location parameter and $\xi$ is the scale parameter.
Figure 3.9: Fitted Distribution for Other Transaction Customer Type

Table 3.1 shows the distributions that were used to model all the randomness for the Charlotte office. All the distributions are fitted based on the data collected during the last week of November or based on expert opinion. All of the EasyFit output data can be seen in Appendix A. In some instances where the data collected combined both the processing time and waiting time (i.e., greeter check in and photo taking), expert opinion as well as a
validation via the system was used to model those situations using BetaPert distributions where a minimum, maximum, and most likely case were determined by the team.

Table 3.1: Distributions that Were Used to Model the DMV

<table>
<thead>
<tr>
<th>Variable Description</th>
<th>SIMIO™ Random Expression</th>
<th>Determined</th>
</tr>
</thead>
<tbody>
<tr>
<td>State IDs</td>
<td>LogLogistic(3.33, 4.16)</td>
<td>Fitted</td>
</tr>
<tr>
<td>License without any test</td>
<td>Lognormal(1.7425, 0.64473)</td>
<td>Fitted</td>
</tr>
<tr>
<td>License before Road Test</td>
<td>3</td>
<td>Expert Opinion</td>
</tr>
<tr>
<td>License After Road Test</td>
<td>Lognormal(1.786426, 0.839898) + .27</td>
<td>Fitted</td>
</tr>
<tr>
<td>License after Road test and written Test</td>
<td>Lognormal(1.786426, 0.839898) + .27</td>
<td>Fitted</td>
</tr>
<tr>
<td>License after written test before road test</td>
<td>Lognormal(1.786426, 0.839898) + .27</td>
<td>Expert Opinion</td>
</tr>
<tr>
<td>Before Written Test</td>
<td>Lognormal(1.786426, 0.839898) + .27</td>
<td>Expert Opinion</td>
</tr>
<tr>
<td>Written test/Computer Test</td>
<td>Lognormal(1.786426, 0.839898) + .27</td>
<td>Fitted</td>
</tr>
<tr>
<td>Written CDL</td>
<td>JohnsonSB(0.18151, 0.41229, 0.68887, 13.23587)</td>
<td>Fitted</td>
</tr>
<tr>
<td>Permits</td>
<td>Lognormal(0.62625,1.8545)</td>
<td>Fitted</td>
</tr>
<tr>
<td>Appointment</td>
<td>Lognormal((1.786426, 0.839898) + .27)</td>
<td>Fitted</td>
</tr>
<tr>
<td>Others Transactions</td>
<td>JohnsonSB(1.0106, 0.63227, 0.68841, 5.9143)</td>
<td>Fitted</td>
</tr>
<tr>
<td>Before Written Provisional I</td>
<td>Gamma(3.2009, 1.4328)</td>
<td>Fitted</td>
</tr>
<tr>
<td>After Written Provisional I</td>
<td>Gamma(3.2009, 1.4328)</td>
<td>Fitted</td>
</tr>
<tr>
<td>Provisional III</td>
<td>Gamma(3.2009, 1.4328)</td>
<td>Fitted</td>
</tr>
<tr>
<td>Motorcycle</td>
<td>Gamma(1.75, 2.8571)</td>
<td>Fitted</td>
</tr>
<tr>
<td>Reinstate</td>
<td>Lognormal(1.786426, 0.839898)+.27</td>
<td>Fitted</td>
</tr>
<tr>
<td>Road Test</td>
<td>Triangular(12.5, 17, 27.5)</td>
<td>Fitted</td>
</tr>
<tr>
<td>Written Test</td>
<td>Poisson(19.8)</td>
<td>Fitted</td>
</tr>
</tbody>
</table>

3.3 Analyze Phase

Based on the initial data collected as well as anecdotal evidence from evaluators, the team identified that the time in system for the customers in the three offices was higher than
expected and what was deemed acceptable. Recall, the Analyze phase of the Lean Six Sigma methodology tries to uncover the potential root causes of the problem through various tools like five whys, statistical data analysis (i.e., correlation, regression, Pareto plots), design of experiments on the real system, etc. If Simulation Modeling is to be employed, the current state simulation model should be built during the beginning of the Analyze phase. The detailed process map and understanding of the problem from the Define stage acts as the beginning point for the initial simulation model. The simulation model building can be started in parallel before the Measure phase is completed and often leads to determining additional data that needs to be collected during that phase. If Simulation Modeling is determined as an appropriate tool right after the Define stage, the initial rough cut model can be built to help determine potential root causes as well as potential data collection. Building the first model of the system forces the team and/or the simulation designer to fully understand the system. Often additional observation and walking the real system (i.e., Gemba) may be needed to fully understand the subtleties of the system being modeled. Design of experiments for data collection and determination of root causes is often used, but in many cases a full DOE on the real system is not practical owing to cost, resources or other constraints (e.g., health care setting or the real system does not exist). Therefore, the current state simulation model of the system can be used for these purposes. To help expedite the process it is important the simulation team be fully involved in the Six Sigma team, as information learned during the development of the model and analysis of the day may benefit either team.
3.3.1 Building the Initial Current State Simulation Model of the DMV

The Six Sigma team already knew ahead of time during the Define and Measure phases that Simulation Modeling was going to be used to help identify potential improvements. Therefore, the initial models were built using the detailed process map from Figure 3.1. The Six Sigma team noted three potential problem offices in the state (Charlotte, Greensboro, and Durham) for potential models. All three offices were visited and observed to aid in the simulation modeling. Also, the Six Sigma team brainstormed potential improvements and wanted the ability to change schedules of personnel, the number of evaluators, etc. Other improvements were discovered after the initial model was built, but these ideas of improvements at this junction helped to shape the model building. Also, the Six Sigma process identifies the metrics used to judge the problem system during the Measure phase which are then used in the simulation model. One advantage of using the DMAIC process in conjunction with simulation rather than Simulation Modeling alone is the identification of the metrics and input variables is built into the methodology.

3.3.2 Simulation tools and software used

The project definition was to build a generic simulation model with 3D animation of the DMV offices with a plan to identify the real-time behavior of the system as well as the model to ascertain impact of improvements. At this junction, the type of simulation model as well as the software used to build the model needs to be determined. In the case of the DMV, a simple Monte Carlo simulation of the process map would not be sufficient owing to the movement of the DMV workers, the waiting times, etc. Therefore, a discrete event simulation (DES) model was deemed the most appropriate. There are dozens of different
DES software on the market as well as home grown. SIMIO™ was chosen as the modeling software owing to its 3D modeling capability, object-oriented framework, sophisticated experimentation and optimization platform to evaluate what-if scenarios, and a friendly user interface. Since the DMV system deals with interaction of people, SIMIO™ offers the ability to model moveable resources as the DMV evaluators move from different workstations as well as accompany customers on road tests. One important advantage of using SIMIO™ is its ability to create custom objects or modify the existing SIMIO™ objects based on the desired logic owing to the object-oriented framework. Like many discrete simulation languages, SIMIO™ provides a basic set of objects the modeler uses to build simulation.

**Entities** have a number of characteristics like size, shape and volume and represent the objects (i.e., customers in the DMV) that move throughout the system to be serviced. They are created dynamically and their behavior is dictated by the network of the system, their properties, and the current state of the system. **Worker** is an entity type that is a moveable resource that can move throughout the system but can also be used as a transporter. The worker object will be used to represent the DMV evaluators who move between stations, road test, and are off shift. The **Source** object creates entity objects based on an arrival pattern and workers. The **Server** is a fixed object in SIMIO™ which is used to delay an entity with variable capacity which may causes entities to queue up in front of the server if it is currently processing another entity or is off shift. The **Sink** is a fixed object used to destroy the entity (i.e., represent the exit of the DMV) and also collects statistics automatically.

