

7-2011

# Using simulation techniques to improve the client flow of the Kentucky Cabinet for Health and Family Services.

Russell Edward Harpring  
*University of Louisville*

Follow this and additional works at: <https://ir.library.louisville.edu/etd>

---

## Recommended Citation

Harpring, Russell Edward, "Using simulation techniques to improve the client flow of the Kentucky Cabinet for Health and Family Services." (2011). *Electronic Theses and Dissertations*. Paper 577.  
<https://doi.org/10.18297/etd/577>

This Master's Thesis is brought to you for free and open access by ThinkIR: The University of Louisville's Institutional Repository. It has been accepted for inclusion in Electronic Theses and Dissertations by an authorized administrator of ThinkIR: The University of Louisville's Institutional Repository. This title appears here courtesy of the author, who has retained all other copyrights. For more information, please contact [thinkir@louisville.edu](mailto:thinkir@louisville.edu).

USING SIMULATION TECHNIQUES TO IMPROVE  
THE CLIENT FLOW OF THE KENTUCKY CABINET  
FOR HEALTH AND FAMILY SERVICES

By

Russell Edward Harpring  
B.S., University of Louisville, 2010

A Thesis  
Submitted to the Faculty of the  
University of Louisville  
J. B. Speed School of Engineering  
as Partial Fulfillment of the Requirements  
for the Professional Degree

MASTER OF ENGINEERING

Department of Industrial Engineering

July 2011



USING SIMULATION TECHNIQUES TO IMPROVE  
THE CLIENT FLOW OF THE KENTUCKY CABINET  
FOR HEALTH AND FAMILY SERVICES

Submitted by: \_\_\_\_\_  
Russell Edward Harpring

A Thesis Approved on:

\_\_\_\_\_  
(Date)

by the Following Reading and Examination Committee:

\_\_\_\_\_  
Gerald W. Evans, Thesis Director

\_\_\_\_\_  
John S. Usher

\_\_\_\_\_  
Stacy M. Deck

## ACKNOWLEDGEMENTS

I would first and foremost like to thank Dr. Gerald Evans, my thesis advisor, for his guidance, support, and patience throughout this project. Of course, this project would not be possible without the generous help of Dr. Gerard Barber, and Dr. Stacy Deck, who gave me the opportunity to work on such a unique and interesting project.

I would also like to send thanks and sincere appreciation to Ms. Vickie Blevins and Ms. Becky Murphey for their direction and on-going support at the Kentucky Cabinet for Health and Family Services. In addition, I'd like to thank Ms. Janet Washington, Ms. Lona West, Ms. Jackie Stamps, Ms. Brenda Lauderdale, and the employees on the 9<sup>th</sup> Floor, who all provided advice and help along the way.

I thank Dr. Suraj Alexander, Dr. John Usher, Ms. Dorothy Goya, Ms. Dru Kemp, and Mr. Todd Trapp for their help with this project, the capstone project, and the promotion of both.

Special thanks to my family, friends, and professors who supported and encouraged me over the past year, it helped me keep going.

## ABSTRACT

The client flow of the Kentucky Cabinet for Health and Family Services is a complex process that deals with a high volume of clients on any given beginning half of the month. A simulation model of an average busy day in the system was built using the Arena Simulation package. Methods for creating a simulation using Arena are included in this work. Statistics were generated from a number of different sources to create an accurate representation of the model.

The simulation model shows long wait times for clients coming into the system to receive benefits. Working with a state budget during variable economic times leaves no room for hiring new workers, so changes must come from within. The flexibility of the simulation model allows for experiments on the system to be created and observed. Experiments included in this work show changes to the main lobby by adding kiosk systems and an alternative method for processing clients. Both experiments show beneficial outcomes, with reduced wait times and better efficiency. The model can also be used for preventive measures. An experiment showing a 10% influx of clients in the system demonstrates the strain the system would undergo should such a scenario occur. Using the simulation model to test changes to the system can help guide decision making by providing valuable output and reliable statistics.

## TABLE OF CONTENTS

APPROVAL PAGE	i
ACKNOWLEDGEMENTS	ii
ABSTRACT	iii
LIST OF TABLES	vi
LIST OF FIGURES	viii
I. INTRODUCTION	1
II. REVIEW OF LITERATURE	5
III. OVERVIEW OF SIMULATION SOFTWARE	11
IV. MODELING THE CHFS SYSTEM	20
A. Problem Formulation	20
B. Understanding the System	21
C. Establishing Clear Goals	25
D. Formulating the Model Representation	26
E. Collecting Accurate Data	28
F. Translating into Arena	31
G. Verification of the Model	50
H. Validation of the Model	51
I. Designing Experiments	52
V. RESULTS AND DISCUSSION OF RESULTS	58
A. Present Model	58
B. Sign In Kiosk Experiment	62
C. 10% Client Influx Experiment	64

D. 4 <sup>th</sup> Floor Alternative Case Processing	66
Experiment	
VI. CONCLUSION	69
VII. RECOMMENDATIONS	70
REFERENCES	71
APPENDIX I	74
APPENDIX II	76
APPENDIX III	78
APPENDIX IV	80
APPENDIX V	83
APPENDIX VI	88
VITA	90



## LIST OF TABLES

2.1	SEVEN PHASES OF LEAN IMPLEMENTATION BY THE LOUISIANA HEALTH AND SERVICES DEPARTMENT	7
4.1	CATEGORIZATION OF CLIENTS FLOWING THROUGH SYSTEM	27
4.2	PROBABILITIES FOR DECIDE MODULES IN EACH SUB-MODEL	45
5.1	PRESENT MODEL SIMULATION OUTPUT RESULTS	59
5.2	KIOSK EXPERIMENT VS. PRESENT MODEL SIMULATION OUTPUT RESULTS	63
5.3	10% CLIENT INFLUX EXPERIMENT VS. PRESENT MODEL SIMULATION OUTPUT RESULTS	65
5.4	4 <sup>th</sup> FLOOR ALTERNATIVE METHOD EXPERIMENT VS. PRESENT MODEL SIMULATION OUTPUT RESULTS	66
A.1	CLIENT ENTITY ARRIVAL SCHEDULES FOR SIMULATION MODEL	77
A.2	3F SCHEDULED PROCESSING TIMES EXPRESSIONS	81
A.3	3F UNSCHEDULED PROCESSING TIMES EXPRESSIONS	81
A.4	4F SCHEDULED PROCESSING TIMES EXPRESSIONS ORIGINAL MODEL	81
A.5	4F UNSCHEDULED PROCESSING TIMES EXPRESSIONS ORIGINAL MODEL	81
A.6	4F SCHEDULED PROCESSING TIMES EXPRESSIONS EXPERIMENT MODEL	82

A.7	4F UNSCHEDULED PROCESSING TIMES EXPRESSIONS EXPERIMENT MODEL	82
A.8	OBSERVED CLIENT DATA SIMULATION OUTPUT FOR ORIGINAL MODEL	84
A.9	OBSERVED SYSTEM DATA SIMULATION OUTPUT FOR ORIGINAL MODEL	84
A.10	OBSERVED CLIENT DATA SIMULATION OUTPUT FOR KIOSK EXPERIMENT MODEL	85
A.11	OBSERVED SYSTEM DATA SIMULATION OUTPUT FOR KIOSK EXPERIMENT MODEL	85
A.12	OBSERVED CLIENT DATA SIMULATION OUTPUT FOR CLIENT INFLUX EXPERIMENT MODEL	86
A.13	OBSERVED SYSTEM DATA SIMULATION OUTPUT FOR CLIENT INFLUX EXPERIMENT MODEL	86
A.14	OBSERVED CLIENT DATA SIMULATION OUTPUT FOR 4F PROCESSING CHANGE EXPERIMENT MODEL	87
A.15	OBSERVED SYSTEM DATA SIMULATION OUTPUT FOR 4F PROCESSING CHANGE EXPERIMENT MODEL	87

## LIST OF FIGURES

3.1	The Create Module and Accompanying Dialog Box	14
3.2	The Process Module and Accompanying Dialog Box	15
3.3	The Decide Module and Accompanying Dialog Box	16
3.4	The Separate Module and Accompanying Dialog Box	17
3.5	The Assign Module and Accompanying Dialog Box	17
3.6	The Record Module and Accompanying Dialog Box	18
3.7	The Dispose Module and Accompanying Dialog Box	18
3.8	Data Module List and Editing Box	19
4.1	High Level Value Stream Map of the Client Flow Process	22
4.2	Full View of the Arena Simulation Daily Model	31
4.3	The Run Setup Dialog Box with Parameters for the Client Flow Model	33
4.4	The Create Modules for Intakes, Walk-Ins, and Scheduled Clients with Dialog Box for Intakes	34
4.5	The Assign Modules for Unscheduled and Scheduled Clients with Dialog Box for Unscheduled Clients	35
4.6	The Process Module for the Scheduled Clients' Timeliness Delay	36

4.7	The Process Module for the Main Lobby Receptionist and Accompanying Dialog Box	37
4.8	The Process Module for the 3F Receptionist and Accompanying Dialog Box	39
4.9	The 3 <sup>rd</sup> Floor Decide Module to Determine if Entity is Scheduled	40
4.10	Series of Decide Modules on 3 <sup>rd</sup> Floor based on Entity Type	41
4.11	Sequence of Steps for the Assigning of the Entity Subtype 1	42
4.12	Overview of Sub-Model for Scheduled Entities on the 3 <sup>rd</sup> Floor with Recertification Meetings	44
4.13	Seize Module for a Scheduled SNAP Recertification Client on the 3 <sup>rd</sup> Floor	47
4.14	Delay Module for a Scheduled SNAP Recertification Client on the 3 <sup>rd</sup> Floor	48
4.15	Process Module with Delay-Release Logic for a Duplicated Entity	49
4.16	Experiment with Additional Kiosk Station in Main Lobby to Reduce Traffic	53
4.17	Sub-Model with Changed 4F Process Using Interviewers and Processors	56
5.1	Chart Tracking WIP Value over Time in the Present Model	61
A.1	Enlarged Full View of Simulation Model Image	75
A.2	Enlarged Sub-Model View for Processing	79
A.3	Enlarged WIP Chart	89

## I. INTRODUCTION

The Kentucky Cabinet for Health and Family Services (CHFS) provides government funding to those in need of temporary financial support. In Jefferson County alone, there are over 109,000 active cases in the system. With so many active cases, workers and resources need to be utilized in a way that makes the process run as efficiently and effectively as possible.

The main CHFS branch for Jefferson County is located in the old L&N Building at 908 West Broadway in Louisville, KY and the Family Support Division encompasses the 3rd and 4th floors of the building. The CHFS provides a number of services and programs in the building such as the Supplemental Nutritional Assistance Program (SNAP), the Kentucky Transitional Assistance Program (KTAP), the Medicaid program, Kinship care, and the Kentucky Works Program (KWP). Each program has its own unique function and set of requirements for participants. Participants of SNAP will receive food related benefits in the form of a transaction card. The Medicaid program is designed to provide health insurance to those who cannot afford it. Participants of KTAP may receive financial support in junction with other programs until the participant finds a job or generates enough income to be ineligible for further benefits. KWP is an extension of KTAP, in which participants can learn a trade, earn a GED, or take classes while receiving benefits. These programs aim to provide assistance to needy families while

encouraging them to become self-sufficient. With an increasing number of clients in the system (about 9% from January 2010 to January 2011), case workers and case managers are trying to get their clients back on their feet as soon as possible. However, with the implementation of six furlough days, over the course of a year, beginning in July 2010 [1] and recent restriction of overtime, employees are feeling pressure to work even harder.

For this purpose, a study has been conducted to create a valid and accurate simulation model of the client flow process, which can then be used to determine the most effective methods to govern the system. However, with such a complex system in place, much data was needed to create a simulation model.

The early workings of the project began in the spring of 2010. A study was completed at the CHFS, as a part of the required Capstone Design project, to understand where the bottlenecks in the system lie and offer suggestions for improvement. The Capstone Design class is required for seniors and tests their knowledge by allowing students to complete a real world project using skills acquired through their engineering courses. For that project, most of the time was spent collecting data and the team found out that the average client wait time is around 90 minutes, with the actual time in the system being a little over 120 minutes. The team did not get to implement any ideas they had for improvement, but did get to give a presentation exclusively to the management at the CHFS. One of the suggestions was to create a detailed simulation model that could encompass multiple factors present in the system. Once the capstone was completed, an opportunity was presented to continue the study, with a narrowed focus on simulating the system.

This project coincided with already existing efforts by the Kent School of Social Work at the University of Louisville to research caseload trends over time and to restructure the Kentucky Works Program, with the goal of increasing client participation rates within the program. The team believes the work and research done in the area of simulation can be tied into their project and will be useful for future studies.

The decision was made to create a simulation model that would replicate one “busy day” in the system. A “busy day” is typically defined as a work day in the first half of any given month. Case workers are encouraged to only schedule client appointments in the first fifteen days of the month, leaving the rest for paperwork processing. The model is loosely based on the model used for the capstone project, but refined. The data was collected from real clients, and categorized based how clients enter the system (scheduled vs. unscheduled), what types of services they are applying for, and whether or not this is their first time in the system. The amount of time a case worker spends with a client depends on the services they are applying for.

By finding patterns in the system, strategies for more effective services can be identified. With the model, real time data can be used to determine how the process can be improved, whether it is adjusting staff between floors, a better layout, reduction in paperwork, etc. The overall goal of this project is to be able to use the simulation model to maximize available resources, reduce client wait time, and identify any additional resources that may be needed to aid the system.

In this work, the process flow of the CHFS will be defined and examined by means of value stream mapping. The simulation model was derived from the process map, with each step, module, and variable explained in detail. Various iterations were created for

the simulation model, to demonstrate its versatility. These iterations include: 1) the effects of a 10% influx of clients into the system, 2) a new work procedure for handling clients, and 3) the effects of a sign in kiosk system.



## II. REVIEW OF LITERATURE

As popular as process improvement is in manufacturing, healthcare, and other service industries, it has not yet taken a strong hold in various departments of government. The United States military adopted lean techniques and process improvement in some areas, such as logistics and supply chain, but lacks in other areas [2]. Some argue that branches of government oftentimes have a hard time defining who the customer is, and what their exact needs are, since so many services can be offered by a single branch [2]. As opposed to private sectors, many government agencies' most important concern is to serve everyone equally. Therefore, there are two types of customers that can be identified in government agencies: the client who utilizes services and the general public (general taxpayers). Performance measures have been put in place to survey the perceived performance from the standpoint of the two customers. Unfortunately, because the general public is the ultimate source of money, they are oftentimes viewed as the most important customer [3]. This can cause complications, as some government services will focus more on satisfying public demand and political agendas rather than measurable process outputs [2]. However, most proponents of Lean techniques in government do not call for tax cuts, but more efficient use of tax dollars, better understanding of the system by state and federal representatives, and a continuous improvement mindset.

One method of overcoming these shortfalls is a push to implement process improvement measures into these government services. However, each branch of service should customize the methods to meet the needs; combining techniques from Total Quality Management (TQM), Lean, and Six Sigma will give each branch tools and measures to improve performance [2, 4]. For example, in 2003 the Florida Department of Revenue initiated a program named, “Six Sigma Light.” The “light” refers to customization of the Six Sigma methodologies and its slow implementation over time, so workers would not feel the heavy burden of change all at once [4]. The Florida Department of Revenue created internal benchmarking since the process began, and has slowly changed the culture of the organization to be “result-minded”, with the bottom line being time saved [4].

As each government service is unique, what works for one program does not always work for another. Recent research was conducted by the IBM Center for Business of Government, which conducted a case study looking at the way Louisiana used Lean principles to reshape its welfare system after Hurricane Katrina hit New Orleans. After the destruction from the hurricane, many workers had relocated or failed to show up again, files were destroyed, and information was lost [5]. The Louisiana officials knew it was important to act quickly, as there would be a large increase in clients due to the disaster. With the ability to recreate the program, leaders decided to implement Lean into the organization through six different actions over seven different phases. The six actions are as follows: 1) view work as a process, 2) redesign the flow of work, 3) redesign the distribution of work, 4) differentiate between internal and external customers, 5) analyze data to improve flow and customer service, and 6) use measurement to communicate and

to give feedback [5]. The seven phases are listed below in Table 2.1, and depict the timeline and actions taken during each phase.

TABLE 2.1

SEVEN PHASES OF LEAN IMPLEMENTATION BY THE LOUISIANA HEALTH AND SERVICES DEPARTMENT

<b>Phase Number</b>	<b>Time Period</b>	<b>Description of Action</b>
Phase One	Feb. – Mar. 2006	Formulate teams, create process flowchart, identify client population for testing changes
Phase Two	May 2006	Develop clear goals and strategies, definable measurements, create program materials
Phase Three	May – Aug. 2006	Accountability between various branches, conference calls reporting results
Phase Four	August 2006	Learn about process monitoring, root cause analysis, and visual management techniques
Phase Five	Aug. 2006 – Feb. 2007	Testing changes, implementing improvements, monthly conference calls to report findings
Phases Six	Feb. 2007	Managing performance, increase productivity, standardizing work, knowledge sharing and transfer, review of policies
Phase Seven	Feb. 2007 - Present	Creation of WorkSmart! program, defined infrastructure, continuous review of methods, continuous reporting of metrics

The implementation was successful and processing times were reduced significantly [5]. Other methods have been proven successful in recent studies, such as planning and operations research [6, 7], value stream mapping [8, 9], mathematical programming [10, 11], and reviewing organizational structure to maximize performance [7, 11-13]. While the methodologies are different for each service and circumstance, the goal remains the same for each organization that has adopted process improvement techniques, to improve client satisfaction. This differs from traditional government service ideologies that look to improve satisfaction from the general public [2, 12]. Oftentimes, government agencies

react and change policies based on general public consensus, and not the client feedback. However, by first understanding the system and the needs of the client, processes can be improved in all service providing industries [6, 12, 13].

One method of process improvement that was not mentioned above is using simulation to analyze a system. The term “simulation” has many definitions, but for the sake of this work, it can be defined as, “...methods and applications to mimic the behavior of real systems, usually on a computer with appropriate software,” [14]. The software package used for this project is Arena<sup>®</sup> Software, which is a simulator that has great modeling flexibility due to being fully hierarchical. That is to say, there is a single graphical interface that allows different kinds of programming [14]. The Arena<sup>®</sup> Software package and applications will be explained in more detail in the Instrumentation Section.

Although Lean and Six Sigma have an abundance of tools used to analyze processes, simulation is oftentimes the best option when looking at a process over time. Spreadsheets and computer programs are great ways to keep track of expenses, schedules, and other statistics, but as the problem grows in complexity, so does the need for accurate algorithms and measures [15]. Simulation takes care of this by including such factors as variability and interdependent operations. While most spreadsheets and programs rely on averages for computations, simulation uses distribution functions to represent the randomness present in most processes [15]. It is important to note that running a simulation is not the only tool that should be used during an analysis. Data must be collected first and put into the model, but a simulation model alone cannot show how to optimize a process [16].

Creating a simulation of a process can be beneficial in many different ways. By creating a simulation of a system in detail, the user is forced to analyze the system in depth and really understand how the system works [14]. Apart from that, simulations offer the ability to measure system performance, improve operations, and see the effects on various designs within the system. However, the most important aspect that simulation has to offer is its ability to create experiments based on actual systems [17]. Many times, it is infeasible (and usually not cost-effective) to alter a system physically based on an experimental idea without doing any sort of prior calculations. With a simulation model, an experiment can be created and run with no harm done to the actual system [17].

While simulation has not been utilized very much in the social services, it has been used extensively in healthcare settings, which operates similarly to social services. Over the past couple decades the healthcare sector has been under pressure to increase quality in all operations while reducing costs [18]. Healthcare models and social services models share many similarities in that they're usually large and complex, have a high volume of clients with different needs, and have unique constraints in the form of federal or state laws. One issue that is often hard to model is the needs for each client coming into the system. If a hospital were being modeled, clients would enter the system for a number of different reasons, and could take multiple paths based on their classification [19]. The same applies for the social services setting [20], where clients will inherently spend different amounts of time in the systems based on their needs and what services they are seeking [21]. To tackle this problem, many models are built based around a specific service within a system to reduce the model complexity [22]. For example, by focusing

solely on the emergency department [18], a cancer treatment facility [22], or outpatient surgery, the scope of the project becomes much smaller. This will generally help the user create a more accurate and simplified model [19]. Healthcare providers use simulation models to analyze current performance and compare alternatives to find how to operate more efficiently and save money.

Simulation is not without fault though, and it is important to understand and know the shortcomings before using it as a tool for analysis. Earlier it was mentioned that a benefit to simulation is the ability to incorporate variability into the system, which can be helpful because of the realistic aspect. However, a stochastic model will always generate random output because the input variables are based on probabilities. The randomness and variability can be reduced in a simulation model through a number of different ways, such as running many replications of the same model and averaging them together [14]. Many sources point out the danger in creating a simulation model and using the output data as metrics before the model itself was ever validated [14-17]. For this project, getting accurate data representative of the system was very important for model validation.