Objects in SIMIO™ can have two types of characteristics properties and state variables. Properties are characteristics (e.g., initial priority of an Entity) which cannot change
throughout the simulation and are typically set at the beginning of the simulation for static objects (i.e., Servers, Sinks, Sources, etc.) or set when dynamic objects are created (i.e., Entities, Workers, etc.). State variables on the other hand can have their value changed throughout the run of the simulation.

Because these models are often being built before all the data is collected, it is important to design the model with as much flexibility as possible from the beginning as many things may change. All inputs that could be changed were modeled as properties or input parameters to the simulation model (i.e., number of workers, passing rates of the written and road test, the processing times of the various steps). No input was hard coded with in the properties of the objects or with in the process logic. Using SIMIO™ data tables to model customer types, worker schedules, appointments, etc. was also employed so these could be easily changed and will be described below.

### 3.3.2.1 Data

Since the offices are basically the same except for layout and number of workstations, the models need to be data driven to reflect the different customer mix, changes in processing times, etc. Table 3.2 shows the fields of the customer type table while Figure 3.10 shows a portion of the actual SIMIO™ table for the customer mix. As can be seen there are basically fourteen basically different classes of DMV customers with a repeat for appointment customers. To facilitate the use of the same processes for both regular customers and appointment customers throughout the model, the same table was needed. As can be seen, the *Probability* field is zero for appointment customers so they are not generated by the normal arrival process and the *AppointmentMix* probability would be zero for regular customers, so
only appointment customers are created by the appointment process. To facilitate the DMV’s
desire to route certain classes of customers to certain workstations (for example., only people
needing road test are sent to the first three evaluators who are more experienced), the
$LstAvailableWorkstations$ field specifies which list of workstations the particular customer
should use to route to the next available evaluator at a workstation.

**Table 3.2: Fields of the Customer Table**

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UniqueID</td>
<td>Unique identifier and row determination</td>
</tr>
<tr>
<td>Probability</td>
<td>Customer Mix Percentage used to determine customer type upon arrival (Type As occur 9.2 percent of the time)</td>
</tr>
<tr>
<td>Type</td>
<td>Different Classes of Customer and the primary key. Note appointment types begin with F</td>
</tr>
<tr>
<td>DescriptionofType</td>
<td>Description of the Customer Class</td>
</tr>
<tr>
<td>LstAvailableStations</td>
<td>The workstation they can be sent to.</td>
</tr>
<tr>
<td>AppointmentMix</td>
<td>Same Customer mix percentage but for customers with appointment as they have higher priority than nonappointment customers.</td>
</tr>
<tr>
<td>TStatTotalTimeInSystem</td>
<td>Tally statistics to collect Time in system for all individual types of transaction</td>
</tr>
<tr>
<td>TstatTimeWaiting</td>
<td>Tally Statistics to collect the Time waiting in the queue for all individual transaction type</td>
</tr>
</tbody>
</table>
Table 3.3 describes the fields of the customer processing time table which is used under the Data tab of SIMIO™. The first column represents the type of customer for example, whether the customer needs a license, state ID, or permit. The priority column assigns a priority to each customer type based on the real system priority structure. For example, if a customer has taken a computer test already then he would not have to wait very long like the customers who just entered the system; rather he would be given a higher priority than others. Figure 3.11 and Figure 3.12 show the priority assignment column for each type of customer and processing times for each type of customer. InpStateIDproc is the input parameter which is used to gauge the sensitivity of the input on the output while the Modelentity.StaSLE is the constant number multiplied to assign 10% of customers who have English as Second Language. The columns showing Road test and Computer test are check boxes which when
checked mean the customer needs to take a road test or a computer test. Figure 3.12: Customer Processing Table as a Related Table to the Customer Table shows the relational table which relates each customer type to another table. Here customer type is assigned as a primary key. Further, the row *Lst Available Stations* shows the available stations for particular type of transaction. For example, workstation 2 and workstation 3 take customers who want to take a road test while other workstations evaluate customers for any other transaction. This is done by adding the workstations in a different list.

**Table 3.3: Customer Processing Time Table**

<table>
<thead>
<tr>
<th>Description</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>WorkstationTimesStages</td>
<td>Type of Customer (Foreign Key)</td>
</tr>
<tr>
<td>Priority</td>
<td>Priority</td>
<td>Description of which stage of the customer</td>
</tr>
<tr>
<td>ProcessingTimes</td>
<td>ProcessingTimes</td>
<td>Priority associated with the customer at this stage</td>
</tr>
<tr>
<td>RoadTest</td>
<td>RoadTest</td>
<td>The processing time associated with the customer type for this stage</td>
</tr>
<tr>
<td>ComputerTest</td>
<td>ComputerTest</td>
<td>Is this stage a road test?</td>
</tr>
<tr>
<td>RowNumber</td>
<td>RowNumber</td>
<td>Is this stage a computer test?</td>
</tr>
<tr>
<td></td>
<td>Description</td>
<td>A row identifier used in the mode to know what is the next stage for this customer</td>
</tr>
</tbody>
</table>
### Figure 3.11: Customer Processing Time and Priority Table

<table>
<thead>
<tr>
<th>Type</th>
<th>Workstation Times Stages</th>
<th>Priority</th>
<th>Processing Times (Minutes)</th>
<th>Road Test</th>
<th>Computer Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A</td>
<td>9</td>
<td>InpStateIDProc * ModelEntity, StasSLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>B1</td>
<td>10</td>
<td>InpLicenseTestWorkStation * ModelEntity, StasSLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>B2</td>
<td>6</td>
<td>InpBeforeRoadTest * ModelEntity, StasSLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>B2</td>
<td>16</td>
<td>InpLicWorkStation * ModelEntity, StasSLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>B3</td>
<td>5</td>
<td>InpLicWorkStation * ModelEntity, StasSLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>B3</td>
<td>17</td>
<td>InpBeforeRoadTest * ModelEntity, StasSLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>B3</td>
<td>18</td>
<td>InpLicWorkStation * ModelEntity, StasSLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>C</td>
<td>5</td>
<td>InpLicWorkStation * ModelEntity, StasSLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>C</td>
<td>15</td>
<td>InpLicWorkStation * ModelEntity, StasSLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>D</td>
<td>4</td>
<td>InpCDLWorkStation * ModelEntity, StasSLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>D</td>
<td>14</td>
<td>InpCDLWorkStation * ModelEntity, StasSLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>E</td>
<td>8</td>
<td>InpPermitWorkstation * ModelEntity, StasSLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>E</td>
<td>13</td>
<td>InpPermitWorkstation * ModelEntity, StasSLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>F</td>
<td>20</td>
<td>InpLicWorkStation * ModelEntity, StasSLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>G</td>
<td>7</td>
<td>InpOtherWorkstation * ModelEntity, StasSLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>H</td>
<td>2</td>
<td>InpProvisionalWorkStation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>H</td>
<td>12</td>
<td>InpProvisionalWorkStation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>I</td>
<td>3</td>
<td>InpProvisionalWorkStation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>J</td>
<td>1</td>
<td>InpMotorcycleWorkStation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>L</td>
<td>4</td>
<td>InpLicWorkStation * ModelEntity, StasSLE</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Figure 3.12: Customer Processing Table as a Related Table to the Customer Table

#### 3.3.2.2 Modeling DMV Evaluators

DMV Evaluators are required to do many types of jobs in the office. They may be the greeter to check-in people to make sure they have the right documentation; the person taking the photo for IDs, permits, and licenses; and the evaluator who performs the eye and sign test, road test, and determines if the customer is allowed to receive a license, etc. As it was mentioned, the evaluators need to move from workstations to become the greeter or photo...
taker or to take customers on road tests. Therefore, the SIMIO™ Worker moveable resource object will be used but modified to create a new worker to help with the DMV scenario. Table 3.4 describes the additional properties and state variables added to the standard Worker object. Currently, the Worker has the ability to specify a home node so that when the worker goes idle or goes off shift, the Worker will move to that home location. However, in the DMV situation these two abilities needed to send them to different locations. If idle the worker would return to the assigned workstation but off shift the worker would leave the system as well as not be available. To handle this situation, two new state variables and a property are needed to switch the home node to the actual home node if the is going on shift but switch it to the off shift location node if the worker is going off shift. These changes are implemented in a modified OnCapacity Changed process event of the Worker processes.

Table 3.4: Added State Variables and Properties of the Worker Object

<table>
<thead>
<tr>
<th>State Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>StaActualHomeNode</td>
<td>This state variable will hold the actual home node where the worker should be normally stationed to perform work (i.e., workstation). This is necessary as the default HomeNode state variable is reassigned during the simulation to be the OffShiftNode or the Home Node.</td>
</tr>
<tr>
<td>StaOffshiftNode</td>
<td>This state variable identifies which node should be used when the evaluator goes off shift (i.e., break, lunch, etc.).</td>
</tr>
<tr>
<td>StaSendToOff</td>
<td>This state variable determines if the worker has gone off shift but needs to finish the current customer either on the road test or at the workstation. When the worker finishes processing the current customer and the value is one, the capacity is changed to zero, sending the worker off shift.</td>
</tr>
<tr>
<td>StaWhichOne</td>
<td>This state variable is used to determine which number this worker is in the</td>
</tr>
</tbody>
</table>
The modified Worker object is being used to model the DMV evaluators. In order to allow the decision maker to specify the number of evaluators that are available on a given day, one instance of the worker object will be used and a Source will be used to create that number of evaluators. Since only one worker object is instantiated, only one home node for all the workers is created, which is not the case for the DMV. To be flexible, the table of home nodes seen in Figure 3.13 is used to specify the home node for each of the evaluators where the Which Col field is linked to the StaWhichOne state variable of the worker described in Table 3.4.

<table>
<thead>
<tr>
<th>Home Node</th>
<th>Which Col</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>13</td>
<td>13</td>
</tr>
<tr>
<td>14</td>
<td>14</td>
</tr>
</tbody>
</table>

Figure 3.13: Home Node Assignment Table for each Worker Based on the Worker ID
Figure 3.14 shows the process that is executed each time a Worker is created from the source **SrcWorker**. The first **Assign** step assigns which worker number (i.e., first worker receives a value of one) to the StaWhichone state variable of the worker. Again, the state variable allows one to determine this worker out of the population of workers. The next Assign step assigns the standard Home Node state variable the home location from the appropriate row from the home table seen in Figure 3.13. Next the process assigns the actual home node for the worker to be the same, as the home node as the home node state variable will be changed when the worker goes off shift. Finally, the worker is sent to the home node to begin the simulation.

**Figure 3.14: Process Executed When each Worker is Created**

Because the evaluators are modeled by the worker and consist of a population, the current SIMIO™ capability is to treat the entire population the same in terms of schedule and home nodes. Figure 3.15 shows a time indexed table in 30 minute increments which is used to dynamically change and optimize the worker schedule. As shown in the figure, row 15 represents the 7:00 AM to 7:30 AM time slot and workers one through five will come start at 7 AM owing to the value of one while a zero indicates an off shift condition. The schedule table can also be used to identify the best schedule by specifying a property instead of a direct zero or one. For example, the property “P137” refer to the worker number 13 and seven means the start time is 7 AM. The property can be used to optimize the schedule for temporary workers by allowing it to be zero or one. Multiple scenarios with different values
for these properties can be run and determine when is the best time for workers to start. Further, the lunch times were optimized to decrease the average time in system during lunch hours using staggered scheduling pattern for lunch hours. Because the table is time indexed, the process in Figure 3.16 will be invoked at the beginning of each time interval (i.e., each half hour). First, the process determines the current row of the table. Next the Search step is used to loop through all workers in the population and assign the capacity associated with that worker based on the StaWhichOne state variable.

<table>
<thead>
<tr>
<th></th>
<th>W1</th>
<th>W2</th>
<th>W3</th>
<th>W4</th>
<th>W5</th>
<th>W6</th>
<th>W7</th>
<th>W8</th>
<th>W9</th>
<th>W10</th>
<th>W11</th>
<th>W12</th>
<th>W13</th>
<th>W14</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>12</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>13</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>14</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>15</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>p127</td>
<td>p137</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>p127</td>
<td>p137</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>Time range for row 15: 12/1/2014 7:00:00 AM to 12/1/2014 7:30:00 AM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>p128</td>
<td>p138</td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>p148</td>
<td>p148</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 3.15: Worker Schedule Table**

![Diagram](image)

**Figure 3.16: Process used to update the Capacities of the Workers for Every Time Interval of the Worker Schedule Table**
Figure 3.17 shows the process logic used for workers to wait in the system until the number of customers in the system is zero after 5:00 PM. If the worker goes off shift at or after 5:00 pm and the number of customers in the system is greater than zero, the system will change the capacity back to one, placing the worker back on shift. Then after ten minutes the process will recheck to see if customers still are present in the system. Once the criteria is false then the system will shut down.

![Diagram](image)

**Figure 3.17: Waiting for the Number in System to go to Zero After 5:00 PM**

### 3.3.2.3 Modifying the Server and Model Entity

The standard **Server** object was sub-classed to better fit the DMV system by modifying a few processes and adding a new state variable. The state variable **StaAvailable** which can take on values of zero or one is used to decide whether the workstation is available (i.e., a DMV evaluator is present) and can accept the next customer if there is currently no customer being served. If an evaluator (i.e., SIMIO™ worker) is not present at the workstation, the value is set to zero indicate that a customer cannot be served by this workstation. Often you have more workstations than available evaluators in the system ( due to breaks, lunches, time of day, etc.). When the assigned worker enters (e.g., worker six in Figure 3.18) the specified node , the program will set the status variable StaAvailable of the associated server (e.g., srvWstn4) to one and zero when it exits the node. The generic SetServerAvailability process
seen in Figure 3.19 is used for all workstation nodes and workers. This process is used by assigning worker and server workstation along with the value as seen in Figure 3.18. The process first decides if his is the correct worker. If this is true, then it sets the new server’s StaAvailable state variable to zero or one depending on the value passed to it. If the server is available, then worker is told to allocate itself to the set of customers.

<table>
<thead>
<tr>
<th>Entered</th>
<th>SetServerAvailability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input Arguments</td>
<td></td>
</tr>
<tr>
<td>Server</td>
<td>srvWstn4</td>
</tr>
<tr>
<td>Set Value</td>
<td>1</td>
</tr>
<tr>
<td>Which Worker</td>
<td>6</td>
</tr>
<tr>
<td>Exit</td>
<td>SetServerAvailability</td>
</tr>
<tr>
<td>Input Arguments</td>
<td></td>
</tr>
<tr>
<td>Server</td>
<td>srvWstn4</td>
</tr>
<tr>
<td>Set Value</td>
<td>0</td>
</tr>
<tr>
<td>Which Worker</td>
<td>6</td>
</tr>
</tbody>
</table>

**Figure 3.18: Specifying the SetServerAvailability Process on the Entering and Exiting of a Node**

**Figure 3.19: Set Server Availability Generic Process**

For the simulation model, Table 3.5 describes the additional state variables (i.e., characteristics) given to the generic ModelEntity object and used in the simulation to process and that route the customers throughout the system along with the assigned tables.
Table 3.5: Added State Variables of the ModelEntity