### III. OVERVIEW OF SIMULATION SOFTWARE

As mentioned in the previous section, the Arena<sup>®</sup> software package, created by Rockwell Automation, was used as a primary tool in this study. Arena<sup>®</sup> uses SIMAN programming language at the lowest level of modeling to execute and run scripts, which allows it to be very flexible. However, the modeling done in this study was done on a higher level, using basic and advanced process modules to link the steps in the system together. This section of the paper will explain key concepts and terms found in the Arena<sup>®</sup> software that will be referenced through the remainder of this work.

Arena<sup>®</sup> uses various parts and objects to replicate a system such as entities, attributes, variables, resources, and queues. Entities are the dynamic objects in the system that are created at the beginning of the model, flow through the system, and are terminated at the end. These entities represent clients in the system in this work, but can take on the role of anything being processed by a system. Each entity must be created to be brought into the system, either by a user defined schedule or by a defined distribution function [14].

Attributes are characteristics of entities that get attached to entities as they flow throughout the system. Attributes offer a way to characterize entities, as an attribute value is tied to specific entities. An entity can have multiple attributes attached to it, and

the specific attributes may have different values for each entity [14], just as each client coming into the system will have different needs and will be requesting different services.

Variables differ from attributes in that they affect the characteristics of a system, rather than specific entities, which is why they're often called global variables [14]. In this software, there are two types of variables: user-defined variables (work in progress, average service time, travel time, etc.) and Arena<sup>®</sup> built-in variables (number in queue, current number busy, current time in simulation, etc.) [14]. Variables are important because they can act as a trigger to change a value in the model over time. For example, a variable to track work in progress can be made so that every time a client enters the system, the value increases by 1, and respectively decreases by 1 when a client exits. Since variables affect the entire system, they are very useful tools for collecting user-defined statistics and metrics.

As an entity flows through the system, it is seized by various resources at process stations. Resources can represent any function that processes an item in a system with a service time [14]. In this project, resources represent the various personnel in the system that work with the clients to get their needs fulfilled. When an entity is seized by a resource, it is held there for a specified amount of time, and then released when the time expires, and the entity can continue to flow further downstream in the process. A single resource can only process one entity at a time, giving the user the option to customize the simulation model to have features such as a scenario with multiple resources in a single module to service more than one entity at a time at a single station. Resources can be used in junction with variables to make changes as the model runs over time. For



instance, a variable can be used to decrease the number of resources available during a lunch hour in the system.

Sometimes, as an entity flows through the system it will not have an available resource to seize it, so it must wait in a queue. Queues can have different features to govern how entities wait in line, such as first-in first-out (FIFO), last-in first-out (LIFO), or can even be based on attribute values. Queues can be a very useful tool to represent physical aspects of the model, such as waiting rooms.

Combining the various pieces of a model will dictate how the model runs and what the output will be, but it is also important to understand how the model works. A model works by having events run at specified times throughout the simulation. An event is any simulated occurrence that might change variables, attributes, or other statistical accumulators [14]. Events happen in order of logic associated with the various parts of the model. In the model created for this project, there are several steps in the process with many events happening simultaneously.

As the model runs, an internal simulation clock keeps track of current time within the system. The simulation clock interacts with the events in the model and flows while each event is executed [14]. The simulation clock is important to the model for many reasons. First, it keeps track of how long a process takes to run, giving valuable recordable output statistics. Second, it can be used to begin and end a simulation at certain times to represent an average working day. Lastly, it can be used for variables as an indicator to perform a specific function at a specific time, like a scheduled lunch break at noon.

To piece together the model, Arena<sup>®</sup> uses flowchart objects called modules to build a model. The important flowchart modules used in this work are as follows: Create, Process, Decide, Separate, Assign, Record, and Dispose. These modules can be connected together to form a flowchart-like process to create a path for the entities. There are other modules that can control advanced features of the model, such as animation. Each module created will have a unique name and characteristics to go along with it, and may affect objects within the model itself, such as variables, attributes, resources, etc.

At the beginning of a flowchart is usually a Create module, which brings entities into the system at specified times. Each Create module can be customized to specify what types of entities are arriving in the system, when they arrive in the system, how many arrive, and what type of distribution accompanies the time between arrivals. This information can be edited in the dialog box, shown below in Figure 3.1.

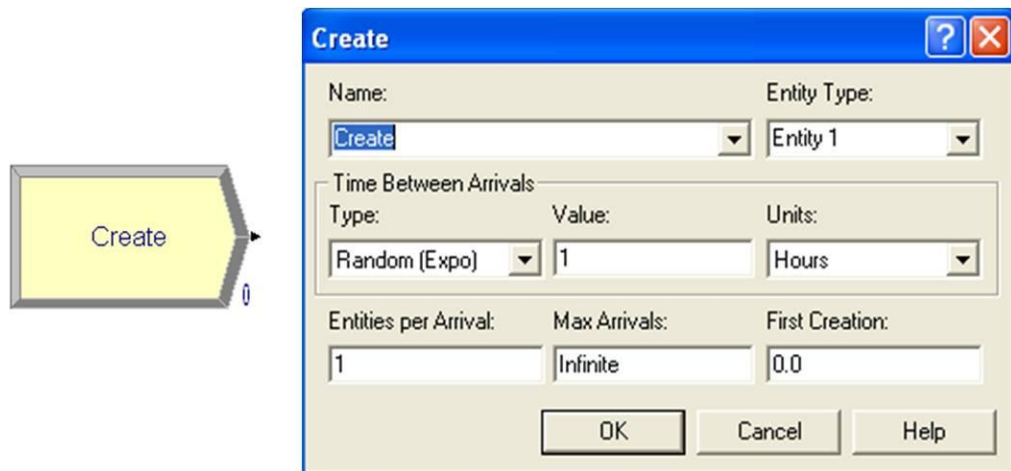


FIGURE 3.1 – The Create Module and Accompanying Dialog Box

A Process module represents a station in the system that an entity can enter and be serviced. Most Process modules have their own resources, queues, and processing time. Many of the Process modules used in this study follow the “Seize Delay Release” logic,

which means that an entity flowing through the Process module will be seized by a resource, delayed for a specified amount of time, and then released to continue its flow downstream. The delay type can be based on a user specified distribution or chosen from the drop down box with key values filled in. The module and dialog box are shown below in Figure 3.2.

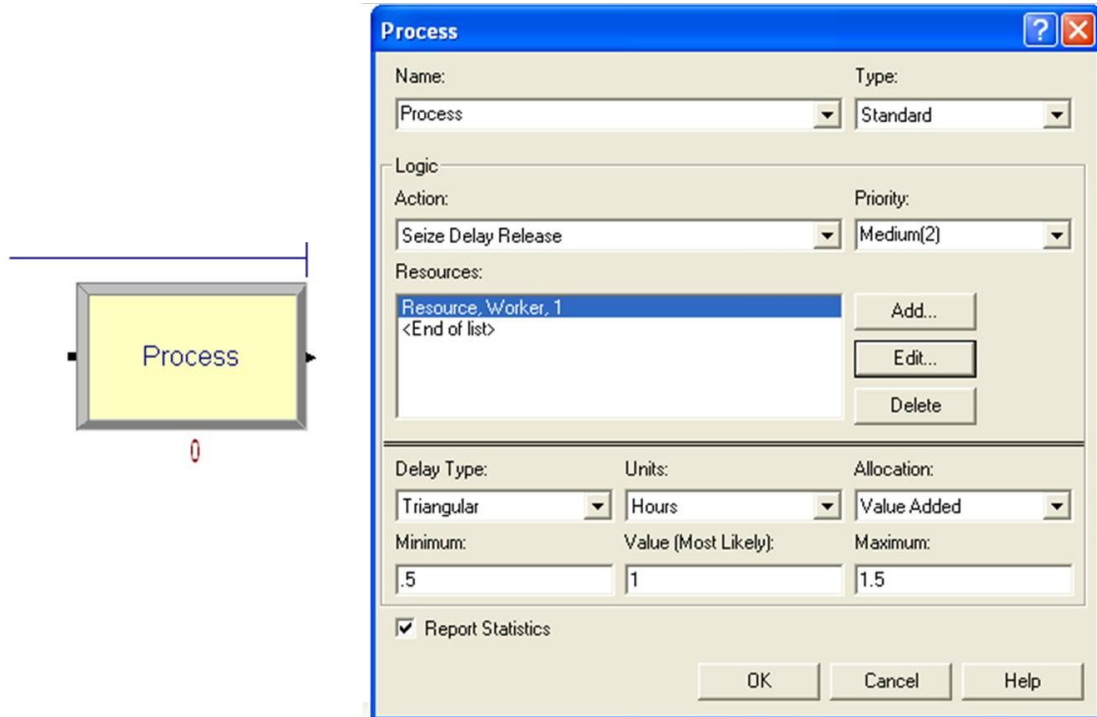


FIGURE 3.2 – The Process Module and Accompanying Dialog Box

The Decide module is a basic module which splits the flowchart into various paths based on a condition or chance. The user can choose N-number of paths to split with the Decide module. This feature becomes particularly useful when entities with certain attributes need to be routed in a certain direction based on condition. When splitting based on chance, the user will put various percentages into the dialog box, with each path leading to its respective step. The total does not need to add to 100%, as the rest of the

entities will be routed in the “Else” pathway. The Decide module and dialog box are shown below in Figure 3.3.

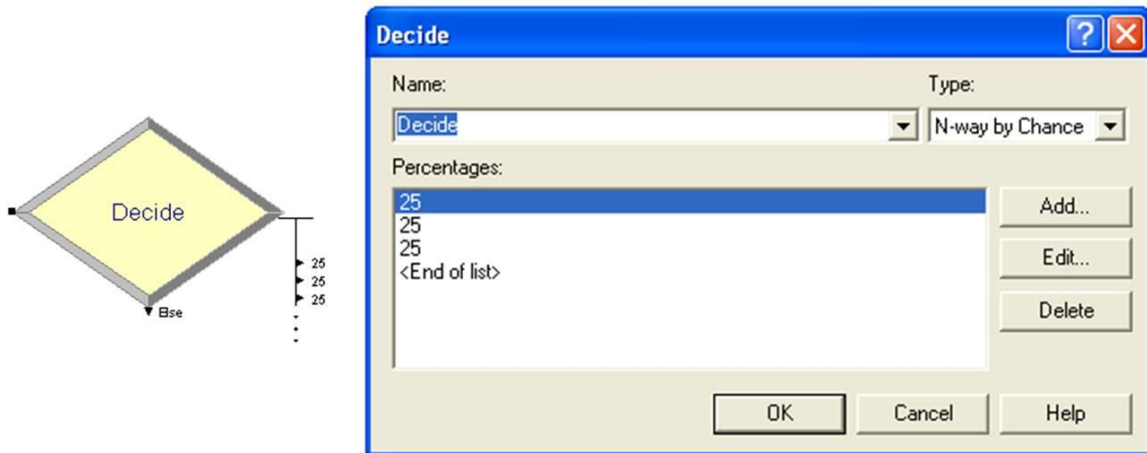


FIGURE 3.3 – The Decide Module and Accompanying Dialog Box

The Separate module is a unique module that allows the user to make duplicates of an entity [14]. The value of this module lies in the fact that other processes can use the duplicates of the entity, while the real entity continues to flow through the system. For example, perhaps an entity (client) flows into a Process module (receptionist station), which represents an interaction with a resource (the receptionist). When the client is finished with the meeting, he or she will continue to flow to the next step in the system. However, there may be instances where the receptionist is busy with paperwork related to that client for a period of time following the interaction. Therefore, the Separate module can be used to keep the resource seized with the duplicate entity, while the original keeps moving downstream, as it would in real life. The Separate module and dialog box are shown below in Figure 3.4.

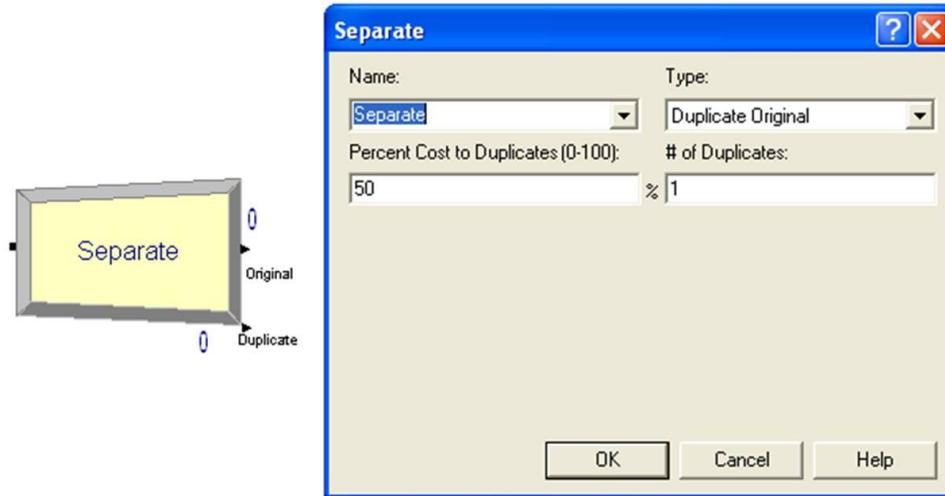


FIGURE 3.4 – The Separate Module and Accompanying Dialog Box

The Assign module is used for allocating attributes to entities, changing entity types, adding new values to variables, or creating global variables. An Assign module allows multiple assignments to be made in one step, but an entity must pass through it for the changes to take place. Assign modules offer a way to categorize entities and manipulate the model. The Assign module and dialog boxes are shown below in Figure 3.5.

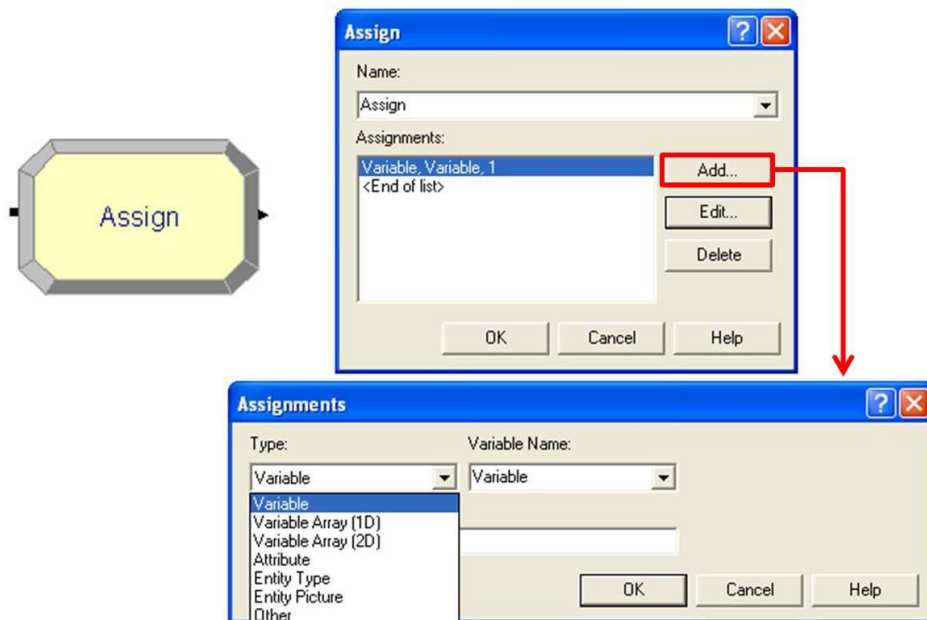


FIGURE 3.5 – The Assign Module and Accompanying Dialog Boxes

The Record module differs from other modules in that it does not affect the system when an entity passes through it. Rather, it is used to collect different types of observational statistics, such a count of number of entities that pass through it, statistics associated with the entity at that time, time interval of the entity, and others. Record modules are very important for collecting user specified statistics, which can be an aid for further analysis of a system. The Record module is shown below in Figure 3.6.



FIGURE 3.6 – The Record Module and Accompanying Dialog Box

Finally, the Dispose module is simply the exit point for entities in the system. Statistics can be collected through a Dispose module, although they are only the pre-selected statistics built into Arena<sup>®</sup>. A model must have at least one Dispose module and can have more than one if the system deems it fit. The Dispose module and dialog box are shown below in Figure 3.7.



FIGURE 3.7 – The Dispose Module and Accompanying Dialog Box

Apart from the flowchart modules, Arena<sup>®</sup> also uses data modules, which are more descriptive elements that work across the whole model. The data modules contain various values, expressions, and other characteristics for process elements, like entities, schedules, variables, resources, sets, and queues. Unlike flowchart modules, an entity will never flow in or out of data modules; instead the modules work “behind the scenes,” controlling different statistical conditions in the model [14]. The data modules are organized in a list format, with unique characteristics for each object, as shown below in Figure 3.8.

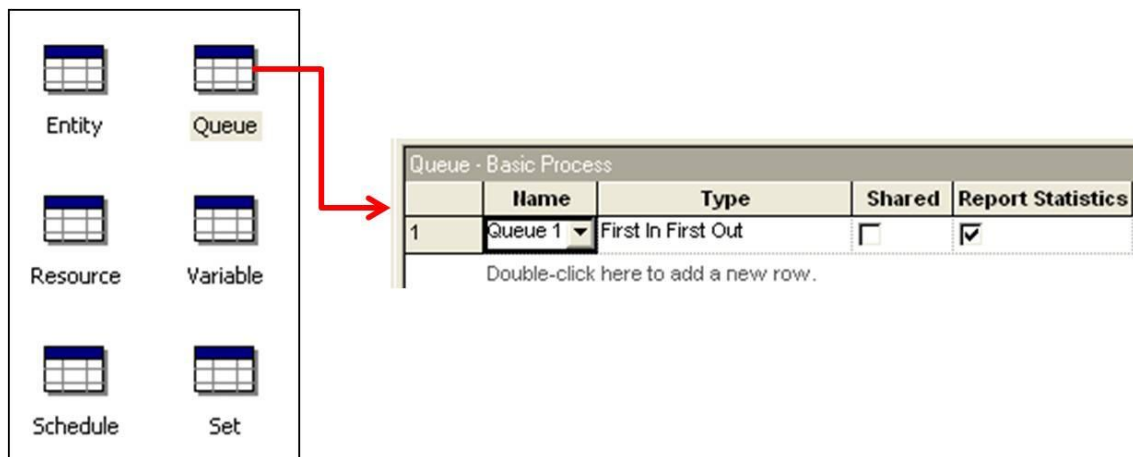


FIGURE 3.8 – Data Module List and Editing Box

These basic building blocks in Arena<sup>®</sup> allow the user to create customizable flowcharts and processes. The flowchart and data modules work together to make model creation and editing easier. By having the process represented as a flowchart, the user can view the system as a whole, allowing for frequent visual analysis.

#### IV. MODELING THE CHFS SYSTEM

Deciding how to model a system is a multi-step process that is unique to every situation. This outline of a simulation study, presented by Kelton [14] et al., describes key aspects to successfully building and presenting a model. It is important to note that these steps are not a “formula” per se; instead, these are frequently used parts that pertain to most studies. For this study, the steps of the process are as follows: 1) formulate the problem, 2) understand the system, 3) establish clear goals, 4) formulate the model representation, 5) collect accurate data, 6) translate into modeling software, 7) verify the model runs as intended, 8) validate the model, 9) design and run the experiments, 10) analyze results, and 11) document and report findings.

##### **Problem Formulation**

Problem formulation was done as a team, with members of the CHFS, Kent School of Social Work, and Speed School of Engineering. The group met and discussed the problem, each providing pieces of the proposed solution. The simulation of the system is only a small part of the whole solution, but for the sake of this paper, will be the only solution discussed. Once boundaries for the problem were set, the team established



future meeting dates for sharing information and project updates. For more information on the problem statement, please refer to the Introduction in the first section.

## **Understanding the System**

To have complete knowledge of such a large system takes a great deal of time and analysis. With a system so complex, understanding the system will also need to incorporate involvement from people who have been working with the system on a day-to-day basis. It was determined that the best course of action would be through immersion: having a working space amongst the workers to witness the system firsthand. Interviews with workers and other personnel were conducted as well to fill in information that could not be collected through simple observation.

The next step in understanding the system was to map it out, using Value Stream Mapping techniques. The Value Stream, shown below in Figure 4.1, is a high level overview that depicts the actions typical clients will take from the time they enter the building until the time they leave. Since this is a high level approach to first understand the system, process times were not documented at this point. The process steps are colored to show how the client's time is divided up while they travel from station to station. Green denotes a value added process, something the client is willing to wait for [23]. Rother and Shook define value added as an action or process that the customer is willing to pay for, with money being their asset. In this case, the client is not paying anything, but time is the most valuable asset, which is what the process steps are judged by. Yellow denotes a non-value added process, an action that the client doesn't want to wait for, but is necessary to the system. Red denotes waste, a process most commonly

associated with wait time or redundant paper. Blue is physical transportation, moving from one area to the next. The goal of this type of flowchart is to visualize the waste and determine the problem areas so a closer look at those processes can be taken.