<table>
<thead>
<tr>
<th>State Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>StaPass</td>
<td>This state variable is used to assign whether a customer has passed or failed the written test or eye/sign test where a value of one indicates a pass.</td>
</tr>
<tr>
<td>StawhichWrkerSeized</td>
<td>This state variable identifies which worker is processing the model entity (customer) at a workstation. The state variable is used to request the same evaluator for the road test who has seen the customer at the workstation.</td>
</tr>
<tr>
<td>StaSLE</td>
<td>A value of 1.2 indicates that English is a second language for this customer; otherwise it is set to one. It generally takes 20% longer to deal with customers who are not native English speakers.</td>
</tr>
<tr>
<td>StaRoadFail</td>
<td>This state variable is used to assign whether a customer has passed or failed the road test where a value of one indicates a pass.</td>
</tr>
</tbody>
</table>

3.3.2.4 Greeter and Photo Area

In some offices, there are dedicated greeters and photo personnel while in some offices these duties are shared by the evaluators. Even when there are dedicated personnel, the evaluators may be called to do these services when these personnel are currently off shift (i.e., taking lunch or gone for the day). Since the normal evaluators are assigned a particular workstation, SIMIO™ default logic would be to send them home if they are idle or to handle requests of the evaluator’s time separately. To prevent a normal evaluator to be requested to greet or take a photo if there is currently some other personnel present at these stations, the model needs to have the worker reject the request. Also, if the worker is processing at the greeting or photo station and there are more people in line, they should not go back to the assigned workstation but keep processing customers.
Figure 3.20 shows the parameterized process for workers to accept or reject a request of service by a customer. The parameterization is used to pass which worker is being requested out of the population. If the customer who is requesting the service is not at the check in or photo stations, the process then checks to see if this worker is currently at the check in station. If the worker is and more people are waiting to check in, the worker will reject the request; otherwise, the worker accepts the request of service. If the worker is not at check in and the false path is taken, the process will check to make sure the worker is not at the photo station. If the worker is not, then it follows the true path to search to make sure there is no one else capable of servicing the entity. If there is, then this worker rejects the process; otherwise the worker accepts the request. If the worker is currently at the photo station and more people are in the photo line, then the worker rejects the request; otherwise the worker accepts the request. If the customer requesting the service is at the check in or photo stations (i.e., the false path of the first Decide), the process will search to see if the dedicated greeter and photo taker are on shift. If they are on shift and this worker is not the dedicated greeter and photo taker, then the worker will reject the request. Now if one or both dedicated personnel are off shift or the worker is one of the dedicated (i.e., Decide2 step), the process will search to make sure no other worker is at either the check in or photo station where the customer is requesting the process or potentially headed that way. If there are other workers that can handle the request, the worker rejects the request; otherwise the worker will accept it.
3.3.2.5 Modeling the Customer Behavior at Workstation

Figure 3.21 shows the two parameterized processes when customers enter and exit a workstation. The *Entering the Workstation* process is triggered when the entity enters the workstation. The process is passed which evaluator to seize and where the evaluator should be moved to. The Assign step assigns the ID of the evaluator that is processing the customer to a state variable which will ensure the same evaluator does the road test if required.

After processing the customer at the workstation, the *OutputExitedFromWorkstation* process will be executed. The first Decide determines if the customer needs to take a road test. If the customer has to take a road test, then the customer will be sent back to the same workstation when he or she returns from the road test via the Set Node step. Then, it executes
the step for incrementing the sequence (i.e., linking to the next row in the related customer processing table seen in Figure 3.12) which sets the next processing time as well as changes the priority level. The next step, Execute, moves the worker (i.e., evaluator) with the customer outside to perform the road test without releasing the evaluator. If the next sequence is not a road test (if no road test is needed or the road test has already been performed), the customer destination node is set to the node leading to the either the photo station or the exit if the customer has failed any of the tests. If the evaluator went off shift while transporting the customer for the road test (i.e., the StaSendToOff of the worker is set to one), then the capacity of the worker is set to zero, sending it off shift before the evaluator is released back to allow for service. If the customer failed either the written or road test, then the overall StaPass is set to zero; otherwise, the eye and sign test is done which determines if the customer has passed all requirements.

![Diagram](image)

**Figure 3.21: Generic Process Logic for Customers Entering/Exiting Workstation**
3.3.2.6  Modeling and Animating the Road Test

Modeling the road test time is handled by a time path in SIMIO™ which allows for movement along a path for a specified random time. Rather than use a delay or another server to delay the time, the time path provides a visual of the customer(\textit{entity}) and evaluator(\textit{worker}) going along the path. Also, the evaluator will be used like a transporter and the customer will ride on the evaluator along the time path so they are in the same vehicle. From the previous section, recall that if the customer needs to take a road test, this will cause the evaluator to move to the road test node outside the facility. This is done to provide a visual of the evaluator moving from the current workstation to the outside without releasing the evaluator to prevent them from being seized by another customer. In Figure 3.22, the first step checks to make sure the object entering the node is the customer and not the evaluator who was sent to the same node. Next the customer is assigned whether they pass the road test or not, and then they request a ride from the evaluator which was assigned in the EnteringtheWorkstation in Figure 3.21. If the current evaluator was scheduled to go off shift while they are processing the current customer, the StaSendOff is set to one to send the evaluator offshift when he or she finishes with the current customer and then the evaluator’s capacity is changed back to one to place the evaluator back on shift. At this point the customer releases the evaluator and then worker logic will allow the evaluator to grab the customer for transport. At this point the customer will ride on the evaluator around the time path for the appropriate time. However, from an animation standpoint this will look odd as they will look like two people going around the track. Figure 3.23 shows the logic that
changes the person into a car for animation purposes by searching the ride queue of the transporter to access the customer.

Figure 3.22: Customer Reaches the Outside for the Road Test

Figure 3.23: Changing the Picture to look like a Car

Recall from Figure 3.21, the customer has specified the workstation to be returned to by the evaluator (i.e., the worker will transport the customer back to the workstation). Once the evaluator arrives back to the entrance of the office, the process in Figure 3.24 is invoked. A similar search is done to turn the customer back into a person for animation purposes. If the worker (i.e., evaluator) goes off shift while transporting (i.e., on the road test), the model again assigns the StaSendOff to one and puts the worker back on shift until the worker completes the current customer. The system will automatically allow the evaluator to transport the customer to the workstation, and the logic in Figure 3.21 will take over.
3.3.2.7 Balking and System Failure Reneging Logic

In a DMV office, people arrive and get in the main check-in queue. Suppose there are more than a certain number of people present in the queue, and the customers do not wait in queue but instead just leave/balk from the system. This decision of not entering the queue based on the length of the queue is called balking. Figure 3.25 shows the balking logic used for customers entering the DMV. The process will check to see if the number waiting in the check-in line plus those in the waiting area is greater than or equal to a random balk number (e.g., InpCheckInBalk or Pert(20, 25, 30)). If the number in the system exceeds the input balk number, the customer will leave the system; otherwise it enters the check-in queue.

In a DMV office, all systems are connected to the main system (i.e., central database). Sometimes, the system breaks down for 30 minutes to 90 minutes. If the system breaks down, customers are told to wait if they are waiting to get a photo or processing at the workstation is in progress; otherwise they are sent home. This system failure logic assumes
that if the breakdown occurs at 4:45 PM which is just 15 minutes before office closes, then anybody waiting to get processed at the workstation or check-in queue will leave the office since it is very unlikely the system will be available before close. Further, if the system breaks down, then all computer systems are disconnected and customers have to wait until they are fixed. Figure 3.26 shows the process logic for system breakdown. The down time or time for the system to be operational again is modeled as a Triangular distribution with a minimum of 10 minutes, a most likely time of 30 minutes, and a maximum of 90 minutes. The down time is determined first and then a table of all workstations and other objects is searched and each one is failed. The next Decide step determines if the time is 4:40 and after; if so then the SystemBreakDownRemove process is executed to remove everyone from the system that is not currently at a workstation. The each failed object is delayed the same down time and then repaired.

Figure 3.26: System Break Down Process and Removal
The process logic in Figure 3.27 will remove anybody from the system who is waiting to do a road test at 4:45 PM, as they will not be able to complete the entire process before the system shuts down.

![Figure 3.27: Remove all Customers Waiting for a Road Test](image)

### 3.3.3 Validation and Verification of the Model

Once the base model is built, the last step in the Analysis associated with the simulation model is to validate the model. Verification and validation are necessary to ensure the base model accurately models the real system before using the model to generate solutions to problem in the Improve phase that was prepared to verify the time in system. The processing times, arrival times and other required data were collected for a week and it was identified that the average time in system was not as high as was observed in the real case. After careful revision of data of the customers whose total time in system was high, it was observed that most of the time spent by these customers was waiting in the waiting area, which resulted in the average time in system to be high. When reviewed, it was identified that these customers
had forgotten a few documents and so left the office to get them or called somebody to get the documents for them. Due to this reason, they waited in the system for more time. The DMV system did not directly cause the unreasonable time in the system for these customers. Further, it was identified from the simulation model that the time in system was highly sensitive to the check-in processing time. Figure 3.28 shows the graph of sensitivity of the input parameter on the output responses that are of interest. In the graph it can be noted that the output is very sensitive to the check-in time. The second most influential parameter in output responses is the photo service time. One important part of the simulation model was that both of these inputs were not properly recorded, so the data input analysis was not valid. Initially, the data collected involved both service and wait time which were not helpful for simulation; hence, an assumption was made to properly gauge these inputs based on the collected data values and pilot readings. The anecdotal evidence was used with a Pert distribution with a minimum of 0.5, a most likely of one, and a maximum of three minutes for the check in processing time and a minimum of 0.5, most likely of 1.5, and a maximum of four minutes for the photo taking. These input assumptions were verified by the DMV personnel as well.
Further, as a part of validation, the real data and the simulated output data was compared. After running 30 replications, the confidence interval of the mean of the total time in system is $25.67 \pm 2.1$ minutes as seen in the SMORE plot of Figure 3.29. The number of replications are randomly decided but can be increased for better confidence. Comparing this to the real data, the confidence interval is $26 \pm 7$ minutes which overlaps with the simulated data. Therefore, considering the logical analysis of the random system variation owing to people’s behavior, it would be considered the best representation of the real system.
The DMV office members together with the Deputy Commissioner validated the simulation model by visually verifying the behavior of the system and the verification of the various data points with the real data. A fully animated simulation model as seen in Figure 3.30 was developed for this portion and is important at this stage of the Analyze phase but takes an enormous amount of time. The model was also visually verified based on the animations and the processing time collected from the system.
Figure 3.30: Animated Model of Charlotte DMV Office
Chapter 4
Data Analysis and Results

The previous chapter outlined the methodology used for building the basic simulation model to create a framework to integrate all the required utilities in the model during the first three phases of the DMAIC process. The simulation model was created with generic programs with use of Tokens in SIMIO™ which helped in decreasing redundancy in the model processes during the Analyze phase. The present state and the future state of the DMV office were simulated with the idea of validating and verifying the changes suggested by the Lean Six Sigma team. The simulation was made flexible enough to give the ability to optimize the number of workers and their schedule which helped in understanding the present bottlenecks and addressing the issues. This chapter will discuss the most important analyses that were performed on the present state and some of the comparisons between the present, referred to as old, and the future state, referred to as the new state.

4.1 Improve Phase

As mentioned, Simulation Modeling is most useful in the Improve phase of the DMAIC process, where the current state and potential solutions are analyzed. Different tools can be used to develop these solutions, but they can be tested virtually in the simulation model in a more cost-effective method. Also, DOE to create optimal conditions can be performed, allowing poor solutions to be eliminated without incurring any real cost to the actual system. The analysis of the present state and validation based on the real system were performed based on the current state simulation model validated in the previous chapter. One could argue some of the analysis could be part of the Analysis phase as well as the Improve phase.
This model was modified for the Improvement phase of the DMAIC process by implementing the improvements suggested by the Lean Six Sigma Team who validated these improvements based on the simulation model. Under the improvement stage, the Opt quest algorithm was used to find the optimal number of workers required if the office started at 7 AM. The simulation model also decreased the time in system by identifying the optimal scheduling pattern by staggering times to go for lunches. The model was also used to manage the number of workers based on the volume on hourly, daily and monthly basis.

4.1.1 Looking at 7:00 AM Start

Figure 4.1 shows the influence on the mean total waiting time in system of different types of transactions based on the number of workers available. The simulation model was run for 30 replications for a typical Monday of a mid-volume month. As seen from the graph, each transaction type has a similar effect on the average time in system with the change in number of workers. Figure 4.2 shows the average time in system comparison by varying the number of workers for each different type of transaction. As one would expect, customers coming in for the road and computer test together take on average the most time while those getting state ID cards spend the least amount of time in the system.

Format and notations explanation:

1) Old system refers to the present state of the DMV office where no recommended change is done. New system refers to the simulation model where all the proposed what-if scenarios are deployed.

2) In several graphs and tables, the numbers 11, 10, 9, etc. show the number of workers.
3) In some of the graphs, on the x-axis it shows 7 to 8, 8 to 9 etc. which represents the hour of the day. For example, 7 to 8 represents time between 7 AM and 8 AM. Similarly, 8 to 9 represents time between 8 AM and 9 AM.

4) The Y axis shows numbers representing average total time in system.

Figure 4.1: Effect of the Number of Workers on the Average Time in System
Figure 4.2: Average Time in System for Each Entity Type Based on the Number of Workers Present

Figure 4.3 shows the comparison of the total average waiting time between the old and the new system. Here, from the graph it is evident that the waiting time almost approaches zero when there are 14 workers, which corresponds to 12 evaluators and a dedicated greeter and photo taker according to the new system which starts at 7:00 AM. The waiting time increased
substantially from an average of 17.44 minutes in case of 11 workers to 74.9 minutes in case of seven workers which usually happened due to workers taking leaves.

![Waiting Queue Time Waiting VS Number of Workers](image)

**Figure 4.3:** Comparison of Old and New System by Changing Number of Workers and its Effects on the Average Time Spent in Waiting Queue

Figure 4.4 shows the difference in number of balking customers between old and new systems with same number of workers on Monday of a mid-volume month. Here, as we can see in Figure 4.4, there is a peak in the time in system at the beginning of the office hours, so the average time in system substantially increases which may cause more people to balk, rather than in the old system where there is a constant time in system observed, which may cause the balking to decrease.
Number of Replications – 20

Day – Monday

Figure 4.4: Number of Balking Customer Comparison between Old and New Systems

Figure 4.5 shows the average comparison of the number of customers seen between the old and new system for a mid-volume month, where the day is Monday, and from 20 replications. Based on the improvements suggested by the Lean Six Sigma team, it was observed that in the new system more customers are seen compared to the old system, but there was not a substantial increase in the number. The old system represents the DMV office scenario before the improvements were performed where there was a random 20 to 40 customers lined up at 8 AM when the door opens. The system had 11 workers, but in the new system they are potentially going towards 14 workers.
Figure 4.5: Comparison of Number of Customers Seen between Old and New Systems

Figure 4.6 and Figure 4.7 show the overview of the comparison of the average time in system between the high volume and low volume months by day of the week for the new system opening up at 7:00 AM for 20 replications. The X-axis shows the days of the week and the Y-axis shows the time in system. If closely observing, one can see the volumes of Thursday and Friday are higher than on any other days, but as the volume increases in high volume months, there are more balking customers from the system which causes the overall time in system to decrease.
Figure 4.6: Average Time in System for Each Day of the Week of High and Low Volume Month

Figure 4.7: Comparison of Average Time in System between Low-Volume and High-Volume Month by Day
The following paragraphs present a lot of graphs and figures which use the fields named as shown in Figure 4.8 to shorten the names of those fields. In this figure, 14 shows that there are 14 workers, the numbers 7 7 8 show that the 12th worker starts at 7 AM, 13th worker starts at 7 AM and 14th worker starts at 8 AM.