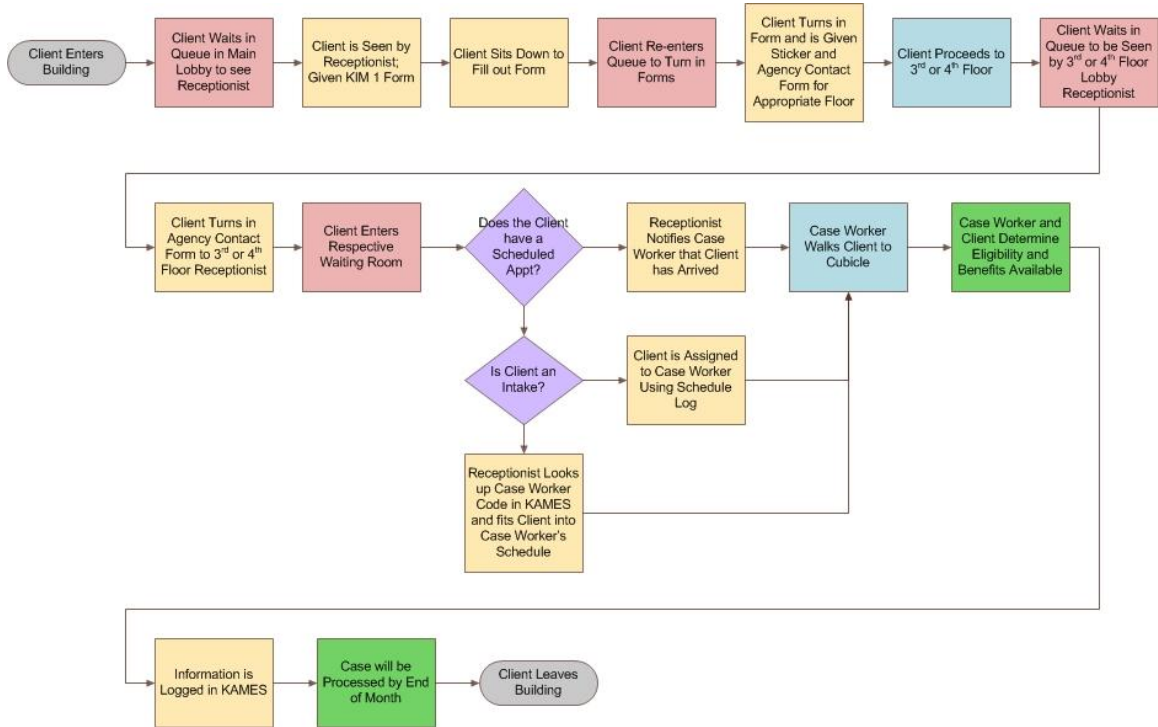


FIGURE 4.1 – High Level Value Stream Map of the Client Flow Process

As shown in Figure 4.1, the process begins as clients enter the system. The clients may be coming in for a number of different reasons, such as applying for new programs, recertifying their information to retain current benefits, or simply to drop off paperwork. Whatever the reason for the visit, clients are directed through one entrance and must wait in line to be seen by the receptionist. When the client approaches the receptionist, he or she will check in and be given the KIM-1 and Agency Contact forms to complete. The client then takes a seat in the lobby to complete the forms. Upon completion, the client re-enters the line to see a receptionist and turns in the KIM-1 form. The receptionist will take the form and check to see if the client is an active client in the system (whether or

not they're currently receiving benefits of any kind) or recently had an active case. If the client is active, then he or she will be given a visitor sticker to grant him or her access to the correct floor. If the client is not active, then he or she will be assigned to a floor based on residential zip code and services the client applying for.

The clients then proceeds to the respective floor denoted by their visitor sticker, taking along with them the Agency Contact form. In this study, only the 3<sup>rd</sup> and 4<sup>th</sup> floors are being analyzed, which encompass the Family Support Division. When the client reaches the appropriate floor, he or she waits in line to be seen by the receptionist. At the receptionist's station, the client will sign in and hand the receptionist the Agency Contact form. The client will then be directed to the waiting room and instructed to wait there until the case worker escorts him or her to the meeting.

The receptionist will check to see if the client has a scheduled appointment with a case worker. If so, the receptionist will verify the client's information in the KAMES computer system, then deliver the Agency Contact form to the respective case worker. If the client is not scheduled, the receptionist will check to see if the client is active in the system or not. If the client is active in the system, the receptionist will deliver the Agency Contact form to the client's respective case worker. If the client does not have a scheduled appointment and is not active in the system, they are considered an "intake" client. Intake clients will be assigned to a case worker based on a block schedule system, which looks at scheduled appointments for a whole floor, and fits into a case worker's schedule for that day. If the wait is going to be lengthy, the worker will let the client know and give the option of scheduling an appointment in the future or waiting to be seen

that day. Due to high volume of scheduled clients in the first fifteen days of each month, unscheduled clients often have very long wait times.

When the time comes for the client and the case worker to have an interview, the case worker will go to the waiting room and call the client back to his or her cubicle. During the interview, the case worker will first find out what benefits the client is seeking, then determine if he or she is eligible to receive those benefits. If the appointment was a scheduled appointment, the client should have brought various documents to the meeting, such as driver's license, social security card, pay stubs (if applicable), rent statement, and others depending on the circumstance. If the client was not scheduled or did not bring the correct forms to the meeting, the worker will give the client a Request for Information (RFI) form that lists all the verification paperwork needed. This form can be mailed back to the case worker or hand delivered. Each benefits program that the client applies for has different policies and procedures that need to be followed, so the meeting time is highly variable. The various programs and benefits chosen for this project will be discussed later.

Once the meeting is concluded, the client will be escorted back to the lobby on the respective floor by the case worker. The client will then exit the building. Depending on how busy the case worker is, the case will either be processed as soon as possible, or later on in the month. All cases need to be processed by the end of the month so that a client will be able to receive the benefits at the start of the next month.

Based on observation and interviews with various workers, the current client flow process only has two value added steps: the interview and the case worker processing the case. Many of the steps are non-value added, but necessary to the process. Since this is a

government organization, there are some parts of the process that cannot be changed because of federal or state laws. However, there are many steps in the process that are pure waste, with the client waiting on the next process. The flowchart identifies four key steps in which the client must wait, often for lengthy amounts of time. While it may not be feasible to completely eliminate these steps, it is possible to reduce the amount of time an average client waits at each area. These areas of pure waste are the target points that the team wishes to change.

### **Establishing Clear Goals**

With a better understanding of the system and awareness of the problems at hand, the next step was to establish goals for the project. This step required consulting from different groups involved in the project to determine what methods would be best to approach the problem, what metrics and statistics were most important to the state, boundaries of the project, and desired results. The goals took time to create and refine, making sure they were realistic and achievable. The list of goals is as follows:

- 1) Create a valid simulation model replicating the system as closely as possible. This will give all members involved a better understanding of the system and a strong model to work with.
- 2) Collect reliable statistics for the model. If statistics are available in spreadsheets or databases, those will be used. If statistics for a process or step are not documented, then they will be collected by observation, or survey.

- 3) Communicate effectively between all members of the project. Since there are a large number of people directly and indirectly involved with this project, communication is vital. This also includes getting buy-in from members of the CHFS, because without their support and cooperation, the project will go nowhere.
- 4) Have members of the CHFS or Kent School of Social work use the model as a means to experiment with theoretical changes to the system. By having a valid model, the team can see the impact a change would have on the system.

The goals listed above came from a larger list that was narrowed down over time. As previously stated, this project is part of an overall larger project to bring about change to the CHFS. One overall goal of the large project is to reduce wait times for clients while not overwhelming case workers with an increased workload. Therefore, a secondary objective of this project is for the simulation model to help achieve the goal for the large project.

### **Formulating the Model Representation**

This step ties in with the Value Stream Map shown in Figure 4.1 with the flowchart being analyzed to determine what kind of detail would be needed to make the simulation model most effective. The first decision was to create the model to follow clients as they flowed through the system. The decision was made after debating whether it would be most effective to focus on the client or the case worker in the model. It was agreed that

both perspectives should be analyzed, but this simulation would focus on the client to reduce their wait times in the system.

Another decision made was to categorize clients based on three factors: 1) whether they were scheduled or not, 2) whether they were applying, recertifying, or visiting for another reason, and 3) what services or benefits they were actually receiving. This level of detail required much data collection, but was deemed important for an accurate representation of the system. Table 4.1 depicts the categorization in more detail.

TABLE 4.1  
CATEGORIZATION OF CLIENTS FLOWING THROUGH SYSTEM

<b>Categorization</b>	<b>Factor</b>	<b>Variations</b>
<b>Main Type</b>	Is Client Scheduled?	Scheduled - Client has an appointment scheduled.
		Walk-In - Client is active in system, but does not have scheduled appointment.
		Intake - Client is not active in the system and does not have a scheduled appointment.
<b>Subtype 1</b>	Purpose of Visit?	Applying - Client is applying for programs not currently receiving.
		Recertifying - Mandatory periodic recertification of current information to retain benefits.
		Miscellaneous - Drop off paperwork, ask question, report changes in financials.
<b>Subtype 2</b>	What services?	SNAP - (Supplemental Nutritional Assistance Program) Food stamp benefits program.
		Medicaid - Health insurance program.
		KTAP + SNAP - Combination of Kentucky Transitional Assistance Program (cash benefits) and SNAP.
		KTAP + Medicaid - Combination of KTAP and Medicaid benefits.
		KWP - (Kentucky Works Program) Transitional program to get clients to work or volunteer for benefits.
		Other - Any other combination of listed programs.

Each client in the system will have a main type, subtype 1, and subtype 2, and will be routed to either the 3<sup>rd</sup> or 4<sup>th</sup> floor allowing for a total of 72 combinations, using the fundamental counting principle [24]. Although Table 4.1 breaks up the unscheduled clients into two categories, both are considered the same main type. The subtypes for services (subtype 2) were chosen based on the highest volume of benefits given to clients. SNAP and Medicaid benefits were the largest in volume, so greater focus was given to ensuring those numbers were as accurate as possible.

Another decision for the model was to only replicate a single busy day in the system. As described in the Introduction, a busy day denotes one of the first fifteen days of a month where case workers schedule the majority of their clients. This decision was made after the decision to follow the flow of the client through the system. Since the processing of the paperwork generally happens in the last half of the month and the client is usually not present, it is not contained in the model.

These decisions helped give shape to the model as it was forming and guided the collection of statistics. Other decisions were made along the way when situations arose, and are discussed later in this work.

### **Collecting Accurate Data**

Once a preliminary outline of the model was constructed, the next step was to collect the data to put into the model. The data used in this project comes from a number of different sources, such as in-house historical data, external historical data, survey results, observation, and worker logs. The focus of this section is to explain the methods of data collection, while the actual data is discussed in the next section.



The client arrival rate was one of the most basic and important pieces of information to gather. This data was collected through observation during the capstone project by recording the number of clients coming into the system. There were two obstacles when collecting this information: 1) determining how many people in a group that arrived together were considered clients and 2) only counting the clients going to the 3<sup>rd</sup> and 4<sup>th</sup> floors. Both of these problems were solved by observing how many KIM-1 forms a group was given and looking at the visitor's stickers for each client. The data was checked with the sign-in log books to ensure accuracy. This data was collected over time course of the project, first from February to March of 2010 and then revisited from October 2010 to January 2011.

With arrival rates collected, the next step was to determine the breakdown of clients who were scheduled, intakes, and walk-ins. Most of this information was collected through individual case worker logs, but also through internal historical data. Records were kept showing the number of intakes per period of time, while the percentage of scheduled clients was estimated using historical data logs. Percentage of walk-in clients was determined by finding the average number in the system on a busy day and subtracting the average number of intakes and scheduled clients. While collecting the data for scheduled clients, an interesting fact arose. According to case workers' logs of scheduled clients, only 49.3% of clients who had a scheduled appointment showed up at the time of their appointment. This data was collected from January 2010 to January 2011.

To determine the subtype 1 statistics, which relates to the purpose of the client's visit, historical records were analyzed. These records showed statistics for type of case,

how many recertifications and new applicants there were each month, which team they were assigned to, and what programs they applied for. These statistics became vital when showing the difference between the 3<sup>rd</sup> and 4<sup>th</sup> floor in terms of number of applicants processed per period per floor. The administrative data was collected from September 2010 to March 2011.

Subtype 2 statistics, which are the types of services the client is receiving, took a bit more time to collect. The general distribution of clients' services could be taken from historical data over several months, but processing times were also needed. This data was collected by observing as many case worker – client interviews as possible. These observations were done during the first half of a month. Case worker experience was a concern for the team, as it would affect processing times greatly, so various case workers were observed with different experience levels. Observing the interviews allowed for statistics to be collected on clients, who could be classified with a main type, subtype 1, and subtype 2 during the interview process. By having real, observed data for processing times (as opposed to estimates), the model should generate more accurate results. This data was collected at various times with random case workers from September 2010 to February 2011.

Two more types of statistics were needed for the model that required observation: receptionist processing times and case worker down time. Receptionist processing times were tracked for each lobby separately (main lobby, 3<sup>rd</sup> floor, and 4<sup>th</sup> floor), simply by observing clients as they came into the lobby. Case worker down time came later after running initial tests with the model. The team realized that downtime needed to be modeled into the system, so this statistic was also tracked through observation. Much of

the downtime occurred between client interviews, however it varied from case worker to case worker and day to day, depending on how busy the system was on that given day. However, since this model is representing an average busy day in the system, the best estimates and averages were used. These times were collected from January to April of 2011.

### Translating into Arena®

With the bulk of the statistics collected, the next step in the process was to put the statistics into Arena®. This part of the report will serve as a graphical guide detailing the construction of the model itself. This section will also give a more detailed analysis of some of the statistics used for the processes. The full model is shown below in Figure 4.2, which does not include the animation steps. Due to the small size of the image, each portion will be broken down and explained, starting from the left and moving to the right. A larger image of the full model is also supplied in Appendix I.

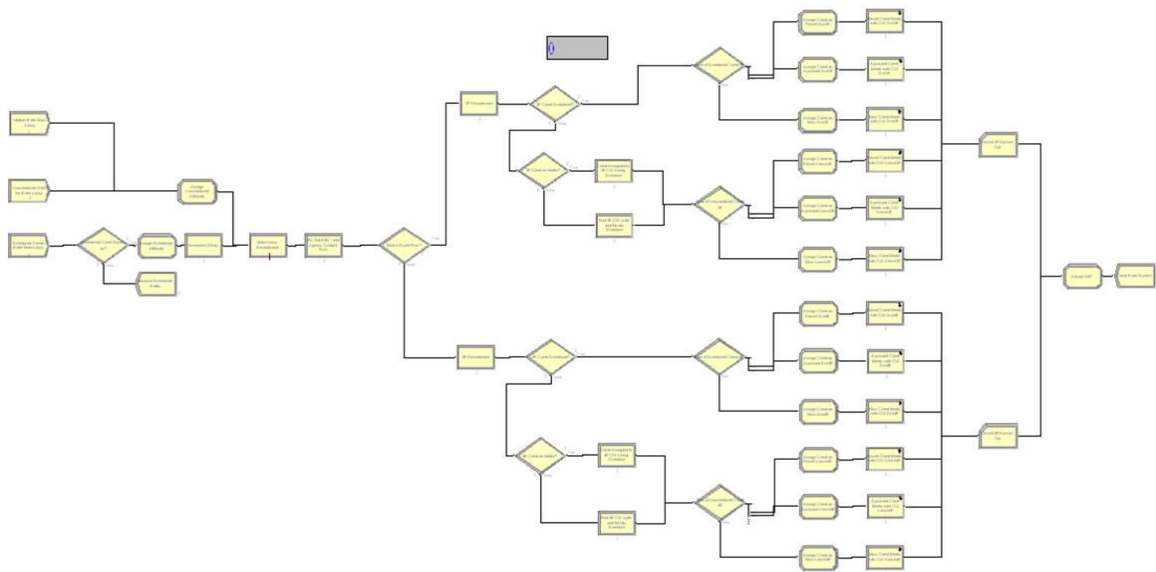


FIGURE 4.2 – Full View of the Arena Simulation Daily Model

Before jumping into the model processes, a few parameters must be set first. Arena<sup>®</sup> allows the user to control specific options in the model in the Run Setup dialog box, as shown in Figure 4.3. Kelton et al., point out the danger in having too few replications in a simulation model [14], so the “Number of Replications” field is set to 100 to reduce variance and have reliable averages in the output. It is also important to note that the base time units were changes to minutes for the whole system. Every expression, distribution, and statistic throughout the model will be in terms of minutes, unless otherwise noted. There are two terminating conditions for this model listed in Figure 4.3: 1)  $TNOW \geq 540$  and 2)  $WIP = 0$ . The first condition uses the built-in Arena<sup>®</sup> syntax  $TNOW$ , which represents the run time in the system. The system will run until the time is greater than or equal to 540 minutes, or 9 hours, which simulates a typical 8AM – 5PM day. The second condition states that the system will not finish running until Work in Process (WIP) equals 0. The model defines any clients in the system as WIP, so all clients must finish traveling through the system before each replication is complete. Both terminating conditions must be met for each replication to finish. Since rules and regulations mandate that clients must be seen when if they enter the system, occasionally a client will still be in the system after the nine hour mark, causing the system to run longer past 5PM. For the system to replicate this, it is not a problem. However, when this happens in the actual system, it is a problem. Overtime has been restricted for all workers, so they feel pressured to finish as quickly as possible if they’re staying late, sometimes at the cost of quality.

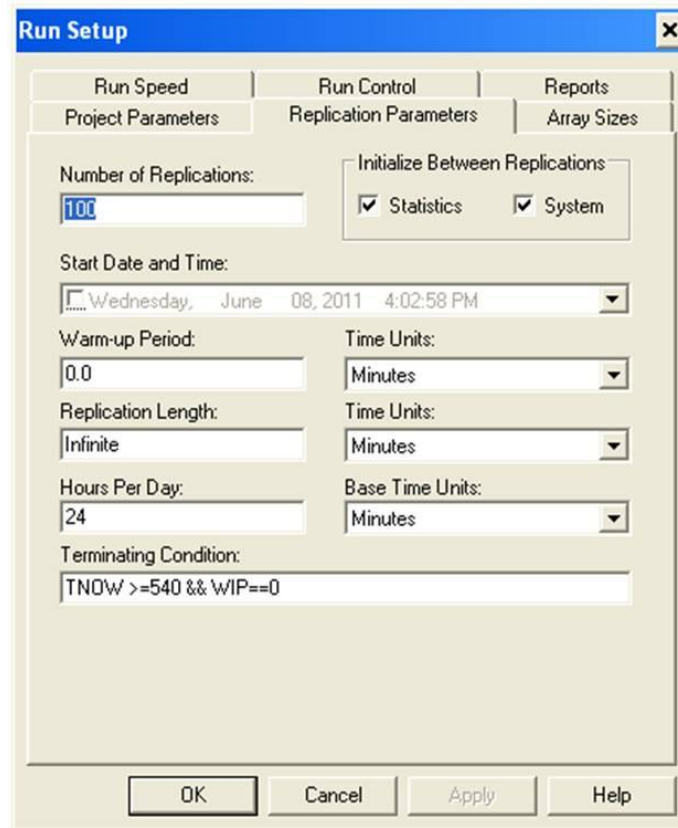


FIGURE 4.3 – The Run Setup Dialog Box with Parameters for the Client Flow Model

The model begins with three different Create modules, as shown in Figure 4.4. The first Create module brings intake type clients into the model, the second brings walk-ins, and the third brings scheduled clients. The main type is denoted by the entity type, which arrive based on a schedule. In Figure 4.4, the dialog box for intake Create module is also displayed, showing the inputs for the module. The Create modules for walk-ins and scheduled clients are similar, except they each have a respective entity type and arrival schedule. The data for the “intake” schedule, “walk-in” schedule, and “scheduled clients” schedule can be found in Appendix II.