Figure 4.8: Worker Notation Explanation

4.1.2 Optimizing the Starting Times for Additional Workers

Starting the Office at 7:00 AM is helping with the time in system, and the large number of people waiting in line when the office opened at 8:00 AM has been eliminated. This change included adding one to three more evaluators (i.e., 12, 13, and 14 workers) to the system. However, the starting times of the new workers needs to be determined. The original nine evaluators have a staggered start time (i.e., three start at 7:00 AM, three at 8:00 AM and then three at 9:00 AM) and the two dedicated for being a greeter and photo taker which represents
the original eleven workers as in case of old system. Analysis will be performed to look at the effect of adding one to three workers when their starting schedules are optimized.

4.1.3 Optimizing Three Additional Workers

Figure 4.9 shows the graph of average time in system with 14 workers showing varying start times for the 12\textsuperscript{th}, 13\textsuperscript{th} and 14\textsuperscript{th} worker where 14\_777 represents all three additional workers starting at 7:00 AM and leaving at 4 PM. The graph was plotted for a Monday of a mid-volume month. The Kim Nelson ranking selection method in SIMIO\textsuperscript{TM} was used to identify the best scenario based on the set objective to have the least average time in system with an indifference zone of 0.001 minutes. The KN algorithm identified that the scenario with the 14\_777 worker schedule was the best with least average time in system.
Further, to test this scenario in case of high-volume month, the model was simulated to identify the effect on time in system and to test if the scenario was also valid for a high-volume month. Figure 4.10 shows the effect of changing the schedule of the 12th, 13th and 14th worker on the hourly average time in system. Again, initially the volume increases during the opening hours of the office due to fewer workers coming in to start the office early, but then the time in system reduces by nine when the full capacity of workers is available. Again, the time in system starts increasing during the lunch hours when the capacity decreases and volume increases owing to the lunch breaks. The 14_777 was determined to be the best scenario in case of least average time in system of customers based
on the KN algorithm. However, the extra volume extends the longer waiting times starting at eleven as compared to the mid volume month. Figure 4.11 shows the SMORE plots for the average time in system for all the different scenarios for fourteen workers with the different starting times.

Figure 4.10: Average Time in System Based on Hour with Best Scenarios Selected using KN Algorithm for 14 Workers with Different Starting Times for a Monday of High-Volume Month
Table 4.1: Scenarios of Different Schedules for 14 Workers (Three Additional Workers) and its Effect on Time in System (Monday)

<table>
<thead>
<tr>
<th>Name</th>
<th>Schedule</th>
<th>Week</th>
<th>Week 2</th>
<th>Week 3</th>
<th>Week 4</th>
<th>Week 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>14_777</td>
<td>14_778</td>
<td>14_779</td>
<td>14_780</td>
<td>14_781</td>
<td>14_782</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>9</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>10</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>11</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>12</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>13</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>14</td>
<td>11</td>
<td>11</td>
<td>11</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>15</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>16</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
</tr>
<tr>
<td>17</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
</tr>
</tbody>
</table>

Figure 4.11: Scenarios of Different Schedules for 14 Workers (Three Additional Workers) and its Effect on Average Time in System (Monday)

Figure 4.12 shows the SMORE plots for the average number of balkers for the different scenarios with fourteen workers which shows that starting the additional workers as...
early possible to handle the initial rush of customers minimizes the number of bakers (i.e., 14_777, 14_778, etc.).

Figure 4.12: Scenarios of Different Schedules for 14 Workers (three additional workers) and its Effect on Number of Balking Customers (Monday)

Figure 4.13 shows the schedule for three temporary workers for a total of 14 workers and their effect on average time in system on Friday of a mid-volume month. The number 14_777 in the graph shows a scenario with 14 workers out of which the 12th worker starts at 7 AM, the 13th worker starts at 7 AM, and the 14th worker starts at 7 AM. These notations used are standard for every scenario. These scenarios were run using a KN algorithm with the most optimal number of replications decided at 95% confidence and 0.05 indifference zone to identify the best scenario, and it was identified that 14_777 was the best scenario with least time in system and least number of bakers.
Figure 4.13: Scenarios of Different Schedules for 14 Workers (three additional workers) and its Effect on Average Time in System (Friday)

Figure 4.14 shows the graph of the number of balkers on Friday for a total of 14 workers. In this graph various schedules were compared to identify the best scenario using the KN algorithm. The KN algorithm is used to identify the best scenarios based on the multiple replications and output objectives. It generates the number of runs based on the indifference zone defined. Based on the observations made from the graph, it is evident that the scenario is the best in terms of no. of balkers, as it is minimum, and considering the time in system which is the primary objective.
Figure 4.14: Scenarios of Different Schedules for 14 Workers (three additional workers) and its Effect on Number of Balkers (Friday)

4.1.4 Optimizing Two Addional Workers

Figure 4.15 and Figure 4.16 shows the box plot of average time in system with two additional workers by running the simulation for 20 replications of a mid-volume month for a Monday and Friday respectively. A similar scheme is employed where number 13_77 represents that there are 13 workers, out of which both the additional workers start at 7:00 AM. From both plots and as determined by the KN algorithm, having the two additional workers start at 7:00 AM is the best scenario owing to a high volume of customers entering the DMV office during the morning hours. On average, the total time in system was approximately 21 minutes and the number of balking customers is 0.5 for a Monday, while total time in system was approximately 29 minutes for a typical Friday. The best scenario chosen based on the KN algorithm was having 11 full time workers and two additional workers who start working at 7 AM. This is denoted by the notation 1377 in Figure 4.15: Scenarios of Different
Schedules for 13 Workers (two additional workers) and its Effect on Average Time in System (Monday)

Figure 4.15: Scenarios of Different Schedules for 13 Workers (two additional workers) and its Effect on Average Time in System (Monday)
4.1.5 Adding One Additional Worker

Figure 4.17 and Figure 4.18 show the scenarios which were tested in a similar fashion of adding one additional worker for a total of 12 workers. These experiments were run during different days and for both the days, if there were 12 workers, then the 12th worker should start the office at 8:00 am to have the least time in system for the customers or at 7:00 am for a high volume month.
Figure 4.17: Scenarios of Different Schedules for 12 Workers (one additional worker) and its Effect on Average Time in System
Figure 4.18: Scenarios of Different Schedules for 12 Workers (one additional worker) and its Effect on Time in System (Friday)

4.1.6 Comparing Best Starting Schedules for Different Number of Workers

Since the best starting schedules have been determined for adding one to three additional workers, the number of workers can now be compared directly. Figure 4.19 and Figure 4.21 shows the effect of different numbers of workers and their best schedules on the average time in system on a Monday of a mid-volume and high-volume month. Here, it can be observed that the confidence interval of the 13th and 14th workers are statistically not similar, but both are below the 25 minute goal. Moreover, even in the case of a high volume month there is no difference in time in system between 13 and 14 workers. Thus it is not economical to have 14 workers during the mid-volume and low-volume months. So, considering the economics of having one additional worker when there is no substantial effect on the average time in
system, one cannot recommend fourteen over thirteen workers. However, looking at Figure 4.20, the average time in system over the entire day shows that 13 and 14 are very similar except 14 does make a difference during the lunch hours.