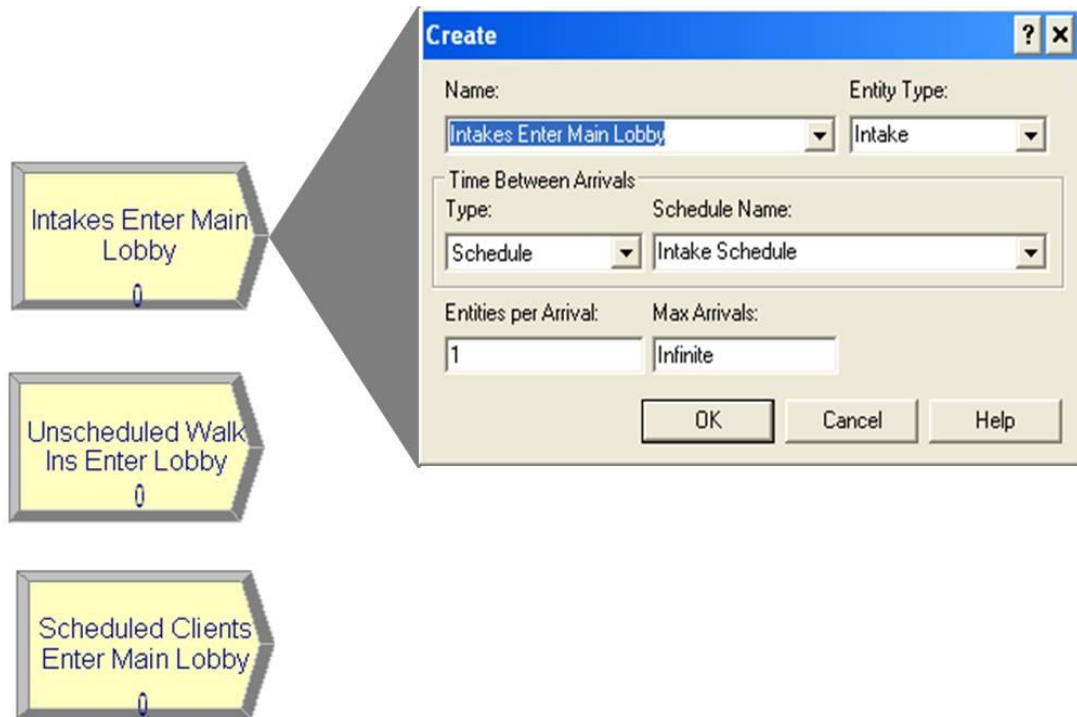


FIGURE 4.4 – The Create Modules for Intakes, Walk-Ins, and Scheduled Clients with Dialog Box for Intakes

Once the correct entities are created and enter the system, they are routed to an Assign module, shown in Figure 4.5, to give the attribute of “Scheduled Client” or “Unscheduled Client”. These attributes serve their purpose later on in the model, when clients will be divided by whether they were scheduled or unscheduled. Each Assign module also generates a variable that increases WIP by 1 unit each time an entity passes through. This helps keep track of how many clients are in the system at any given point. Earlier it was mentioned that only 49.3% of scheduled clients show up during their time block, so a Decide module was used to route 50.7% of the scheduled entities out of the system before passing through the Assign module so WIP would not be affected. Refer to Figure 4.2 to see the layout of the Decide module relative to the Assign module.

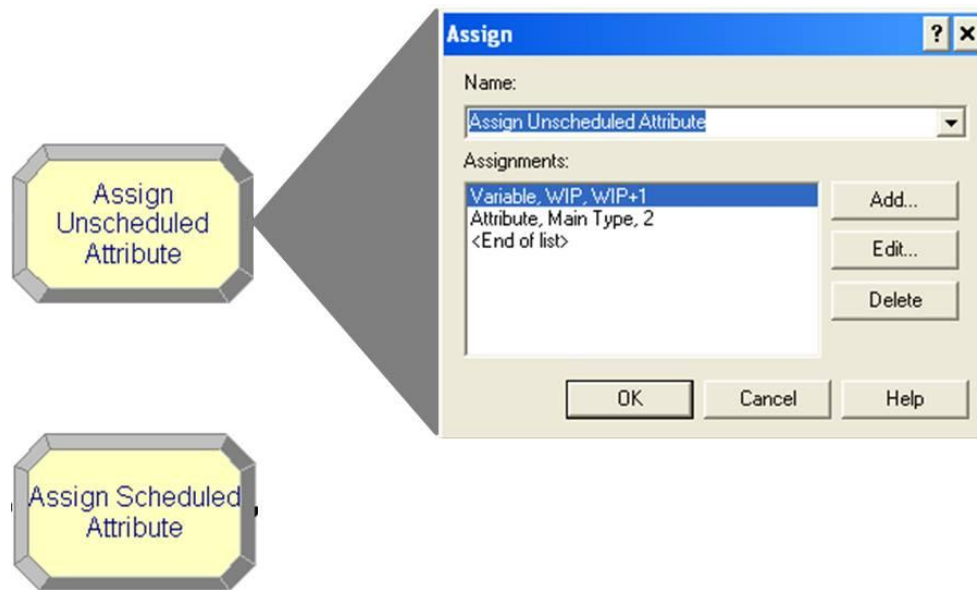


FIGURE 4.5 – The Assign Modules for Unscheduled and Scheduled Clients with Dialog Box for Unscheduled Clients

It is important to note one other unique item that pertains only to scheduled clients, which is a timeliness delay. It was found through observation, that even if scheduled clients do show up for their appointments, they are oftentimes late due to bus schedules or lack of transportation options. The Process module for this step is shown below in Figure 4.6, with the accompanying dialog box showing the distribution of the delays. A triangular distribution was chosen for the timeliness delay because of the definite minimum and maximum values. In this case, the client is given the benefit of the doubt, and the most likely delay time is 0 minutes, with a maximum of 15 minutes. From observation, most scheduled clients showed up on time, however some arrived up to around 15 minutes late. Generally, if a client had not yet shown up 15 minutes after the scheduled appointment time, it was assumed that he or she was not coming.

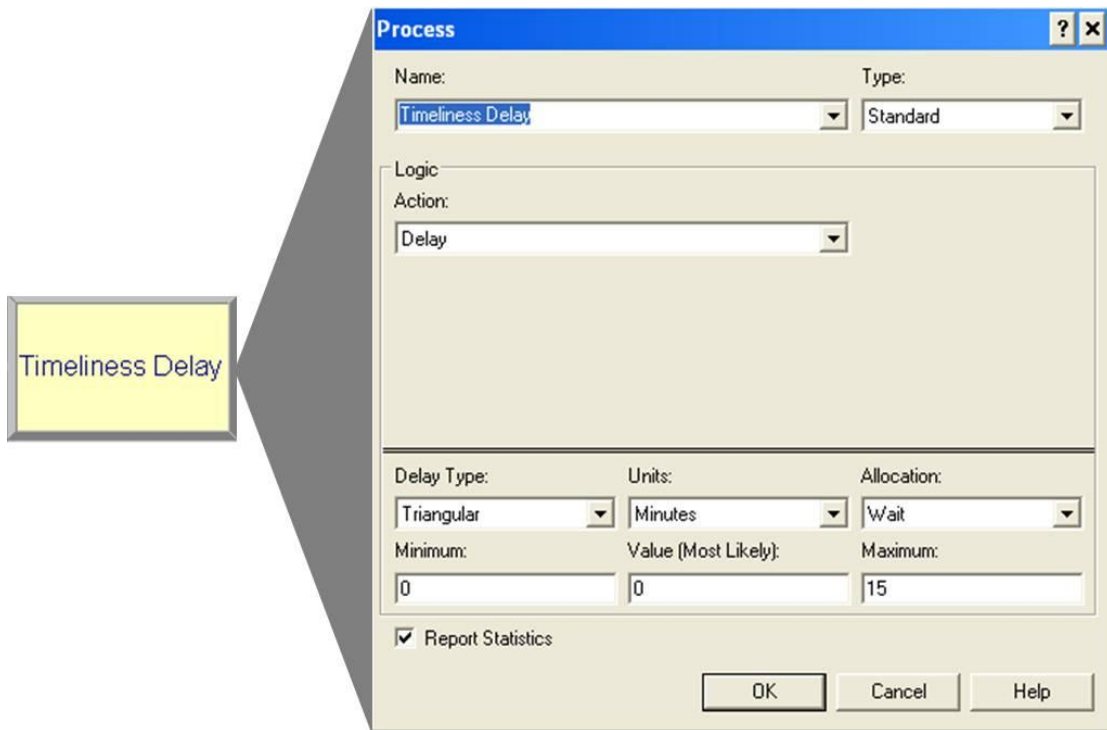


FIGURE 4.6 – The Process Module for the Scheduled Clients’ Timeliness Delay

Triangular distributions are useful for this particular process, and others mentioned later, because it gives the user the ability to state the minimum, maximum, and average values for a process. For this project, the minimum, maximum, and average values were observed and documented during the data collection phase. While there may be certain rare instances where a data point falls outside of the distribution, these are outliers in the system and do not improve the quality of the objective of the day model. Triangular distributions are denoted in Arena<sup>®</sup> by the expression  $TRIA(a, m, b)$ , with the minimum (a), mode (m), and maximum(b) containing values for the distribution specified as real numbers with  $a < m < b$  [14].

Getting back to the model, once the entities flow through the respective Assign modules, they join together and enter a single queue to wait for the main lobby receptionist. The main lobby receptionist is represented by a Process module, as shown



in Figure 4.7. The dialog box shows that the Process follows the Seize-Delay-Release logic, meaning that when an entity flows into this module it will seize a certain number of resources, then delay to represent processing time, and then release the resource so the next entity can seize it. For this process, the receptionist is the resource with a fixed capacity of 3, meaning there are always 3 receptionists ready to take on clients throughout the day. The expression for the delay type is TRIA(2,4,5), with minutes as the standard unit of time.

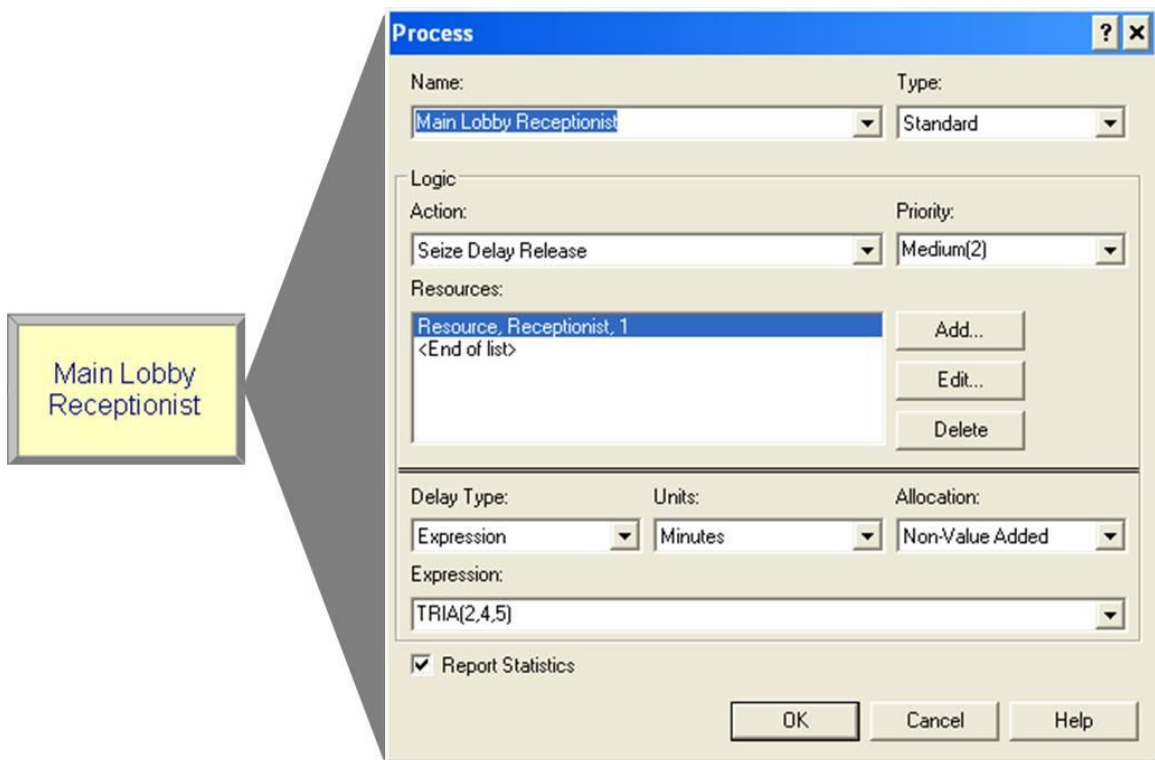


FIGURE 4.7 – The Process Module for the Main Lobby Receptionist and Accompanying Dialog Box

The next step in the model is another Process module to show the time it takes for a client to fill out the KIM-1 and Agency Contact forms. This module only uses delay logic, since no resources are required to complete the process. The expression used to represent this step is UNIF(1, 4), meaning a uniform distribution from 1 to 4 minutes.

The time to complete the forms depended on how quickly people wrote and how familiar they were with their current information. Observation times were fairly equally spread over the given range, which allowed for a uniform distribution.

The next step the client would follow in the process would be to re-enter the queue to turn in paperwork and get a sticker for a given floor. For the sake of modeling purposes, this step was combined with the Main Lobby Receptionist Process module. Experiments were run and it was decided that adding an extra step to show the client re-entering the queue just to turn in paperwork does not affect the model as a whole, but complicates the use of resources. Therefore, the steps were combined and the wait time appropriately adjusted so the client only sees the main lobby receptionist once.

The next logical function in the system is the Decide module routing clients to either the 3<sup>rd</sup> or 4<sup>th</sup> floor. When statistics were collected to determine the distribution of clients to the two different floors, it was found that roughly 48% of clients went to the 3<sup>rd</sup> floor, while 52% of clients went to the 4<sup>th</sup> floor. This may not seem like an uneven distribution upon first glance, but with a large volume of clients flowing into the system every month, a 4% difference can lead to greater work load over time. This problem is discussed further in the Results and Discussion of Results section.

Once the entity is routed to its respective floor, it will enter either the 3F Receptionist or 4F Receptionist Process module. Both modules are basically the same, just with different resources and slightly different processing times. The 3F Receptionist module, as shown in Figure 4.8, seizes the Third Floor Receptionist resource, while the 4F Receptionist module seizes the Fourth Floor Receptionist resource. Both resources have a capacity of 1, meaning that only one entity can be waited on at any given time.

The actual system has two receptionists working the lobby on each floor, but only one receptionist is authorized to use the computer (each floor has a full time receptionist and a KWP volunteer; the full time employee is the only one authorized to use the computer due to sensitive information being logged), creating a constraint. Since information must be logged into the computer, the resource capacity for each floor is set to 1. The processing times for the 3F Receptionist and 4F Receptionist modules are TRIA(1, 1.5, 3) and TRIA(1, 1.25, 3), respectively.

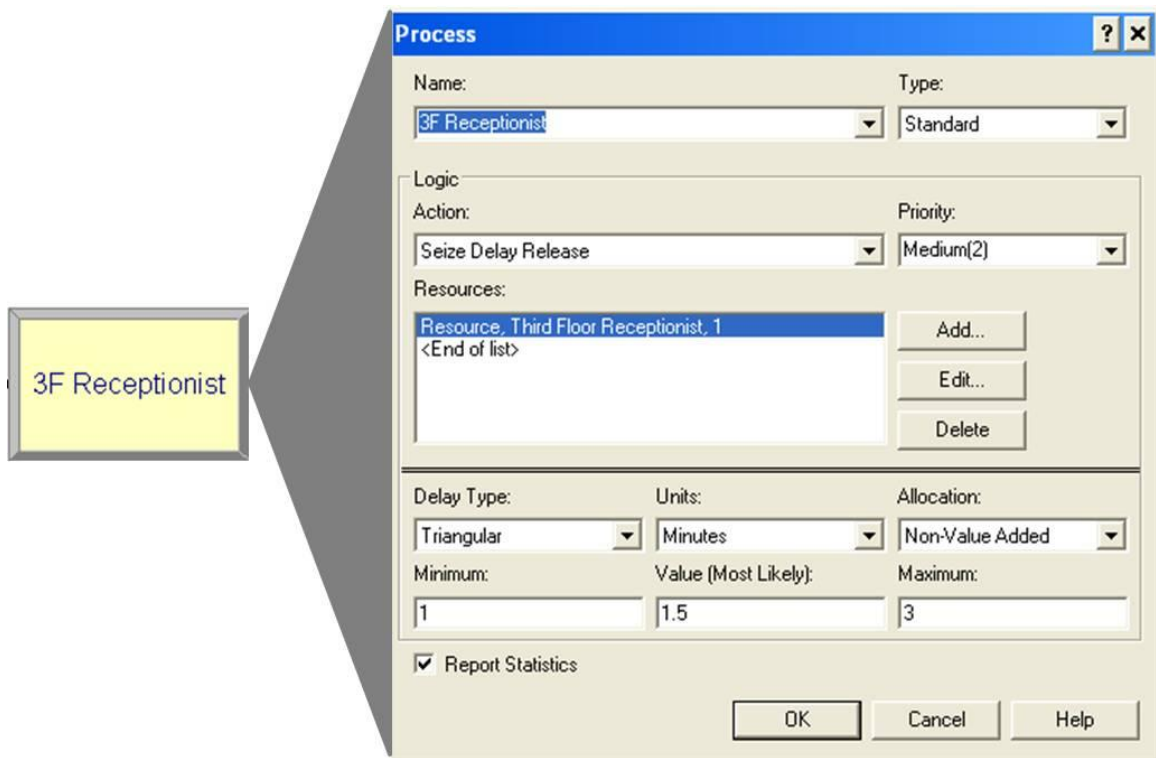


FIGURE 4.8 – The Process Module for the 3F Receptionist and Accompanying Dialog Box

Once the entity releases their respective receptionist resource, it will pass through a series of Decide modules, as displayed in Figure 4.10, to determine where to route the entity. The first in the series of Decide modules determines if the entity is scheduled or not. The logic for this Decide module, shown in Figure 4.9, simply states that if the

entity was created by the “Scheduled Clients Enter Main Lobby” Create module and is a “Scheduled Entity”, then it is assigned a “true” value, and is routed out the right side of the module. If the entity came from one of the unscheduled Create modules, then it is assigned a “false” value and routed out of the bottom of the Decide module.

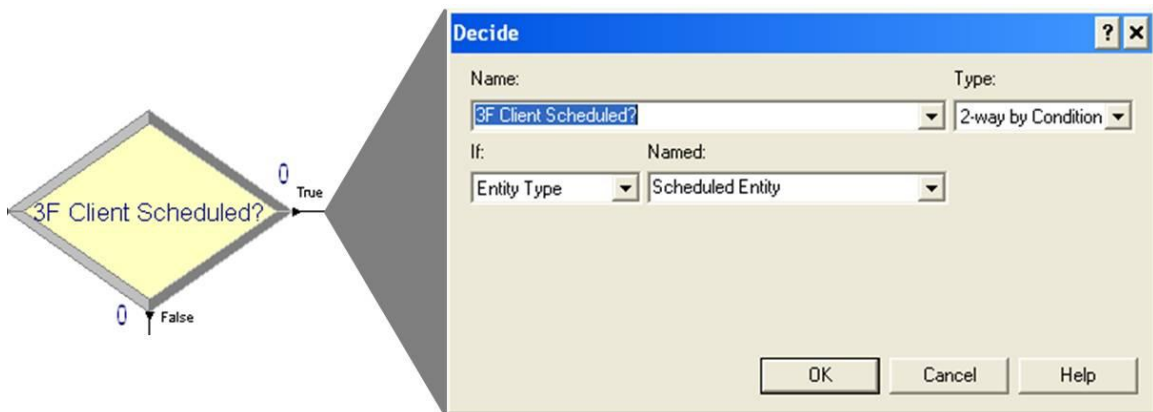


FIGURE 4.9 – The 3<sup>rd</sup> Floor Decide Module to Determine if Entity is Scheduled

Following the unscheduled entity, the next step is to determine whether or not the entity is an intake. This Decide module uses the same logic as the one shown in Figure 4.9, with the Entity Type being “Intake”. If the entity is an intake, it will be then be routed to the “Client Assigned to 3F CW Using Schedule” Process module (or 4F if on the 4<sup>th</sup> floor). If the entity is not an intake, it will go to the “Find 3F CW code and Fit into Schedule” Process module. These steps correspond to the actions taken when an unscheduled client visits his or her destination floor and want to see a case worker. Time is taken to review the master schedule for the floor to see if a time slot is available. It is important to note that this step takes place at the receptionist’s desk. However, both Process modules only follow the delay logic and never seize a resource, much like the Process shown in Figure 4.6. Again, this relates back to the fact that there are actually two receptionists at the station on the third and fourth floors. This step is usually

completed by the volunteer receptionist who does not have access to the computer, but not always. Depending on the circumstance, the full time receptionist will help to complete these steps, which would make mean a variable number of resources depending on the situation. When the station is busy, the full time receptionist will always be operating the computer to keep the line moving, as replicated in the model. To keep this step simple, the Process modules were made to just follow the delay logic to show the time it would take to get them assigned to a case worker. The distributions for “Client Assigned to 3F CW Using Schedule” and “Find 3F CW code and Fit into Schedule” Process modules are TRIA(0.5, 1, 1.5) and TRIA(1, 2, 3), respectively. In this step, the 4<sup>th</sup> floor modules use the same delay time distributions (Process module names are the same, with the exception of “4F” instead of “3F”).

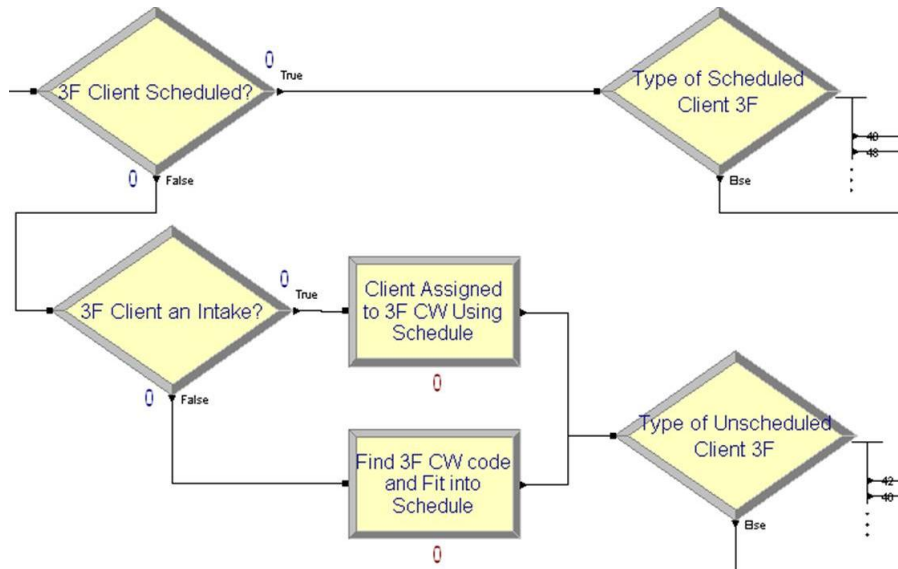


FIGURE 4.10 – Series of Decide Modules on 3<sup>rd</sup> Floor based on Entity Type

Referring back to Figure 4.10, the entity will now flow to either the “Type of Scheduled Client” or “Type of Unscheduled Client” Decide module for either floor. These Decide modules are not based on entity type, but rather probability. These

statistics were part of the cluster collected from historical data records. The Decide module itself is used to route the entity to one of three subsequent Assign modules, each representing a subtype 1 category and assigning the client as a recert, applicant, or miscellaneous. An example of this is shown in Figure 4.11.

The probabilities for these routes are different for each main type and each floor. For the “Type of Scheduled Client 3F” Decide module, roughly 40% of clients are routed to be assigned as recerts, 48% are routed to be assigned as applicants, and 12% are assigned as miscellaneous. For the “Type of Unscheduled Client 3F” module, the probabilities are 42% recerts, 46% applicants, and 12% miscellaneous. On the 4<sup>th</sup> floor, the probabilities for the “Type of Scheduled Client 4F” module are 40% recerts, 46% applicants, and 14% miscellaneous. Finally, for the “Type of Unscheduled Client 4F” module, the probabilities are 42% recerts, 46% applicants, and 12% miscellaneous.

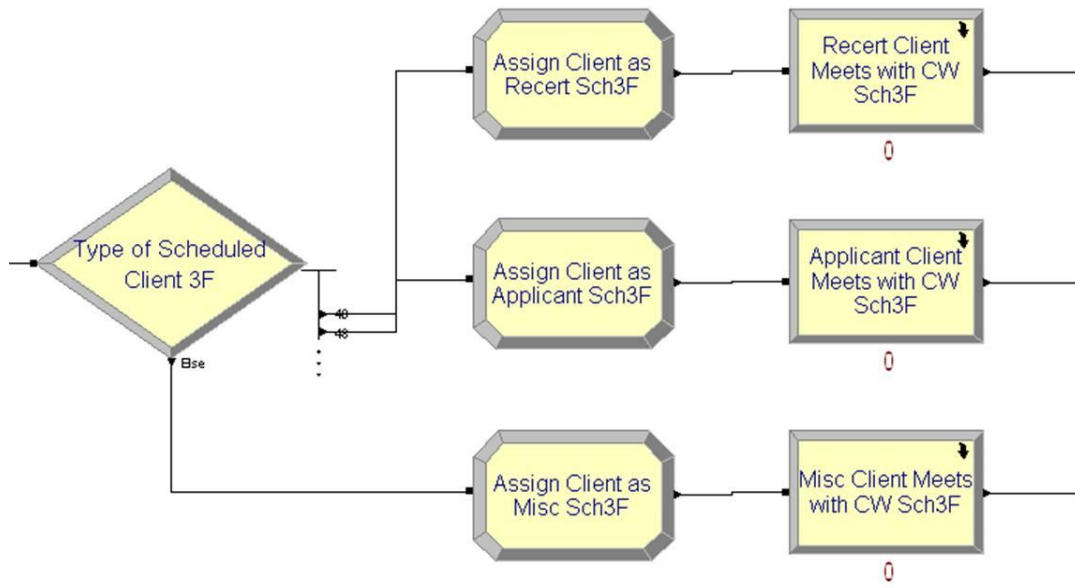


FIGURE 4.11 – Sequence of Steps for the Assigning of the Entity Subtype 1

Once the entity has been given a subtype 1 attribute by passing through one of the corresponding Assign modules shown in Figure 4.11, it will then travel to a respective

Process module, representing the interview between the client and the case worker. Up to this point, each entity in the system has been given a main type and subtype 1, and has been assigned to a floor. This leaves subtype 2 (services the client is applying for) as the last unique separating factor. However, it is noted earlier in Table 4.1 that subtype 2 contains six different categorizations for types of services. Representing this in one model window would take up large amounts of space and be hard to follow visually. Therefore, these steps are programmed into what Arena<sup>®</sup> calls “sub-models”.

Essentially, a sub-model is a model that allows for more customization and hierarchical steps contained within a single Process module [14]. A Process module that contains a sub-model is denoted with a curved arrow symbol in the top right corner of the module, as shown in Figure 4.12. When a Process module is changed into a sub-model, more modules are built within that sub-model that acts as a single step with multiple phases. When an entity enters a sub-model Process module, it will need to pass through the appropriate modules before exiting the sub-model. The entity starts at the left side, denoted by the triangle and moves to the right, finally exiting the sub-model at the denoted small square, as seen in Figure 4.12. Sub-models follow the same logic as normal models, so entities can be created and disposed with a sub-model. As the entity flows through the sub-model and exits back to the higher-level model, statistics are collected and updated upon its return. However, desired statistics for modules in sub-models must be defined by the user; otherwise they will be grouped and summed together under the highest level Process module which contains the sub-model. A larger version of the general model within each sub-model can be found in Appendix III.

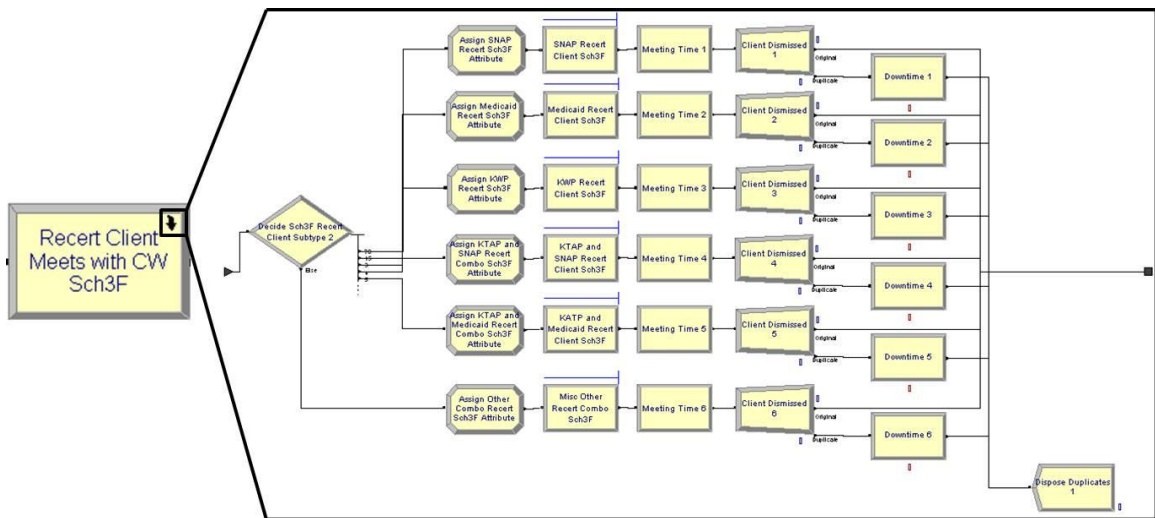


FIGURE 4.12 – Overview of Sub-Model for Scheduled Entities on the 3<sup>rd</sup> Floor with Recertification Meetings

Sub-models were the most appropriate method for representing client meetings for a number of reasons. First, it cuts down on clutter and allows the model to be viewed and edited more easily. Second, it creates a hierarchical approach to the model logic. Since everything within a sub-model shares the same main type, subtype 1, and floor number, it keeps the model organized, which is important with a large number of modules.

When an entity enters the sub-model, it must pass through another Decide module to send it to one of the six Assign modules to get a subtype 2 characteristic assigned to it. The probabilities associated with each Decide module come from administrative data showing the number of new applicants for each program as well as the number of clients who came back for recertification. Each route the entity takes for subtype 2 looks the same, but each has a unique processing time distribution attached to it. Table 4.2 below shows a summarized version of probabilities in each Decide module.



TABLE 4.2  
PROBABILITIES FOR DECIDE MODULES IN EACH SUB-MODEL

		Probability for Subtype 2					
		SNAP	Medicaid	KWP	KTAP + SNAP	KTAP + Medicaid	Other
<b>Decide Module Name</b>	<b>Decide Sch3F Recert Client</b>	70%	15%	3%	4%	5%	3%
	<b>Decide Sch3F Applicant Client</b>	72%	14%	3%	4%	5%	2%
	<b>Decide Sch3F Misc Client</b>	17%	17%	17%	17%	17%	15%
	<b>Decide Unsch3F Recert Client</b>	67%	18%	3%	4%	5%	3%
	<b>Decide Unsch3F Applicant Client</b>	72%	14%	3%	4%	5%	2%
	<b>Decide Unsch3F Misc Client</b>	17%	17%	17%	17%	17%	15%
	<b>Decide Sch4F Recert Client</b>	55%	30%	3%	4%	5%	3%
	<b>Decide Sch4F Applicant Client</b>	62%	24%	3%	4%	5%	2%
	<b>Decide Sch4F Misc Client</b>	17%	17%	17%	17%	17%	15%
	<b>Decide Unsch4F Recert Client</b>	55%	30%	2%	3%	5%	5%
	<b>Decide Unsch4F Applicant Client</b>	62%	24%	3%	4%	5%	2%
	<b>Decide Unsch4F Misc Client</b>	17%	17%	17%	17%	17%	15%

As mentioned earlier, there are six different categorizations for subtype 2 for each entity, as shown in Table 4.1. The top five most widely distributed benefit programs were chosen as categorizations, with one extra categorization added to include any other program or combination thereof. With the addition of this final subtype, the entity will have specific attributes attached to it which helps determine the processing times it needs for the interview.

One problem that was encountered by having a Process module for each possible type of interview was that Arena<sup>®</sup> gives a unique queue to every Process module. Even though each Process step dealing with case worker and client interviews pulled from the same pool of resources for each floor, lines would build up in different queue areas, which made client tracking difficult. This method was also not representative of the real

system, where clients waited in a single waiting room on each floor before being seen. To overcome this problem, shared queues were implemented for all Process modules representing the case worker and client interviews; one for the 3<sup>rd</sup> floor and one for 4<sup>th</sup> floor. Having shared queues allows for easier entity tracking as they flow through the system and is more representative of the real system.

To have shared queues, some advanced programming is required. Arena<sup>®</sup> allows the user to select Process modules that only perform partial functions, and then can be strung together to have a complete function. For instance, in each possible interview module, the Process module with Seize-Delay-Release logic is replaced by three different modules: one Seize module, one Delay module, and one Process module with Delay-Release logic. By breaking the steps down into multiple modules, more flexibility is given to the user.

After the subtype 2 Assign module, each interview sequence starts with a Seize module, such as the one displayed in Figure 4.13. It was previously stated that Seize modules offer the advantage of specifying which queue the entity will wait in, which can also be seen in the figure at the bottom of the dialog box. As the entity enters the Seize module, it immediately seizes a user specified resource, which for this work is either the 3F or 4F case worker. One more important note to make in this step is the priority system. After discussions and observations with case workers, it was found that precedence is given to scheduled clients over unscheduled clients in terms of order for interviews. Most case workers encourage clients to schedule their appointments to help be more prepared for the interview and streamline the process. To show the effectiveness of this and not be contradictory, a case worker will usually give precedence to a

scheduled client. The priority system can be seen in the top right corner of Figure 4.13, as this step is marked as High.

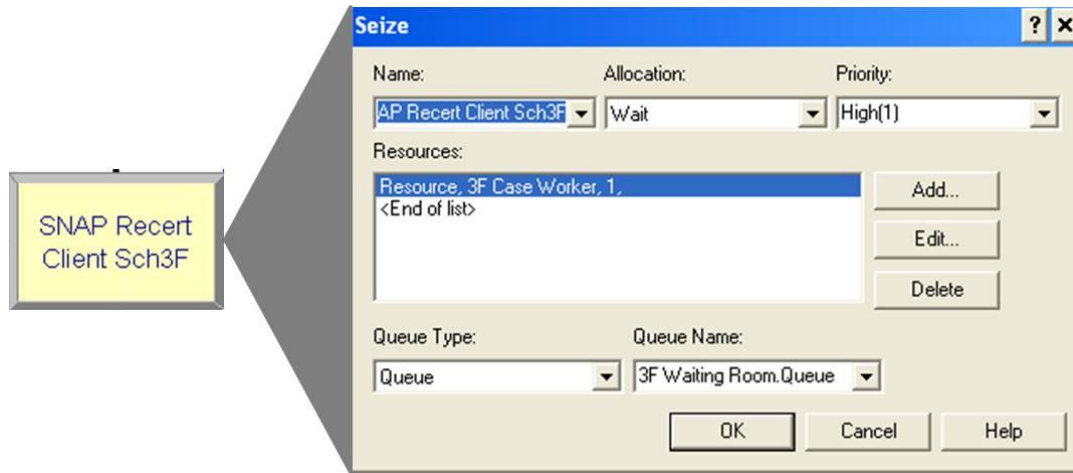


FIGURE 4.13 – Seize Module for a Scheduled SNAP Recertification Client on the 3<sup>rd</sup> Floor

With a resource seized, the entity will follow the flow of the system and continue into its respective Delay module, shown in Figure 4.14. The Delay module logic operates very similarly to a normal Process module, but the resource is already seized when the entity passes into it. Each Delay module is named “Meeting Time” followed by a sequential number, since each module name must be unique. Notice in the dialog box of Figure 4.14 that the Delay Time is allocated by “PrTIMESch3F(1,1)”. This term refers to the Expression spreadsheet data module, specified as PrTIMESch (process times for scheduled clients), and uses the value or expression in cell (x, y). In this work, x denotes subtype 1, and y denotes subtype 2. For example, “PrTIMESch3F(1,1)” refers to the process times Expression spreadsheet for scheduled SNAP (subtype 2) recertification (subtype 1) clients on the 3<sup>rd</sup> floor. Every user created Expression listed uses a triangular distribution to represent the interview times. For this step, depending on the floor and main type of the entity, one of four Expression spreadsheets will be accessed:

PrTIMESch3F, PrTIMEUnsch3F, PrTIMESch4F, and PrTIMEUnsch4F. Using Expressions to organize the process times keeps the data in one location, making editing and viewing much easier on the user. The spreadsheets used in the Expressions data modules can be viewed in Appendix IV. It is also important to note that the time the entity spends in the Delay module is allocated as value added, shown in the top right corner of Figure 4.14, since the interview with a case worker is the main purpose for a client visit.

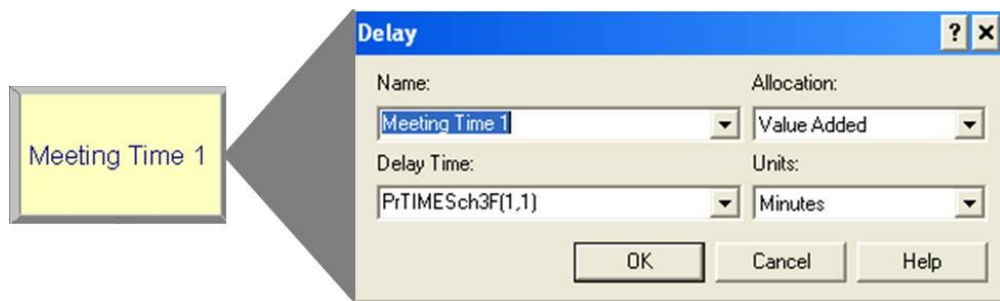


FIGURE 4.14 – Delay Module for a Scheduled SNAP Recertification Client on the 3<sup>rd</sup> Floor

Another concern the team had for the model was what to do when the time allocated by the Delay module expires. If the entity simply passed to a Release module, the resource would be released and could immediately be seized by another entity. In the real system, that is not the case. After the client is dismissed from the interview, the case worker will use the next little bit of time as downtime to finish up paperwork for a case file, return phone calls, get a drink of water, etc. Of course, the amount of time a case worker chooses to have for downtime is variable, but for the sake of this work, a triangular distribution was used to represent the downtime. This was modeled by the entity passing into a Separate module once passing through the Delay module.

Recall the Separate module and dialog box in Figure 3.4; when an entity flows into the module, a duplicate entity is created. The user chooses where to then send the duplicate and original entity. In this model, since the client has finished with their interview and is now free to go, the original entity is routed out of the top connector line of the module and exits the sub-model to return to the higher-level model. The duplicate entity is routed out of the lower right hand corner of the Separate module and then flows into the Process module, entitled “Downtime” followed by a sequential number, which employs the Delay-Release logic shown in Figure 4.15. Since the entity is a duplicate of the original, it still occupies the same resource that was seized for the interview. This allows the resource to be seized for a period of time even if the original entity has already exited the system. The expression used for this module is also stored in a specified Expression data module. However, in this case, the spreadsheet only contains one cell that is accessed by all the “Downtime” Process modules. The distribution in the spreadsheet is defined as TRIA(10, 15, 20), which is typical for the majority of caseworkers.

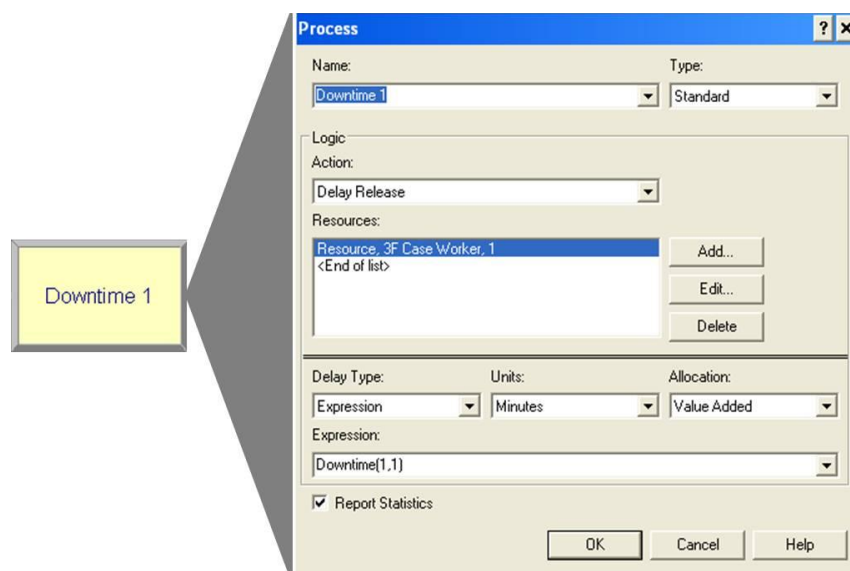


FIGURE 4.15 – Process Module with Delay-Release Logic for a Duplicated Entity

When the time expires in the “Downtime” Process module, the duplicate entity releases the resource and is routed to a Dispose module in the sub-model. The resource can then be seized by another entity waiting in the shared queue and the process begins again. Since statistics are not desired for this duplicate entity, the Dispose modules in each of sub-models have the box for “Collect Statistics” unchecked so duplicate entity statistics do not interfere with original entity statistics.

Returning back to the main level of the model, the entity will exit the sub-model and pass through a Record module, to gather statistics from the entity. There is a different Record module for the 3<sup>rd</sup> and 4<sup>th</sup> floor branch. Once the statistics are gathered, the branches from the 3<sup>rd</sup> and 4<sup>th</sup> floor are united again so that every entity will pass through one final Assign module before exiting. The final Assign module simply re-adjusts the WIP by having a global variable that decreases the WIP by one unit every time an entity passes through it. Once the entity passes through the Assign module and the WIP is adjusted, the entity will exit the system through the final Dispose module.