Figure 4.19: KN Algorithm-based Best Scenarios of Different Worker Populations and their Effect on Average Time in System (Monday, Mid-Volume)

Figure 4.20: Time in System over the Time Customer Arrives
Figure 4.21: KN Algorithm-based Best Scenarios of Different Worker Populations and their Effect on Average Time in System (Monday, High-Volume)

Figure 4.22 shows the effect of different numbers of workers and their schedule on the overall average time in system on a Friday of a mid-volume month. Here, it can be observed that the confidence interval of the 13\textsuperscript{th} and 14\textsuperscript{th} workers overlap, and there is practical similarity in the time in system. Further, the time in system in case of 12 workers is also considerable from a customer standpoint. So, Friday being the highest volume day, it can be recommended that having 13 workers is economically beneficial on a Friday of a mid-volume month.
Figure 4.22: KN Algorithm-based Best Scenarios of Different Worker Populations and their Effect on Average Time in System (Friday, Mid-Volume)

Figure 4.23 shows the number of balking customers for all the best scenarios selected by the KN algorithm. Here, it can be seen that the number of balking customers in case of 14 and 13 are almost negligible and statistically significant. It is recommended that based on the statistical significance and economic decision, having 12 workers is better than having 14 or 13 workers if the primary objective is to minimize the number of balking customers with a minimized cost.
Figure 4.23: Number of Balking Customers for a Monday of Mid-Volume Month based on the Number of Workers present

Figure 4.24 shows the number of balking customers for all the best scenarios selected by the KN algorithm for a Friday of a high-volume month. Here, it can be seen that the number of balking customers in case of 14, 13 and 12 are statistically similar. It is recommended that based on the statistical significance and economic decision, having 11 or 12 workers is better than having 14 or 13 workers if the primary objective is to minimize the number of balking customers with a minimized staff. This decision is taken based on the assumption that each additional worker would have fixed cost each month. If there is no substantial effect on balking customers, even after increasing by one or two workers in a high volume month, then these workers would not be of any great advantage in mid-volume and low-volume months. So, it is better to have 11 workers if this is the primary objective.
Figure 4.24: Number of Balking Customers for a Friday of High-Volume Month based on the Number of Workers Present

Figure 4.18 shows the time in system based on the KN algorithm and its effect on the average time in system on a Friday of a mid-volume month. The graph shows 12_7 has the least time in system. Hence, with fewer workers and less time in system, a lower number of balking customers can be achieved which can be practiced on a Friday to identify the real effects on the system.
Figure 4.25: KN Algorithm-based Best Scenarios of Different Worker Populations and their Effect on Time in System (Friday, High -Volume)

Figure 4.26, Figure 4.27 and Figure 4.28 show statistics for number of customers served for Monday of a low-volume, mid-volume and high-volume month. The time in system has varied effects in all of the cases, such that numbers of customers seen are almost equal in cases of 14 and 13 workers for the low-volume month, while 14 workers serve more customers in mid-volume and high–volume months. The simulation model used 30 replications to determine the average number of customers.
Figure 4.26: Number of Customers Served on Monday of a Low-Volume Month with Different Numbers of Workers and Best Schedules Selected Based on KN Algorithm

Figure 4.27: Number of Customers Served on Monday of a Mid-Volume month with Different Numbers of Workers and Best Schedules Selected Based on KN Algorithm
Figure 4.28: Number of Customers Served on Monday of a High-Volume month with Different Numbers of Workers and Best Schedules Selected Based on KN Algorithm

Figure 4.29, Figure 4.30 and Figure 4.31 show the number of customers served statistics for Friday of a low-volume, mid-volume and high-volume month. The time in system has varied effects in all of the cases, such that numbers of customers seen are almost equal in case of 12, 13, and 14 workers for the low-volume month, while 14 workers serve more customers in a high-volume month. Since Friday is one of the busiest days of the week, it can be assumed that having 13 workers would fill the need for low-volume, mid-volume and high-volume months across the board.
Figure 4.29: Number of Customers Served on Friday of a Low-Volume Month with Different Numbers of Workers and Best Schedules Selected Based on KN Algorithm

Figure 4.30: Number of Customers Served on Friday of a Mid-Volume Month with Different Numbers of Workers and Best Schedules Selected Based on KN Algorithm
4.1.7 Effect of Adding Three Temporary Workers

The Six Sigma team was considering adding three part-time workers instead of adding three permanent workers owing to overheads, benefits, etc. Evaluators who have retired from the State of North Carolina are allowed to work part time (i.e., four hours a day) and not affect their retirement. As seen in the previous analysis, 14 workers are not necessary for low volume months and potentially for mid-volume months. In many cases 13 workers would work as well except during the lunch hours.

Figure 4.32 shows the effect of having one to three temporary workers who come in at 7:00 AM and work until 11:00 AM which is denoted by 14_7-11 which represents a total of 14 workers and 7-11 representing that three workers are temporary working from 7:00 AM to 11:00 AM. The simulation was run for 30 replications on a Monday of a mid-volume
month. The graph shows an effect of time in the system for different numbers of workers for each hour. One of the observations that can be made from the graph is that 14 and 13 workers have a statistical similarity (Overlapping confidence intervals) for most of the hours but are better than the current 10 and 11 workers. Owing to the lunch crowd, Figure 4.33 shows what happens if the temporary workers work from 10 AM to 2 PM. Again, 13 and 14 are similar, but the lunch hours average time in system is quite decreased as compared to the current system. However, the morning customers have to wait the exact time as in the old system.

**Figure 4.32: Effect of Temporary Workers Working from 7 AM to 11 AM on Average Time in System per hour**
Figure 4.33: Effect of Temporary Workers Working from 10 AM to 2 AM on Average Time in System per hour

Since there is a rush of customers at the beginning of the day as well as during lunch, a combination of part-time worker schedules will be employed and compared to the full time schedules.
Table 4.2 explains the combination of part-time schedules and shows the average time in system for all the different scenarios for 20 replications of Monday for a mid-volume month. Combinations of both 7 AM to 11 AM and 10 AM to 2 PM are very close to the scenario if one hired the same number of full time employees. This conclusion can also be seen in Figure 4.35 which shows the average time in system for customer arrival time period as those combinations seem to match very close to the full-time schedules. As was noted earlier, having 13 full-time workers was close to having 14, but actually having 13 full-time workers and one part-time to come in during lunch is actually better in terms of average time in system for all customers’ arrival hours and number of balking customers.

Table 4.2: Description of the Temporary Work Schedule Pattern
<table>
<thead>
<tr>
<th>Pattern</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>14_777</td>
<td>3 Full Time Workers (7 am to 4 pm)</td>
</tr>
<tr>
<td>14_77_10-2</td>
<td>2 Full Time Workers (7 am to 4 pm) and 1 Part Time Worker (10am -2pm)</td>
</tr>
<tr>
<td>14_7-11_10-2</td>
<td>3 Part Time Workers (2 work 7am – 11 am and 1 works 10am - 2pm)</td>
</tr>
<tr>
<td>14_7-11</td>
<td>3 Part Time Workers (7am – 11 pm)</td>
</tr>
<tr>
<td>14_10-2_7-11</td>
<td>3 Part Time Workers (1 works 7am – 11 am and 2 work 10am - 2pm)</td>
</tr>
<tr>
<td>14_10-2</td>
<td>3 Part Time Workers (10am -2pm)</td>
</tr>
<tr>
<td>13_77</td>
<td>2 Full Time Workers (7 am to 4 pm)</td>
</tr>
<tr>
<td>13_7_10-2</td>
<td>1 Full Time Worker (7 am to 4 pm) and 1 Part Time Worker (10 am - 2pm)</td>
</tr>
<tr>
<td>13_7-11</td>
<td>2 Part Time Workers (7am – 11 am)</td>
</tr>
<tr>
<td>13_10-2_7-11</td>
<td>1 Part Time Workers (1 works 7am – 11 am and 1 works 10am - 2pm)</td>
</tr>
<tr>
<td>13_10-2</td>
<td>2 Part Time Workers (10am -2pm)</td>
</tr>
<tr>
<td>12_7-11</td>
<td>1 Part Time Worker (7am – 11 am)</td>
</tr>
<tr>
<td>12_7</td>
<td>1 Full Time Worker (7 am to 4 pm)</td>
</tr>
<tr>
<td>12_10-2</td>
<td>1 Part Time Worker (10am -2pm)</td>
</tr>
<tr>
<td>11</td>
<td>0 Part Time Workers</td>
</tr>
<tr>
<td>10</td>
<td>0 Part Time Workers</td>
</tr>
</tbody>
</table>
Figure 4.34: Average Time in System for Different Full and Part Schedules
Figure 4.35: Comparing Different Part Time Schedules for One, Two and Three Temps