### **Verification of the Model**

Verification is defined as the process of debugging the model, ensuring that the model runs the way it was intended, and follows the guidelines set forth for the model [14]. During this project, verification steps were done over a period of time, not just once the model was complete. Parts of the model were added and changed over time as the real system changed. After each addition and revision, debugging had to be done to make sure the model was running as it was supposed to.

Kelton [14] et al., discusses one simple, yet effective method to debug a simulation program. In this method, any Create modules are replaced by one Create module that only allows one entity in the system. Slow the run speed down and watch the entity flow through the system to ensure it is behaving as it is supposed to.

Other checks were made to ensure verification, such as creation of animation, testing under extreme circumstances, and experimenting with different discrete distribution times for processes. These tests help define the limits of the system for the user, but also ensure that the model is what it is intended to be.

### **Validation of the Model**

Validation differs from verification in that it is the process of ensuring that the model is an accurate representation of the real system given the particular objectives of the project [17]. Validation is a difficult process and requires a number of different factors, some being very subjective. First, it must be understood that complex systems can never achieve absolute validity. The more time and effort that is spent on a model, the more valid it becomes, but this does not always lead to better insights on the system [17].

One method chosen as a form of validation was comparing the output of the simulation model to statistics from the real system. Only so many statistics were compared due to lack of metrics collected in the real system. Some statistics, such as number of clients per day and distribution of various programs, were compared with the simulation model.

Another form of validation for this project was buy-in and support from management at the CHFS. As most members of management had been working with the system for

many years, they understood how it should flow on a daily basis. A meeting was held to demonstrate the capabilities of the simulation model and discuss what could be done to improve accuracy in the model. Suggestions were made and the simulation model was edited based on those suggestions. A follow up meeting was conducted with the improved model to again ensure that the model was accurate.

Validation should be an on-going process. Each time a new experiment is run or factors change within the model, it should be re-validated. In this project, each time a feature was added or changed, the goals and objectives were once again reviewed to ensure that the model was still following the guidelines laid out at the beginning of the project.

### **Designing Experiments**

Referring back to the goals set forth early on in the project timeline, one was to be able to use the model to run experiments to understand how theoretical changes will impact the system. Suggestions were taken from members of the team and the CHFS for various experiments to demonstrate the model's flexibility. As a result, three designed experiments were chosen for further study and are as follows: 1) effects of a sign in kiosk in the main lobby, 2) effects of a 10% influx of clients into the system, and 3) an alternative method for processing clients on the 4<sup>th</sup> floor. This section discusses the experiments that were chosen, while the following section analyzes the results of the experiments.

The first experiment received its origins during the capstone phase of the project. The team agreed that having clients check into a kiosk in the main lobby should help



streamline the process. The idea behind the kiosk was for it to be linked to a network within the CHFS. When a client logged in with a unique ID or if the SNAP card could be swiped and unique information pulled from it, the system would recognize him or her and the client could be able to “check in” through the computer. The computer would then print a sticker with the appropriate floor for the client. For clients checking in to a scheduled appointment, the case worker would be notified on their computer if the kiosk is synced to the CHFS network. If the client is not scheduled, but active in the system, he or she could check in and see when the next available time is to meet with their case worker. If the client is an intake, there would still be main lobby receptionists to help the new client out. This experiment was designed to see if the process could be streamlined by reducing paperwork and having the client not enter the queue twice in the main lobby. The client would get a printed sticker from the computer and continue to their respective floor as usual.

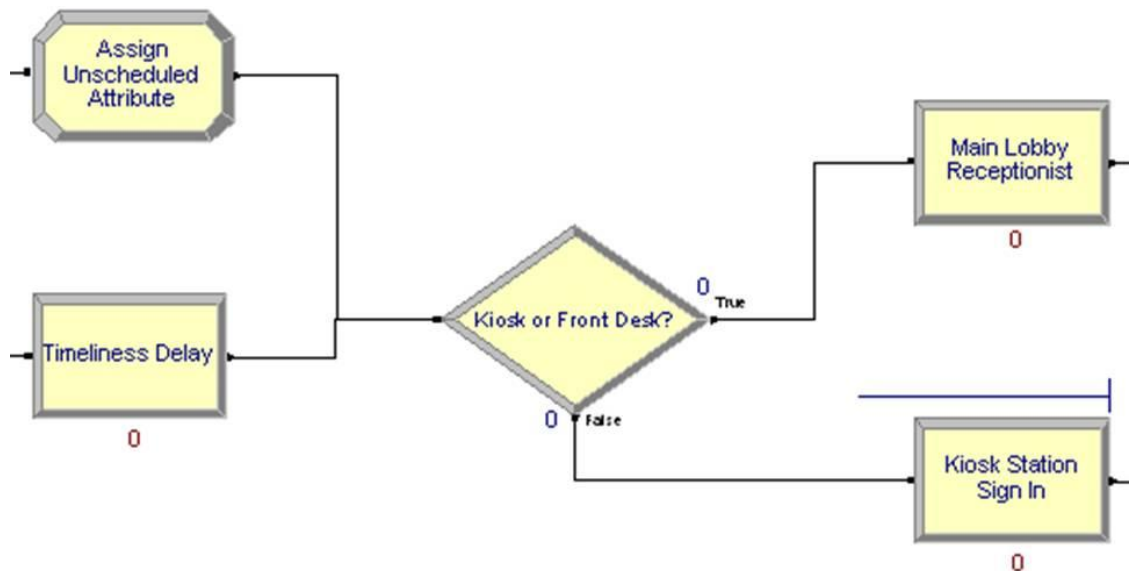


FIGURE 4.16 – Experiment with Additional Kiosk Station in Main Lobby to Reduce Traffic

A Decide module was placed before the main lobby receptionist Process module, as seen above in Figure 4.16, with a second branch allowing for clients to go either to the kiosk, or to the front desk. In this model, of the clients who were able to go to the kiosk, it is assumed that 65% would still choose the main desk, at least while the kiosk system was new. To simplify the process, the kiosk would be as user-friendly as possible. A preferable system would have the client only enter numerical values into the computer, thus reducing the probability of an error happening. Every other feature on the interface would be point and click, so that computer literacy would not be a factor. After observation from a similar system, it was decided that the distribution to represent this process would be TRIA(1,2,4). Since it is a theoretical distribution, the numbers stay on the conservative side. Three resources were used for this Process, to keep the clients flowing smoothly.

The second experiment to analyze is the effects on the system if a 10% influx of clients were to enter the system. Of course, this is a pure “what-if” experiment, but with a 9% increase in clients over the past year, it is a very plausible scenario. Economic uncertainty and healthcare reform are two driving factors for this experiment. The likelihood of the actual scenario happening is hard to determine, but seeing how the system reacts is valuable in itself.

The model for this experiment stays the same as the original, except that the client arrival Schedules are edited to allow 10% more clients into the system. Performance measures would be compared between the two models, such as average case worker utilization, client wait time, and average time the last entity exits the system. Other

iterations were run with this experiment to determine what factors, such as number of case workers, it would take to return the system back to its original state.

The third experiment chosen was an idea that members of the CHFS created and designed. The experiment looks at an innovative method of interviewing clients and processing paperwork. Instead of having multi-functional caseworkers on the 4<sup>th</sup> floor, they would be broken up into two groups: interviewers and processors. The interviewers deal with the face-to-face time with the client, collecting information. The processors, on the other hand, deal only with processing paperwork and with the client when there is missing information in the case. There's a 3:1 ratio of interviewers to processors, since the interview process is generally more time consuming. This experiment was put into place as a pilot program, with specialized teams volunteering to give the new system a try. Only the 4<sup>th</sup> Floor was used for the pilot program, so the model reflects this and changes were only made to that floor, while the 3<sup>rd</sup> was left unchanged.

To model this, changes were only made in the sub-model, as shown below in Figure 4.17. The first change made was to the module following the Assign module for every category. The logic in the Seize module still remains the same, only that a 4F Interviewer resource is seized. After being seized, the entity will pass into a Delay module, with unique processing times documented on the 4<sup>th</sup> floor. A Separate module is still used to duplicate the client entity as it exits the sub-model. The duplicate client flows into a Process module named "Process Paperwork", with a sequential number following it. In this process, the duplicate entity (now representing the client's paperwork) seizes the same 4F Interviewer resource and delays it for a small amount of

time. From there, the duplicate will be disposed while the original exits the sub-model and the process continues as it normally would.

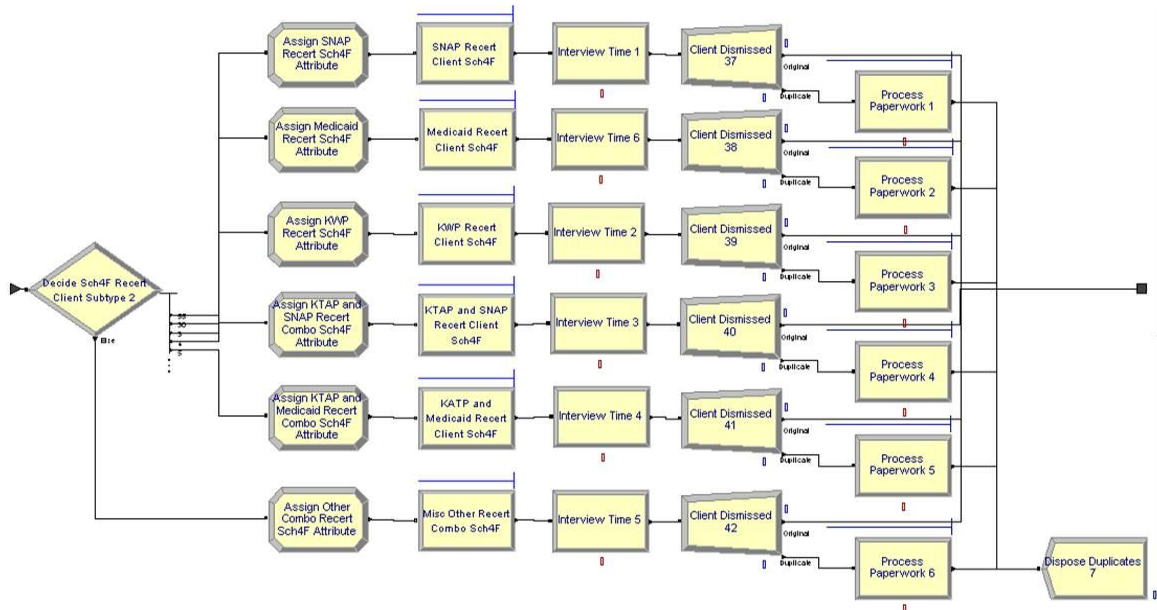


FIGURE 4.17 – Sub-Model with Changed 4F Process Using Interviewers and Processers

The downtime delay is much shorter in this experiment than the original model because the specialized interviewers only have to finish the documentation and pass the case file along to a case processor. Recall that the original method requires each case worker to process their own cases, whereas this revised process has separate interviewers and processers. It is important to note that times for processing case files are not included in this experiment, as it does not affect the client flowing through the system in a typical busy day. The case files are processed by the end of the current month, so that the benefits can be distributed on time.

In this experiment, the processing times for the 4<sup>th</sup> Floor are much different than the 3<sup>rd</sup> Floor because of the new interview method. After collecting data for the interviewers

and processors on the 4<sup>th</sup> Floor, it was found that interview times decreased on average. These changed times process times for this experiment can be seen in Appendix IV.

## V. RESULTS AND DISCUSSION OF RESULTS

Reviewing the steps listed in the previous section shows only nine of the eleven listed steps have been discussed. The last two steps, analyzing the results and documenting the findings, will be lumped together and discussed in this section. Once a simulation model is validated and verified, there is still some work to be done. The output Arena<sup>®</sup> provides the user must be analyzed to properly understand the generated report. Statistical measures, such as paired t-tests to compare means, were used to delve deeper into the output. There are also two other types of analysis methods discussed by Kelton [14] et al. that were used. Comparative analysis is used to analyze the best design out of a finite set of systems. Predictive analysis deals with one system and is used to help find the best performance of that system.

### **Present Model**

The present model was set to run for 100 replications so that the results would be reliable. For convenience, key metrics are summarized in Table 5.1 below. Additional statistics from the simulation output can be found in Appendix V. It was found that an average of 393 clients flow into the system per busy day to request a number of different services. Of the 393 clients that enter the system on average, 39.3% are intakes, 36.4%

are scheduled clients, and 24.3% are walk-ins. Note that the number of scheduled clients only includes those clients who show up for their appointments.

TABLE 5.1  
PRESENT MODEL SIMULATION OUTPUT RESULTS

Variable	Type of Client		
	Scheduled	Intake	Walk In
Average Client Wait Time	35.6 min	33.3 min	32.6 min
Max Client Wait Time	155 min	148 min	151 min
Average Client Time in System	76.1 min	78.9 min	79.4 min
Max Client Time in System	205 min	240 min	231 min
Average Work in Progress	11.8	39.6	12.3
Average Number Clients Per Day	143	154	96

Clients spent an average of 78.15 minutes in the system, with the minimum being around 12.9 minutes and the maximum being 240.4 minutes. These numbers show that the system can flow very smoothly at its best, but can also be very slow when strained. On average, scheduled clients flowed through the system most quickly, followed by intakes, then walk-ins with times of 76.1, 78.9, and 79.4 minutes respectively.

Of the time spent in the system, an average of 31.38 minutes (40.1% of total time) was allocated to value added activities, which mainly consisted of the interview time. Non-value added activities averaged 7.96 minutes (10.2% of total time), which included activities such as paperwork and check-ins. Travel time accounted for the smallest average time, at 5.0 minutes (6.4% of total time). However the largest portion of time spent in the system for clients was wait time, averaging 33.81 minutes (43.3% of total time). This number is not too surprising considering the large volume of clients in the system per day. What is surprising is the wait time per type of client: walk-ins averaged

the shortest wait time (32.6 minutes), followed by intakes (33.3 minutes), and then scheduled clients (35.6 minutes). Further analysis of this showed that many clients were scheduled during peak hours, so although they're given precedence over unscheduled clients, they still have to wait because of unscheduled clients who may have come in before them. The averages for the unscheduled clients are a bit lower because their arrival is more spread out throughout the day. This problem could be mitigated by creating a more effective scheduling system to encourage more clients, especially intakes, to book their appointments in advance.

Looking at the work in progress for each of client types further demonstrates the strain that intakes can cause to the system. On average, for scheduled clients, walk-ins, and intakes, there were 11.8, 12.3, and 39.6 units of work in progress at any given time. It is known that interviews for intakes generally take longer than clients who are active in the system due to pre-screening procedures and adding or updating information into the KAMES system. Some steps have been taken recently to try to reduce this number, such as extending the period between recertification meetings and encouraging clients to recertify by phone or mail.

The overall work in progress chart, shown below in Figure 5.1 (a larger view can be found in Appendix VI), gives a look at the times during the day in which the system is the busiest. At around 9:30 AM, the WIP makes a big jump where it remains relatively steady until noon, where the system reaches its peak. Many clients, especially intakes, come into the system around 11 AM. After noon, the WIP starts to drop off and steadily decreases throughout the rest of the day. Due to the large amount of clients coming in



around noon, many case workers take staggered lunch breaks early or late in the day, so they can be available to work during peak times.

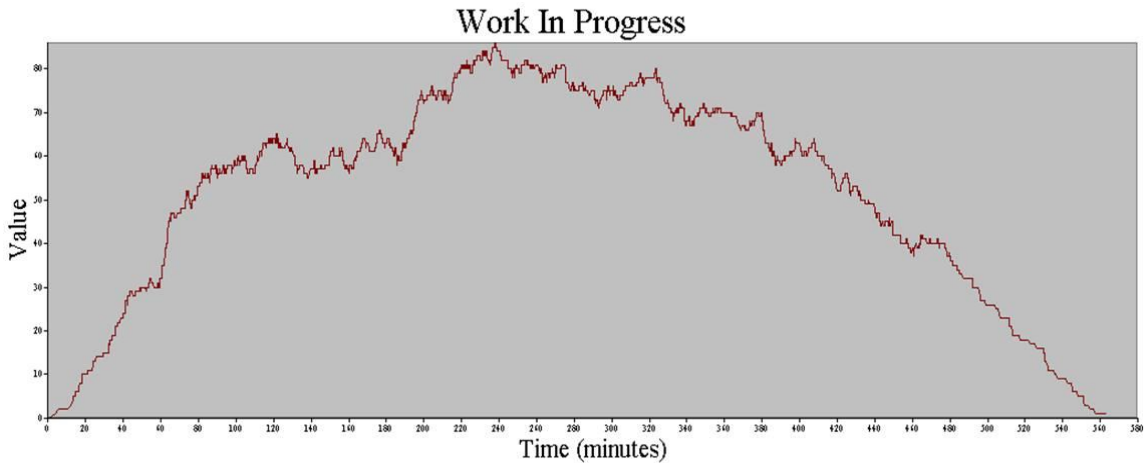


FIGURE 5.1 – Chart Tracking WIP Value over Time in the Present Model

Analyzing the queues and utilization uncovers another underlying issue that could go overlooked in the model. The average waiting time for the 3<sup>rd</sup> floor waiting room was found to be 17.88 minutes, while the 4<sup>th</sup> floor waiting room was 27.31 minutes. Furthermore, the average number waiting for the 3<sup>rd</sup> and 4<sup>th</sup> floor waiting rooms were 6.1 and 9.9 clients respectively. With the processing times for each of the floors being very similar, it was hard to determine why there was such a discrepancy between the floors. The utilization numbers showed that the 4<sup>th</sup> floor was typically busier than the 3<sup>rd</sup>, with utilization rates of 83.6% on the 4<sup>th</sup> floor compared to 78.0% on the 3<sup>rd</sup> floor. The answer to the question lay in the distribution of clients to each floor, 48% to the 3<sup>rd</sup> and 52% to the 4<sup>th</sup>. At first glance, those numbers may seem pretty even, but over the course of time, the 4% difference adds up. According to the output, the 4<sup>th</sup> floor averages 16 more clients than the 3<sup>rd</sup> floor per day. Since both floors have an equal number of case workers to process the clients, it's easy to see why the 4<sup>th</sup> floor is busier.

The problem is that clients are assigned to each floor based on the zip code of their residence. As times change, so do the economic demographics of any populated region. It occurred to the team that the distribution may have been even when the boundaries were originally set in place, but has probably shifted over time. The reason for the distribution by zip code is filing of paperwork (i.e. knowing exactly where a case file would be based on the zip code of the client). It seems that the optimal solution to the problem would be a paperless system, eliminating the boundaries for the floors. However, suggesting such a large change without further evidence is outside the scope of this project. For an immediate fix, the team found that by swapping zip codes 40202 and 40208, the distribution would change to 49.1% on the 3<sup>rd</sup> floor and 50.9% on the 4<sup>th</sup> floor. While the distribution is still not even, a 1.8% difference is much better than a 4% difference, and helps even out the utilization.

### **Sign In Kiosk Experiment**

The sign in kiosk aims to make changes in the main lobby, but should result in changes throughout the system once implemented. When the changes were made to the system, average wait time and total time in the system shrunk. The average number of receptionists busy in the main lobby and main lobby receptionist utilization was also affected. A chart comparing the effects of the kiosk system is shown below in Table 5.2. Paired t-tests were run to compare the original model and the kiosk system model at a 95% confidence interval to determine if the differences are significant.

TABLE 5.2

## KIOSK EXPERIMENT VS. PRESENT MODEL SIMULATION OUTPUT RESULTS

Variable	Comparison			
	Original Model	Kiosk Model	% Change	Significant Difference
Average Client Wait Time	33.8 min	33.5 min	0.90%	No
Average Client Time in System	78.15 min	77.26 min	1.10%	No
Average Main Lobby Receptionist Utilization	63.6%	41.3%	35.0%	Yes
Average Number Receptionists Busy	2.54	1.65	35.1%	Yes
Average Main Lobby Queue Wait Time	5.25 min	0.40 min	92.3%	Yes
Maximum Main Lobby Queue Wait Time	20.8 min	10.0 min	51.9%	Yes

The average wait time for a client in the system is reduced from 33.8 minutes in the original model to 33.5 minutes, a change of 0.9%. The average total time a client spends in the system is reduced from 78.15 minutes to 77.26 minutes, a change of 1.1%. According to the t-tests results, these differences are not significant. However, it is important to remember that only the main lobby modules were changed in this experiment. It is likely that wait times would be slightly reduced on each floor as well, since the receptionist for the floors would not have to notify the case worker that the client has arrived via agency contact form. However, this is speculative and difficult to simulate, so it was left out of the simulation. All other t-tests resulted in significant differences for the variables listed above.