108
Figure 4.36 shows the SMORE plot of the number of balking customers for each of the scenarios with different numbers of workers. While the full time workers of 14_777 and 13_77 have the lowest average number of balking customers, working the part-time evaluators during lunch has the biggest impact on reducing the average number of balking customers. Having 13 full-time workers and 1 part-time employee who comes in during 10 am and 2 pm actually generates the lowest average number of balking customers, as there is always one additional person during at least one lunch hour. Finally, Figure 4.37 shows the average number of customers served with some of the part-time doing as well or better owing to cutting down on the number evaluators breaking for lunch and having a total complement of evaluators during the lunch period. Again the notion of 13 full-time and one part-time during the lunch for the Monday mid volume month is quite good on the average number of customers served by the system (i.e., the number of balking is decreased).

Figure 4.36: Effect of Temporary Workers Working from 7 AM to 11 AM on the Number of Balking Customers
Figure 4.37: Effect of Temporary Workers Working Full and Part Time on the Number of Customers Served

4.2 Control Phase

Under the control phase, Simulation modeling can be used to run the model for multiple days or multiple replications of the same day to identify the statistical effects of the Simulation model. The Simulation model can be used to forecast the responses based on the multiple numbers of runs. If an improvement is suggested, the Simulation model can be used to validate and test the level of improvement that can be achieved over a period of time. In places where a control phase needs to be performed, which may take a substantial amount of time, then virtual validation can help in estimating the results. The FMEA can be performed using simulation modeling to identify the failure risk mitigation effect and its validation. In case of control charts, Simulation can give a graphical representation of a process or product meeting the specification and control limits. It can also be used to forecast the level of control that can be achieved. One of the important parts of the Control phase is
documentation, which is achieved in SIMIO™ by exporting the reports readily from its interface to any external interface. Simulation can also be used in some of the frequently used tools during the control phase like testing and inspection, auditing, etc. Simulation modeling can also be used to update the data on a regular basis which can be helpful in case of continuous improvements and improvements revision. In the present case, it was difficult to validate the system in all the offices of the DMV, so Simulation modeling has been a really helpful tool in validating and controlling the improvements suggested in a virtual environment to test the hypothesis.

Further, the optimal values which were obtained by the OptQuest™ and KN algorithm were tested and audited to be true for different scenarios and multiple runs in the simulation model.
Chapter 5
Conclusions and Future Work

This study has provided a platform for using a simulation in a DMAIC process in a Six Sigma project. One of the important features of this case study has shown that apart from Simulation itself being a method, it can also be used as a tool in driving various management and process optimization projects. The main purpose of simulation modeling is to save time, money and efforts in executing real life improvements.

This case has explained how simulation modeling is useful in DMAIC processes. There are several scenarios where practical execution of the various DMAIC tools and processes is not feasible due to time and financial barriers, red tape, etc. This has led to failures of most of the DMAIC projects. This thesis has shown how important simulation has been in solving real world problems using a virtual computing environment for optimization and how Lean philosophies and Six Sigma methods can be combined with simulation modeling to attain required objectives.

5.1 Summary of Conclusions

This thesis shows how Simulation modeling can be used in the Measure phase of the DMAIC project when the given data is insufficient or inaccurate. It shows that even before creating a Data collection plan, Simulation can be used to set the data standards and resolutions. It can also be used to identify necessary sensitive data and overlook the non-value added data points.
Further, Simulation is a very essential part of the Analyze, phase as lot of evident effects of the present system’s performance are effectively gauged and graphically interpreted to analyze the bottlenecks and the areas needing attention. The analyze phase is followed by the improvement phase, in which all the improvements are tested prior to real life execution to validate the proof of concept. The simulation model also gave the optimal values of the required objectives by running special algorithms in a virtual environment.

Having 13 full-time workers and one temporary worker working from 10 AM to 2 PM gives a better result in all of the scenarios of high volume, low volume and mid-volume months. It is not statistically the best, but considering practical significance, it is the best scenario considering the very little difference in time in system between this scenario and the best scenario of having 14 workers with two workers starting at 7 AM and one working from 10 AM to 2 PM. The cost of having additional workers which do not make a substantial difference on the response variables is unnecessary.

5.2 Limitations and Recommendations

Based on the data analysis from the simulation model and the propositions by the Lean Six Sigma teams, it has been observed that there is statistical similarity in the time in system of the customers when the numbers of workers present were 13 and 14. In this case, it is economically not effective to have a 14th worker in any of the office scenarios in any of the months. Owing to this observation and considering the main objective of time in system, it is better and recommended to have 13 workers in the DMV office.

Further, there is a time in which the photo person would be mostly idle during the beginning of the day when there is huge volume and most of the people have just started
coming in the DMV office. At this time, the photo person can help the greeter in checking the customers into the office which would substantially help the morning peak in time in system. Increasing the capacity of check-in would adversely affect the time in system as the model and hence the DMV office is very sensitive to the check-in time.

One more important recommendation is to have temporary workers during the morning hours which can help part time and reduce the time in system during the peak hours. So, when 13 workers are recommended, 11 are permanent and 2 are temp workers who only work during peak hours and high-volume months.

5.3 Future Work

For future work, the simulation model can be made to automatically import the real time data from the DMV online time mapping system and do real time validation and analysis to address and forecast volumes and their solution can be presumed. Further, the data needs to be collected with fine resolution and in terms of simulation input format. Moreover, there is need to collect the monthly volumes and other data to validate the model based on real data and not mere assumptions. The data about worker behavior and worker breaks was assumed in this system and needs to be measured again. The simulation model is also not very sensitive to the weather and other conditions that may affect the arrival process which should be considered. The arrival pattern is designed to address the problem of arrival changing by hour, by day and by month, but it has a statistical limitation which needs to be corrected. The processing time data needs to be collected again to capture double processing times at the same workstations which was not considered but assumed based on pilot readings in the present model. The phone calls attended by the DMV office workers need to
be considered. In case of full-time and part-time workers, the work schedule needs to be optimized using OptQuest™ to find the best possible schedule.
References


Chramcov, B., & Bucki, R. (2014). Lean manufacturing system design based on computer simulation : Case Study for Manufacturing of Automotive Engine Control Units."


Goforth, Kelly “Adapting Lean Manufacturing principles to Textile Industry.”(2007)


Luce, K., Trepanier, L., Ciochettoc, F., & Goldman, L. (n.d.). "Simulation and Optimization as Effective DFSS Tools."


116


Appendices
Appendix A

Figure 38, Figure 39 and Figure 40 show the animated Simulation model for Greensboro, Durham and Charlotte office respectively. These three were the target offices for the DMV office.

Figure 38: Greensboro Office Model
Figure 39: Durham Office Model

Figure 40: Charlotte Office Model