The utilization rates for the main lobby receptionist show a positive change, with a decrease from 63.6% in the original model to 41.3% in the experiment. This made the average number of busy receptionists in the main lobby drop from 2.54 to 1.65. This

frees up one of the main lobby receptionist positions, assuming that at least 35% of clients will actually use the kiosks. The kiosks also have capacity to spare, averaging only 19.1% utilization. Comparing the two sign in procedures more closely reveals that average wait time for the kiosk is only 0.04 minutes with a maximum of 3.7 minutes, while the wait time for the main lobby receptionist is 0.40 minutes with a maximum of 10.0 minutes. It's an even larger change when compared to the original system, where the average wait time for the main lobby receptionist is 5.25 minutes, with a maximum of 20.8 minutes.

Based on these findings, a kiosk system in the main lobby would help reduce wait times there. It is also likely that wait times throughout the whole system would be reduced, as paperwork is eliminated through the implementation of the kiosks.

### **10% Client Influx Experiment**

This experiment differs from the rest, as it is a predictive model to prepare the KCHFS for the future. With the whole healthcare system undergoing reform and the unemployment rate still up in the state of Kentucky, this model is important to understand how the system acts under increased constraints. Running this model gives an estimate of how much wait times, utilization rates, and total times increase, as seen below in Table 5.3. It is assumed that utilization rates for the kiosk system would go higher than 35% over time as clients became more comfortable with the new system.

With a 10% increase of clients into the system, the number of entities processed rises from 393 to 439. With that, the average wait time in the system increased from 33.81 minutes in the original model to 55.34 minutes, an increase of 38.9%. The total time a

client spends in the system also increased from 78.15 minutes to 99.74 minutes, or 21.6%. Maximum wait time and total time for the experiment are 186 minutes and 264 minutes, respectively. Maximum client time in system is the only compared variable in this experiment that is not significantly different.

TABLE 5.3

10% CLIENT INFLUX EXPERIMENT VS. PRESENT MODEL SIMULATION  
OUTPUT RESULTS

Variable	Comparison			
	Original Model	Influx Model	% Change	Significant Difference
Average Client Wait Time	33.8 min	55.3 min	38.9%	Yes
Max Client Wait Time	155.7 min	186.4 min	16.5%	Yes
Average Client Time in System	78.2 min	99.7 min	21.6%	Yes
Max Client Time in System	240.4 min	264.2 min	9%	No
Average Work in Progress	21.25	27.18	21.8%	Yes
Average Number Clients Per Day	393	439	10.4%	Yes
Average Case Worker Utilization	80.77%	83.71%	3.5%	Yes

Other metrics that increase in the influx model are work in progress and case worker utilization. Average work in progress in the system rose from 21.25 units to 27.18 units, which is consistent with Little's Law [26]. Along with it, the average case worker utilization rose in the influx model to 83.71%. With utilization rates so high, case workers will be under increased pressure to complete their work in a timely manner, assuming the number of case workers remains consistent in the system. Further tests were run to determine the number of case workers it would take to reduce the utilization back down below 80%. Experimentation showed that the lowest number of case workers needed to reduce the utilization rates back below 80% were 21 on the 3<sup>rd</sup> floor and 22 for the 4<sup>th</sup> floor.

#### 4<sup>th</sup> Floor Alternative Case Processing Experiment

When the alternative 4<sup>th</sup> floor case processing procedure was in its pilot phase, it was met with some resistance and skepticism. Workers were unsure of whether the new process was faster or not; and if it was not faster, they questioned why they were doing it. This experiment was conducted to determine if this system is more efficient, and if so, to gain buy in from the case workers. As seen from the results in Figure 5.4, the proposed method processes clients faster and reduces the time the client spends in the system.

TABLE 5.4

#### 4<sup>th</sup> FLOOR ALTERNATIVE METHOD EXPERIMENT VS. PRESENT MODEL SIMULATION OUTPUT RESULTS

Variable	Comparison			
	Original Model	4F Change Model	% Change	Significant Difference
Average Client Wait Time	33.8 min	30.8 min	8.9%	No
Average Client Time in System	78.2 min	72.0 min	7.9%	Yes
Average Work in Progress	21.25	19.96	6.1%	Yes
Average 4F Case Worker Utilization	83.6%	80.8%	3.3%	Yes
Average 4F Number Waiting	9.9	7.8	21.2%	Yes
Average 4F Queue Wait Time	27.3 min	20.7 min	24.1%	Yes

Some of the most noticeable differences are the reductions in wait time and number waiting on the 4<sup>th</sup> floor. Recall that both the metrics in the original model were higher than those on the 3<sup>rd</sup> floor due to an unbalanced client distribution. With this change on the 4<sup>th</sup> floor, the average number waiting drops to 7.8 (21.2% change from the original model) and the queue wait time drops to 20.7 minutes (24.1% change from the original model). These numbers are now much closer to the 3<sup>rd</sup> floor numbers (6.1 average

number waiting and 17.8 minutes for queue wait time), with a reduced number of interviewers.

The average client wait times and total time in system are both reduced as well. While the difference in total time is significantly different, the reduction in wait times is not. It is important to note that these numbers include averages from the 3<sup>rd</sup> floor as well, which remained unchanged. However, just changing the 4<sup>th</sup> floor process dropped average client wait times to 30.8 minutes. Total time was reduced to 72 minutes due to shorter wait times and processing times on the 4<sup>th</sup> floor. Though a pilot program has not been run for the 3<sup>rd</sup> floor, it is estimated that times would decrease even further if the 3<sup>rd</sup> floor adopted this change as well.

At first glance, it appears that the case worker utilization for the 4<sup>th</sup> floor does not change very much from the original model to the alternative model. However, keep in mind that the number of resources for processing clients was reduced from 19 to 16, since some workers were allocated to process the case files. Work in progress for the system was reduced due to the decrease in cycle time on the 4<sup>th</sup> floor. Just as with the wait time and total time for the system, this number could see a further reduction if the 3<sup>rd</sup> floor adopted the new interview system as well. The case workers dedicated to interviewing are able to process more clients in less time without having to worry about working on the case files during or between interviews. In the same respect, the processors only have to worry about getting the case files processed and don't have client interviews to do. It is also possible that numbers could decrease further as workers become more accustomed to the new system. These numbers were all collected during the pilot phase, while

workers were still adjusting to the new system. Steady-state numbers may differ from these results.



## VI. CONCLUSION

The current model output shows that clients deal with long wait times in the system, averaging 33.8 minutes. This is around 43% of the average total time spent in the system, which is 78.2 minutes. Much of the client traffic comes from the large number of intakes, about 39% of the client population, that enter the system on any given busy day. If current economic trends persist and an influx in the number of clients occurs, numbers will continue to increase. A theoretical experiment with a 10% increase in clients showed that average wait times would increase by over 20 minutes. Using the present simulation model to test theoretical changes can help find methods to improve the system and decrease wait times.

Two proposed changes to the system are the implementation of kiosk systems in the main lobby, and adoption of a revised method to process clients. Experimentally observed values show that adding kiosk systems to check clients into the system in the main lobby allow average main lobby wait times to drop by 92%, from 5.25 minutes to 0.4 minutes. The pilot program for the 4<sup>th</sup> floor with separate interviewers and processors showed decreased wait time for clients as well. The average number waiting on the 4<sup>th</sup> floor was reduced from 9.9 to 7.8. The average waiting time in the 4<sup>th</sup> floor waiting room dropped from 27.3 minutes to 20.7 minutes. Implementing the program on the 3<sup>rd</sup> floor could result in similar changes, further decreasing client cycle time for the whole system.

## VII. RECOMMENDATIONS

To further this study, a more detailed long-range model would be a great asset to use in combination with this model. Though this model replicates an average busy day in the system, it does not track important cyclical trends that occur over time. As times and processes change, so must the daily model. A model that tracked the progress of clients and case workers over a longer period (3 months or more) would be a very valuable tool.

It would also be beneficial to create models to replicate each different program, as suggested in by Cayirli, Rosen, and Veral [19]. The model would be especially useful if it tracked cohorts of clients from when they first applied to the program to when they go off the program. Since each program has different requirements, separate models would help identify inefficiencies at a micro level, whereas the daily model analyzes the system at a macro level.

The daily model itself has its shortcomings, such as the lack of scheduled breaks and the large processing time distributions. It was difficult to create distributions that encompassed processing times for experienced case workers as well as newcomers. The model would be more in depth if the case workers were clustered based on experience, and then processing times were gathered for each cluster.

## REFERENCES

- [1] Steitzer, S. 2010. "Kentucky State Workers to be Furloughed 6 Days." *The Courier Journal (Louisville, KY)*. pp. A1. July 9.
- [2] Swiss, J. 1992. Adapting Total Quality Management (TQM) to Government. *Public Service Administration*, Vol. 52, No. 4, pp. 356-362
- [3] Brodtkin, E. 2007. Bureaucracy Redux: Management Reformism and the Welfare State. *Journal of Public Administration Research and Theory*. Vol. 17, No. 1, pp. 43-56.
- [4] Maleyeff, J., 2007. Improving Service Delivery in Government with Lean Six Sigma. *IBM Center for The Business of Government (Washington, DC)*. p. 32.
- [5] Grant, V. C., 2010. *Transforming State Government Services through Process Improvement: A Case Study of Louisiana*, IBM Center for the Business of Government (Washington, DC). pp. 5-23.
- [6] Adivar, B., Atan, T., Oflaç B., & Örtten, T. 2010. Improving social welfare chain using optimal planning model. *Supply Chain Management: An International Journal*, Vol. 15, No. 4, pp. 290-305
- [7] Johnson, M. 2009. How Can Community-Based Practitioners Improve Planning and Operations with Management Science? *INFORMS Annual Conference (San Diego, CA)*.
- [8] Gurumurthy, A, Kodali, R., 2011. Design of lean manufacturing systems using value stream mapping with simulation: A case study. *Journal of Manufacturing Technology Management*, Vol. 22, No. 4, pp.444 – 473
- [9] Millard, R. 2001. Value Stream Analysis and Mapping for Product Development. Master of Science dissertation, Massachusetts Institute of Technology.
- [10] Issel, L. M., Narasimha, K. 2007. Creating complex health improvement programs as mindful organizations: From theory to action. *Journal of Health Organization and Management*. Vol. 21, No. 2, pp.166 – 183

- [11] Israel, N., Wolf-Branigin, M. 2011. Nonlinearity in Social Service Evaluation: A Primer on Agent-based Modeling. *Social Work Research*. Vol.35, No.1, pp. 20-24.
- [12] Seddon, J., 2005. *Freedom from Command and Control: Rethinking Management for Lean Service*. New York, NY: Productivity Press
- [13] Pot, F. 2011. Workplace innovation for better jobs and performance. *International Journal of Productivity and Performance Management*. Vol. 60, No. 4, pp. 404-415.
- [14] Kelton, W., Sadowski, R., & Sturrock, D. 2007. *Simulation with Arena: Fourth Edition*. New York, NY: McGraw-Hill.
- [15] Zeigler B., Praehofer H., & Kim T. G., 2000. *Theory of Modeling and Simulation: Second Edition*. New York, NY: Academic Press.
- [16] Grabau, M., Sadowski, D. 2003. Tips for Successful Practice of Simulation. *Winter Simulation Conference (New Orleans, LA)*. Proceedings, 31-36
- [17] Law, A. 2007. *Simulation Modeling & Analysis: Fourth Edition*. New York, NY: McGraw- Hill.
- [18] Evans, G. W., Gor, T., & Unger, E. 1996. A Simulation Model for Evaluating Personnel Schedules in a Hospital Emergency Department. *Winter Simulation Conference (San Diego, CA)*. Proceedings, pp.1205-1209
- [19] Cayirli, T., Rosen, H., & Veral, E. 2008. Assessment of Patient Classification in Appointment System Design. *Production and Operations Management*. Vol. 17, No. 3, pp. 338-353
- [20] Nguyen, H., Godbole, G., Kalkundri, K., & Lam, S. 2009. Simulation of a Food Warehouse for a Hunger Outreach Program. *IIE Annual Conference (Miami, FL)*. Proceedings, pp. 1646-1651.
- [21] Deck, S., Barber, G., & Morris, V., 2010. *Systematic Review of Kentucky Works Program Participation Patterns: Annual Update for Jefferson County and the Northern and Southern Bluegrass Regions*, University of Louisville (Louisville, KY) pp 1-30, 56-79
- [22] Sepulveda, J., Thompson, W., Baesler, F., & Alvarez M., 1999. The use of simulation for process improvement in a cancer treatment center. *Winter Simulation Conference (Phoenix, AZ)*. Proceedings, pp. 2: 1541-1548
- [23] Rother, M., Shook, J. 1999. *Learning to See*. Boston, MA: Lean Enterprise Institute

- [24] Gossett, E. 2009. *Discrete Mathematics with Proof: Second Addition*. Hoboken, NJ: John Wiley & Sons, Inc.
- [25] Arena v13.0. Rockwell Automation Technologies, Inc., Milwaukee, WI
- [26] Hopp, W., Spearman, M. 2008. *Factory Physics: Third Edition* – New York, NY: McGraw-Hill

APPENDIX I

ENLARGED SIMULATION MODEL

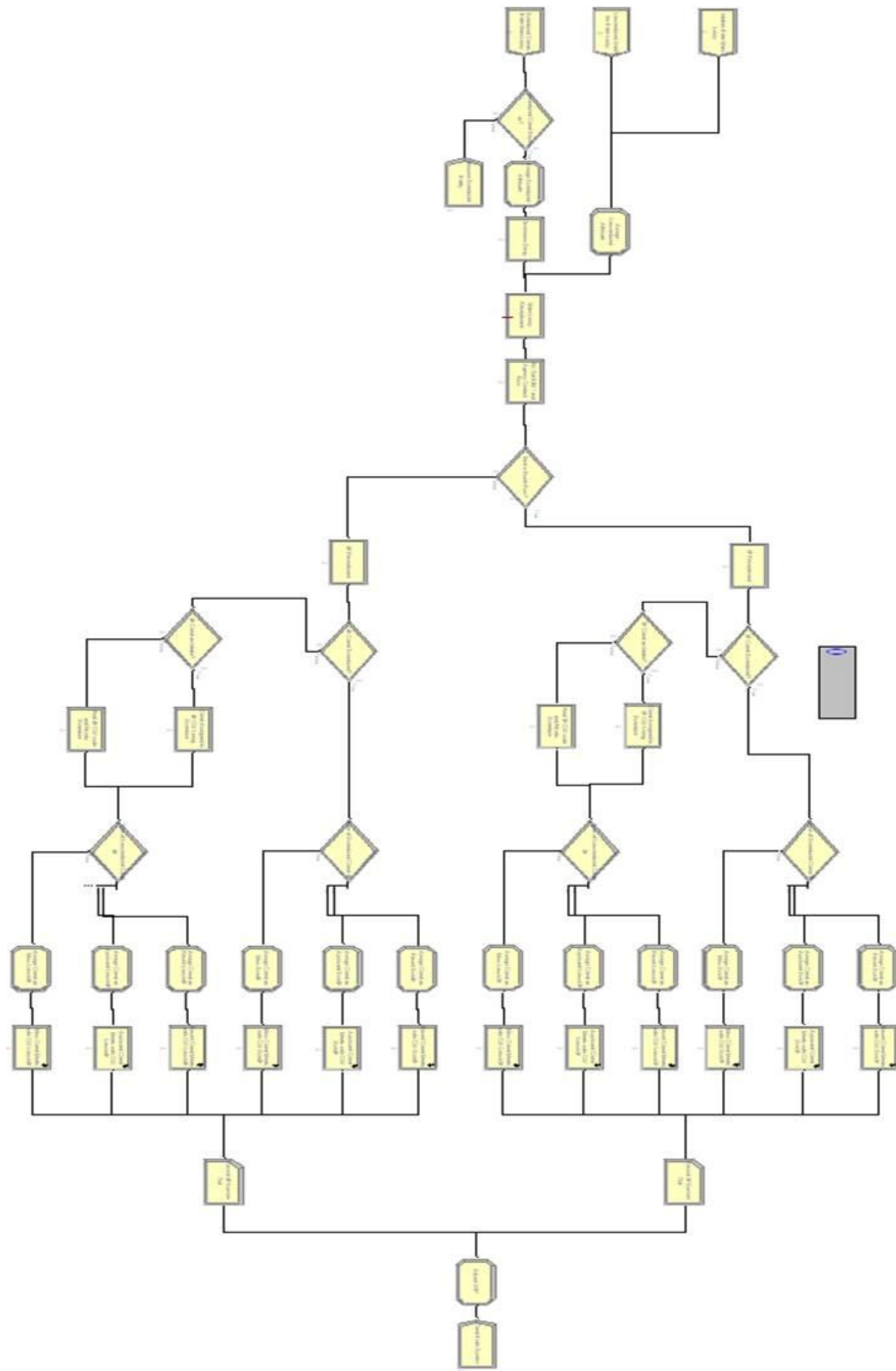


FIGURE A.1 – Enlarged Full View of Simulation Model Image

APPENDIX II

ENTITY ARRIVAL SCHEDULE TABLES



TABLE A.1

CLIENT ENTITY ARRIVAL SCHEDULES FOR SIMULATION MODEL

Time	Original, Kiosk, and 4F Process Change Model Client Arrivals			10% Influx Model Client Arrivals		
	Scheduled	Intake	Walk- In	Scheduled	Intake	Walk- In
8:00 AM	11	23	7	12	26	9
8:30 AM	23			26		
9:00 AM	37	47	13	40	52	14
9:30 AM	26			29		
10:00 AM	40	29	10	44	32	12
10:30 AM	23			26		
11:00 AM	38	44	14	41	49	15
11:30 AM	28			30		
12:00 PM	5	30	10	6	32	12
12:30 PM	0			0		
1:00 PM	1	33	9	2	39	10
1:30 PM	11			12		
2:00 PM	29	24	8	31	27	9
2:30 PM	18			20		
3:00 PM	8	14	5	9	16	6
3:30 PM	2			3		
4:00 PM	1	0	0	1	0	0
4:30 PM	0			0		
5:00 PM	0	0	0	0	0	0

APPENDIX III

ENLARGED SUB-MODEL VIEW

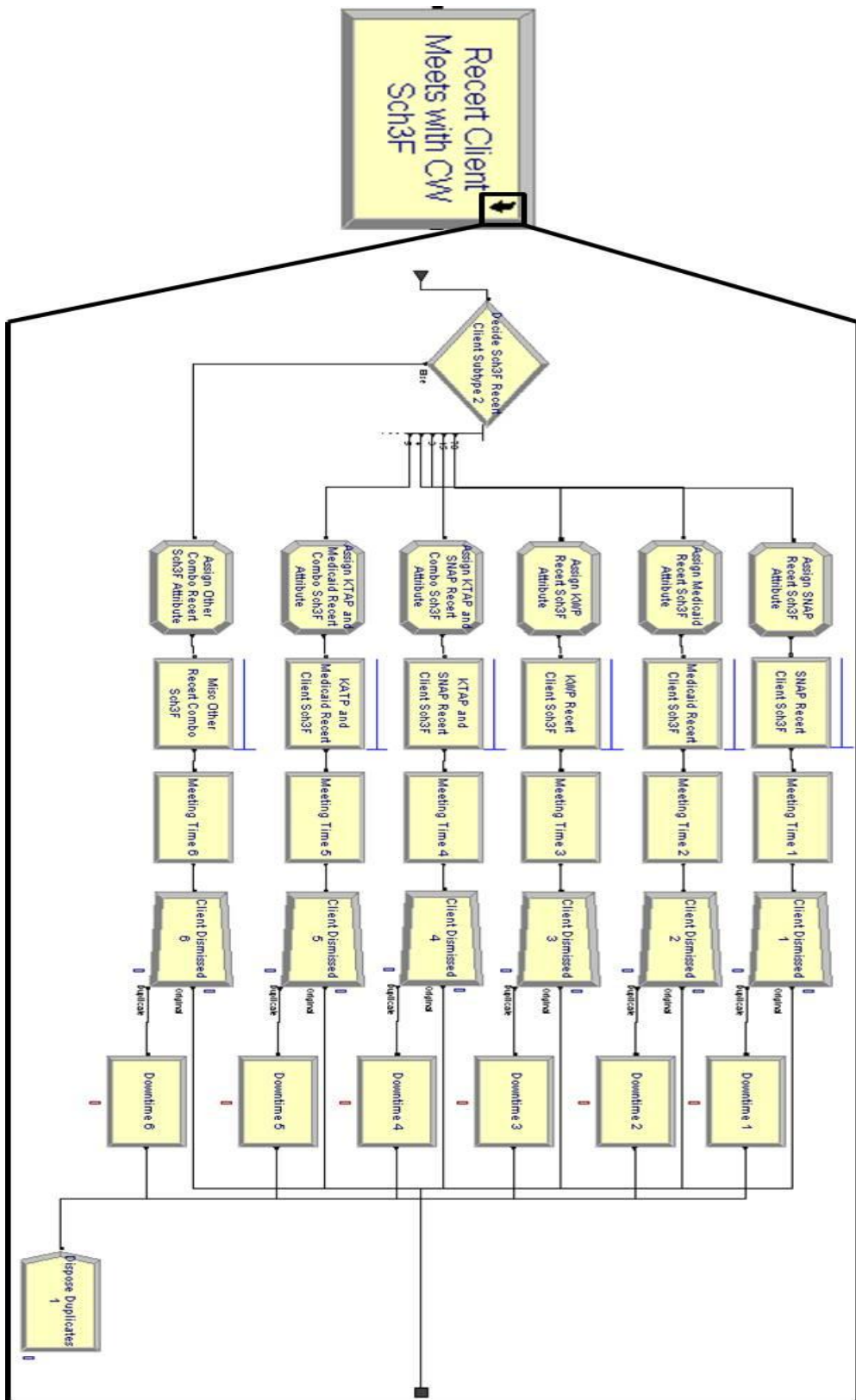


FIGURE A.2 – Enlarged Sub-Model View for Processing

APPENDIX IV

PROCESSING TIMES EXPRESSION TABLES

TABLE A.2

3F SCHEDULED PROCESSING TIMES EXPRESSIONS

	3F Scheduled Clients Processing Times (All Models)					
	SNAP	MA	KWP	KTAP/SNAP	KTAP/MA	Other
<b>Recert</b>	TRIA(8,21,35)	TRIA(19,32,42)	TRIA(33,39,55)	TRIA(24,34,60)	TRIA(32,39,59)	TRIA(12,18,28)
<b>Applicant</b>	TRIA(18,34,45)	TRIA(29,44,60)	TRIA(50,60,85)	TRIA(33,56,68)	TRIA(50, 60, 85)	TRIA(16,20,30)
<b>Misc.</b>	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)

TABLE A.3

3F UNSCHEDULED PROCESSING TIMES EXPRESSIONS

	3F Unscheduled Clients Processing Times (All Models)					
	SNAP	MA	KWP	KTAP/SNAP	KTAP/MA	Other
<b>Recert</b>	TRIA(12,29,37)	TRIA(26,37,45)	TRIA(33,47,56)	TRIA(27,34,60)	TRIA(32,42,61)	TRIA(15,21,28)
<b>Applicant</b>	TRIA(23,40,48)	TRIA(31,48,60)	TRIA(56,64,90)	TRIA(40,56,70)	TRIA(56,64,90)	TRIA(18,22,30)
<b>Misc.</b>	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)

TABLE A.4

4F SCHEDULED PROCESSING TIMES EXPRESSIONS ORIGINAL MODEL

	4F Scheduled Clients Processing Times (Original, Kiosk, and Influx Models)					
	SNAP	MA	KWP	KTAP/SNAP	KTAP/MA	Other
<b>Recert</b>	TRIA(8,19,25)	TRIA(17,27,38)	TRIA(33,40,55)	TRIA(22,34,60)	TRIA(32,42,59)	TRIA(14,18,24)
<b>Applicant</b>	TRIA(18,32,40)	TRIA(28,44,59)	TRIA(50,60,85)	TRIA(34,55,67)	TRIA(49,60, 85)	TRIA(18, 24, 31)
<b>Misc.</b>	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)

TABLE A.5

4F UNSCHEDULED PROCESSING TIMES EXPRESSIONS ORIGINAL MODEL

	4F Unscheduled Clients Processing Times (Original, Kiosk, and Influx Models)					
	SNAP	MA	KWP	KTAP/SNAP	KTAP/MA	Other
<b>Recert</b>	TRIA(10,24,33)	TRIA(26,32,46)	TRIA(40,49,62)	TRIA(28,38,61)	TRIA(32,42,61)	TRIA(14,25,30)
<b>Applicant</b>	TRIA(24,34,48)	TRIA(34,48,61)	TRIA(56,69,90)	TRIA(45,60,75)	TRIA(59,70,90)	TRIA(20,29,35)
<b>Misc.</b>	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)

TABLE A.6

4F SCHEDULED PROCESSING TIMES EXPRESSIONS EXPERIMENT MODEL

	4F Scheduled Clients Processing Times (4F Processing Change Model)					
	SNAP	MA	KWP	KTAP/SNAP	KTAP/MA	Other
<b>Recert</b>	TRIA(4,19,30)	TRIA(7,21,35)	TRIA(31,40,55)	TRIA(19,23,40)	TRIA(12,28,49)	TRIA(5,19,34)
<b>Applicant</b>	TRIA(7,21,50)	TRIA(15,29,51)	TRIA(50,60,85)	TRIA(14,32,69)	TRIA(32,54, 79)	TRIA(8, 22, 45)
<b>Misc.</b>	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)

TABLE A.7

4F UNSCHEDULED PROCESSING TIMES EXPRESSIONS EXPERIMENT MODEL

	4F Unscheduled Clients Processing Times (4F Processing Change Model)					
	SNAP	MA	KWP	KTAP/SNAP	KTAP/MA	Other
<b>Recert</b>	TRIA(5,20,33)	TRIA(9,22,37)	TRIA(33,46,62)	TRIA(20,27,51)	TRIA(22,33,51)	TRIA(7,24,37)
<b>Applicant</b>	TRIA(7,24,55)	TRIA(16,33,61)	TRIA(56,68,89)	TRIA(26,38,76)	TRIA(34,55,80)	TRIA(10,25,45)
<b>Misc.</b>	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)	TRIA(2,3,5)

APPENDIX V

SIMULATION OUTPUT TABLES

TABLE A.8

OBSERVED CLIENT DATA SIMULATION OUTPUT FOR ORIGINAL MODEL

<b>Simulation Output for Present Day Model</b>					
	<b>Type of Client</b>				
<b>Metric</b>	<b>Intake</b>	<b>Scheduled</b>	<b>Walk-In</b>	<b>Average</b>	<b>Total</b>
Value Added Time (in minutes)	32.75	27.59	33.81	31.38	-
Non-Value Added Time (in minutes)	7.97	7.95	7.97	7.96	-
Wait Time (in minutes)	33.25	35.57	32.61	33.81	-
Transfer Time (in minutes)	5.00	5.00	5.00	5.00	-
Total Time (in minutes)	78.97	76.11	79.38	78.15	-
Total Number Clients Per Day	143.00	154.00	96.00	-	393.00
Work In Process	39.61	11.80	12.34	21.25	-

TABLE A.9

OBSERVED SYSTEM DATA SIMULATION OUTPUT FOR ORIGINAL MODEL

<b>Simulation Output for Present Day Model</b>				
<b>Metric</b>	<b>Minimum Average</b>	<b>Maximum Average</b>	<b>Average</b>	<b>Maximum Value</b>
Number of Clients to 3F	163.00	220.00	188.66	-
Number of Clients to 4F	167.00	243.00	204.50	-
3F Receptionist Queue Time (in minutes)	1.23	13.36	3.69	39.24
4F Receptionist Queue Time (in minutes)	0.99	21.13	4.66	43.89
3F Waiting Room Time (in minutes)	1.57	46.35	17.88	89.67
4F Waiting Room Time (in minutes)	2.30	55.00	27.31	130.00
Main Lobby Wait Time (in minutes)	0.86	20.84	5.25	44.21
3F Receptionist Queue Number Waiting	0.34	5.09	1.26	21.00
4F Receptionist Queue Number Waiting	0.35	7.82	1.71	25.00
3F Waiting Room Number Waiting	0.44	17.31	6.10	40.00
4F Waiting Room Number Waiting	0.74	20.74	9.96	51.00
Main Lobby Number Waiting	0.60	14.98	3.67	49.00
3F Case Worker Utilization	65.08%	89.66%	77.99%	100.00%
4F Case Worker Utilization	71.05%	90.59%	83.55%	100.00%
Main Lobby Receptionist Utilization	57.57%	69.44%	63.59%	100.00%
3F Case Worker Number Busy	12.36	17.04	14.82	19.00
4F Case Worker Number Busy	13.50	17.21	15.87	19.00
Main Lobby Receptionist Number Busy	2.30	2.78	2.54	4.00



TABLE A.10

OBSERVED CLIENT DATA SIMULATION OUTPUT FOR KIOSK EXPERIMENT MODEL

<b>Simulation Output for Main Lobby Kiosk Experiment Model</b>					
	<b>Type of Client</b>				
<b>Metric</b>	<b>Intake</b>	<b>Scheduled</b>	<b>Walk-In</b>	<b>Average</b>	<b>Total</b>
Value Added Time (in minutes)	33.73	28.01	34.26	32.00	-
Non-Value Added Time (in minutes)	6.65	6.66	6.66	6.66	-
Wait Time (in minutes)	32.50	35.30	33.04	33.53	-
Transfer Time (in minutes)	5.00	5.00	5.00	5.00	-
Total Time (in minutes)	77.88	74.97	74.96	77.26	-
Total Number Clients Per Day	143.00	155.00	96.00	-	394.00
Work In Process	39.29	11.52	12.69	21.17	-

TABLE A.11

OBSERVED SYSTEM DATA SIMULATION OUTPUT FOR KIOSK EXPERIMENT MODEL

<b>Simulation Output for Main Lobby Kiosk Experiment Model</b>				
<b>Metric</b>	<b>Minimum Average</b>	<b>Maximum Average</b>	<b>Average</b>	<b>Maximum Value</b>
Number of Clients to 3F	157.00	220.00	187.23	-
Number of Clients to 4F	176.00	245.00	206.73	-
3F Receptionist Queue Time (in minutes)	1.60	23.68	5.80	55.17
4F Receptionist Queue Time (in minutes)	1.60	26.23	7.63	59.50
3F Waiting Room Time (in minutes)	1.22	48.47	18.44	87.36
4F Waiting Room Time (in minutes)	3.51	57.61	30.51	118.20
Main Lobby Wait Time (in minutes)	0.09	1.50	0.40	10.04
3F Receptionist Queue Number Waiting	0.46	8.73	1.98	30.00
4F Receptionist Queue Number Waiting	0.57	10.34	2.84	33.00
3F Waiting Room Number Waiting	0.36	17.59	6.26	38.00
4F Waiting Room Number Waiting	1.15	21.15	11.30	51.00
Main Lobby Number Waiting	0.04	0.68	0.18	11.00
3F Case Worker Utilization	65.66%	90.28%	77.78%	100.00%
4F Case Worker Utilization	70.80%	91.95%	84.67%	100.00%
Main Lobby Receptionist Utilization	36.83%	47.77%	41.32%	100.00%
3F Case Worker Number Busy	12.48	17.15	14.77	19.00
4F Case Worker Number Busy	13.45	17.47	16.09	19.00
Main Lobby Receptionist Number Busy	1.47	1.91	1.65	4.00

TABLE A.12

OBSERVED CLIENT DATA SIMULATION OUTPUT FOR CLIENT INFLUX EXPERIMENT MODEL

<b>Simulation Output for 10% Client Influx Experiment Model</b>					
	<b>Type of Client</b>				
<b>Metric</b>	<b>Intake</b>	<b>Scheduled</b>	<b>Walk-In</b>	<b>Average</b>	<b>Total</b>
Value Added Time (in minutes)	32.75	27.72	33.85	31.44	-
Non-Value Added Time (in minutes)	7.96	7.94	7.97	7.96	-
Wait Time (in minutes)	55.13	54.89	55.99	55.34	-
Transfer Time (in minutes)	5.00	5.00	5.00	5.00	-
Total Time (in minutes)	100.84	95.96	102.81	99.74	-
Total Number Clients Per Day	172.53	159.80	106.68	-	439.00
Work In Process	50.63	14.37	16.56	27.19	-

TABLE A.13

OBSERVED SYSTEM DATA SIMULATION OUTPUT FOR CLIENT INFLUX EXPERIMENT MODEL

<b>Simulation Output for 10% Client Influx Experiment Model</b>				
<b>Metric</b>	<b>Minimum Average</b>	<b>Maximum Average</b>	<b>Average</b>	<b>Maximum Value</b>
Number of Clients to 3F	178.00	258.00	210.86	-
Number of Clients to 4F	187.00	258.00	228.18	-
3F Receptionist Queue Time (in minutes)	1.49	25.93	5.23	54.70
4F Receptionist Queue Time (in minutes)	1.82	27.03	6.36	58.75
3F Waiting Room Time (in minutes)	7.84	53.52	29.84	120.07
4F Waiting Room Time (in minutes)	9.30	68.38	41.82	133.26
Main Lobby Wait Time (in minutes)	2.29	24.89	11.98	53.46
3F Receptionist Queue Number Waiting	0.42	10.09	1.85	29.00
4F Receptionist Queue Number Waiting	0.61	10.62	2.40	33.00
3F Waiting Room Number Waiting	2.29	20.33	10.48	52.00
4F Waiting Room Number Waiting	3.15	26.56	15.68	62.00
Main Lobby Number Waiting	1.46	17.43	8.64	60.00
3F Case Worker Utilization	66.21%	90.84%	81.02%	100.00%
4F Case Worker Utilization	74.96%	91.86%	86.40%	100.00%
Main Lobby Receptionist Utilization	58.29%	71.45%	65.89%	100.00%
3F Case Worker Number Busy	12.58	17.26	15.39	19.00
4F Case Worker Number Busy	14.24	17.45	16.42	19.00
Main Lobby Receptionist Number Busy	2.33	2.86	2.63	4.00

TABLE A.14

OBSERVED CLIENT DATA SIMULATION OUTPUT FOR 4F PROCESSING  
CHANGE EXPERIMENT MODEL

<b>Simulation Output for 4th Floor Alternative Method Experiment Model</b>					
	<b>Type of Client</b>				
<b>Metric</b>	<b>Intake</b>	<b>Scheduled</b>	<b>Walk-In</b>	<b>Average</b>	<b>Total</b>
Value Added Time (in minutes)	29.15	25.39	30.14	28.23	-
Non-Value Added Time (in minutes)	7.96	7.95	7.94	7.95	-
Wait Time (in minutes)	29.87	33.09	29.72	30.89	-
Transfer Time (in minutes)	5.00	5.00	5.00	5.00	-
Total Time (in minutes)	71.97	71.42	72.80	72.06	-
Total Number Clients Per Day	144.00	155.00	96.00	-	395.00
Work In Process	36.86	11.24	11.90	20.00	-

TABLE A.15

OBSERVED SYSTEM DATA SIMULATION OUTPUT FOR 4F PROCESSING  
CHANGE EXPERIMENT MODEL

<b>Simulation Output for 4th Floor Alternative Method Experiment Model</b>				
<b>Metric</b>	<b>Minimum Average</b>	<b>Maximum Average</b>	<b>Average</b>	<b>Maximum Value</b>
Number of Clients to 3F	160.00	238.00	189.79	-
Number of Clients to 4F	171.00	244.00	205.57	-
3F Receptionist Queue Time (in minutes)	0.81	15.92	3.67	36.84
4F Receptionist Queue Time (in minutes)	1.19	19.16	4.96	39.90
3F Waiting Room Time (in minutes)	0.54	46.02	17.60	110.39
4F Waiting Room Time (in minutes)	0.57	46.95	20.69	102.51
Main Lobby Wait Time (in minutes)	0.68	18.16	5.61	51.00
3F Receptionist Queue Number Waiting	0.23	5.79	1.28	20.00
4F Receptionist Queue Number Waiting	0.40	7.53	1.88	24.00
3F Waiting Room Number Waiting	0.16	16.75	6.16	45.00
4F Waiting Room Number Waiting	0.19	18.99	7.84	47.00
Main Lobby Number Waiting	0.45	13.73	4.05	55.00
3F Case Worker Utilization	67.81%	92.42%	80.21%	100.00%
4F Case Worker Utilization	67.21%	91.02%	80.88%	100.00%
Main Lobby Receptionist Utilization	58.11%	71.38%	65.34%	100.00%
3F Case Worker Number Busy	12.88	17.56	15.24	19.00
4F Case Worker Number Busy	10.73	14.56	12.94	16.00
Main Lobby Receptionist Number Busy	2.32	2.86	2.61	4.00

APPENDIX VI

ENLARGED WIP CHART

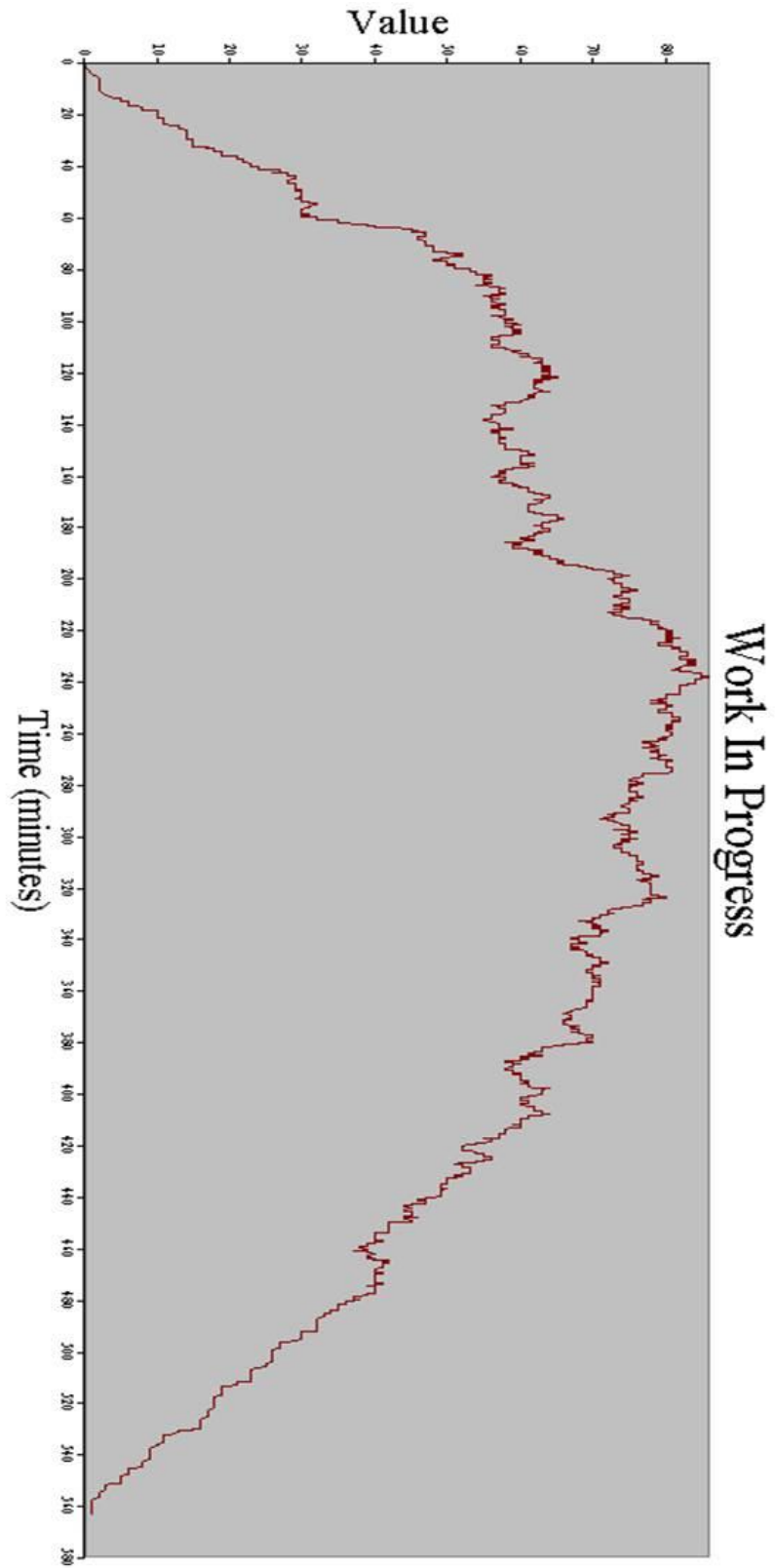


FIGURE A.3 – Enlarged WIP Chart

## VITA

NAME: Russell Edward Harpring

PERMANENT ADDRESS: 3731 Willmar Ave.  
Louisville, KY 40218

EDUCATION & TRAINING: B.S., Industrial Engineering  
University of Louisville  
2005-10

M.Eng., Industrial Engineering  
University of Louisville  
2010-11

PROFESSIONAL SOCIETIES: Institute of Industrial Engineers, Engineers Without Borders

### NATIONAL MEETING

#### PRESENTATIONS:

“UV Safety and the Blue Light Hazard”  
MegaRust 2008 Annual Meeting – August 2008  
Louisville, KY

“Simulating the Client Flow of the Kentucky  
Cabinet for Health and Family Services”  
Institute of Industrial Engineers Great Lakes Annual  
Student Meeting – February 2011  
Athens, OH

“Using Software Simulation to Improve Income  
Support Client Flow Process”  
NAWRS Annual Meeting 2011 – September 2011  
Vail, CO

“Simulating the Client Flow of the Kentucky  
Cabinet for Health and Family Services”  
INFORMS Annual Meeting 2011 – November 2011  
Charlotte, NC